ORACLE University

Integrated Cloud Applications & Platform Services

Oracle Linux System Administration I

Activity Guide – Volume II

D103151GC10

Edition 1.0 | February 2019 | D106144

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

ORACLE

Author

Eric Craig

Technical Contributors and Reviewers

Michael Oreilly
Antoinette O'Sullivan
Craig McBride
Michael O'Reilly
Dave Goff
Steve Miller
Rowan Puttergill
Nita Heieck
Craig Carl
Wim Coekaerts
Sergio Leunissen
Keshav Sharma
Hanlin Chien
Jim Williams
Ankur Kemkar
Jared Greenwald
Lawrence Gabriel
Tim Caster
Steven B. Nelson
Harish Niddagatta
Sebastien Colas
Matt Slingsby
Al Flournoy
Jamie Iles
Simon Coter
Avi Miller
John Haxby
Saar Maoz
Todd Vierling
Sreejith Mohan
Paul Kasewurm
Corey Leong

Graphic Designer

Yogita Chawdhary

Editors

Raj Kumar
Moushmi Mukherjee

Publishers

Giri Venugopal
Sumesh Koshy
Raghunath M

Table of Contents

Practices for Lesson 12: Partitions, File Systems, and Swap

Practices for Lesson 12: 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 12-1: 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 **dom0**.

Tasks

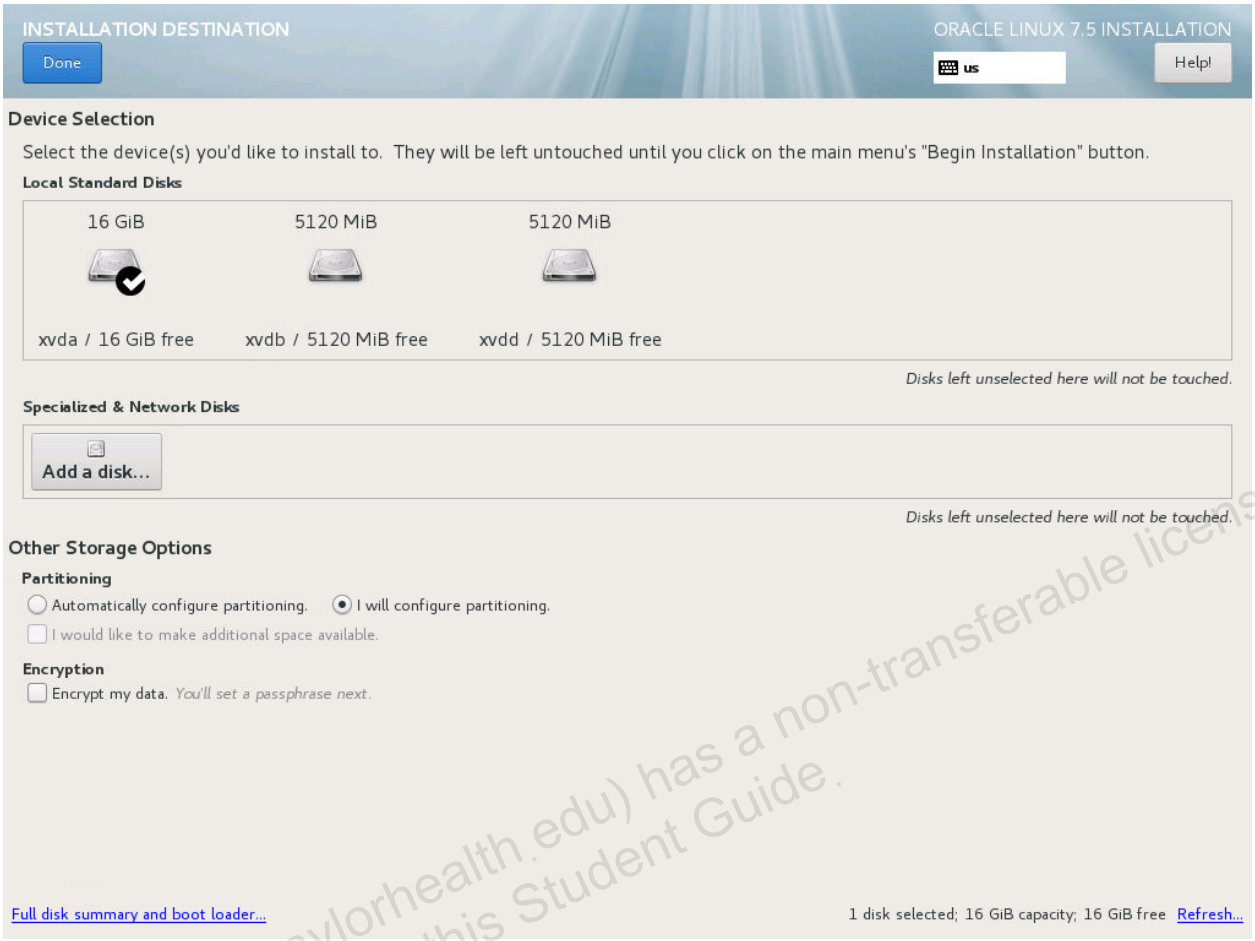
1. Log in to **host03**.
   a. Connect to the **host03** guest by using the `xm vncviewer host03&` command.

```
# xm vncviewer host03&
```

   The GNOME login window appears.
   b. Select Oracle Student from the GNOME login window; enter the password.
   c. Right-click the GNOME desktop and select **Open Terminal** from the pop-up menu.
   d. In the terminal window, become the `root` user by entering the `su -` command and providing the `root` password.

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

2. Relate the partition table to selections made during installation.
   - Three virtual disk images were created before initiating the installation:
     - A 16 GiB disk image (`system.img`) for the operating system
     - A 5 GiB disk image (`u01.img`) for the various storage administration practices
     - A 5 GiB disk image (`u02.img`) for the various storage administration practices
   - These three storage devices were presented during the installation, as shown in the following screenshot:

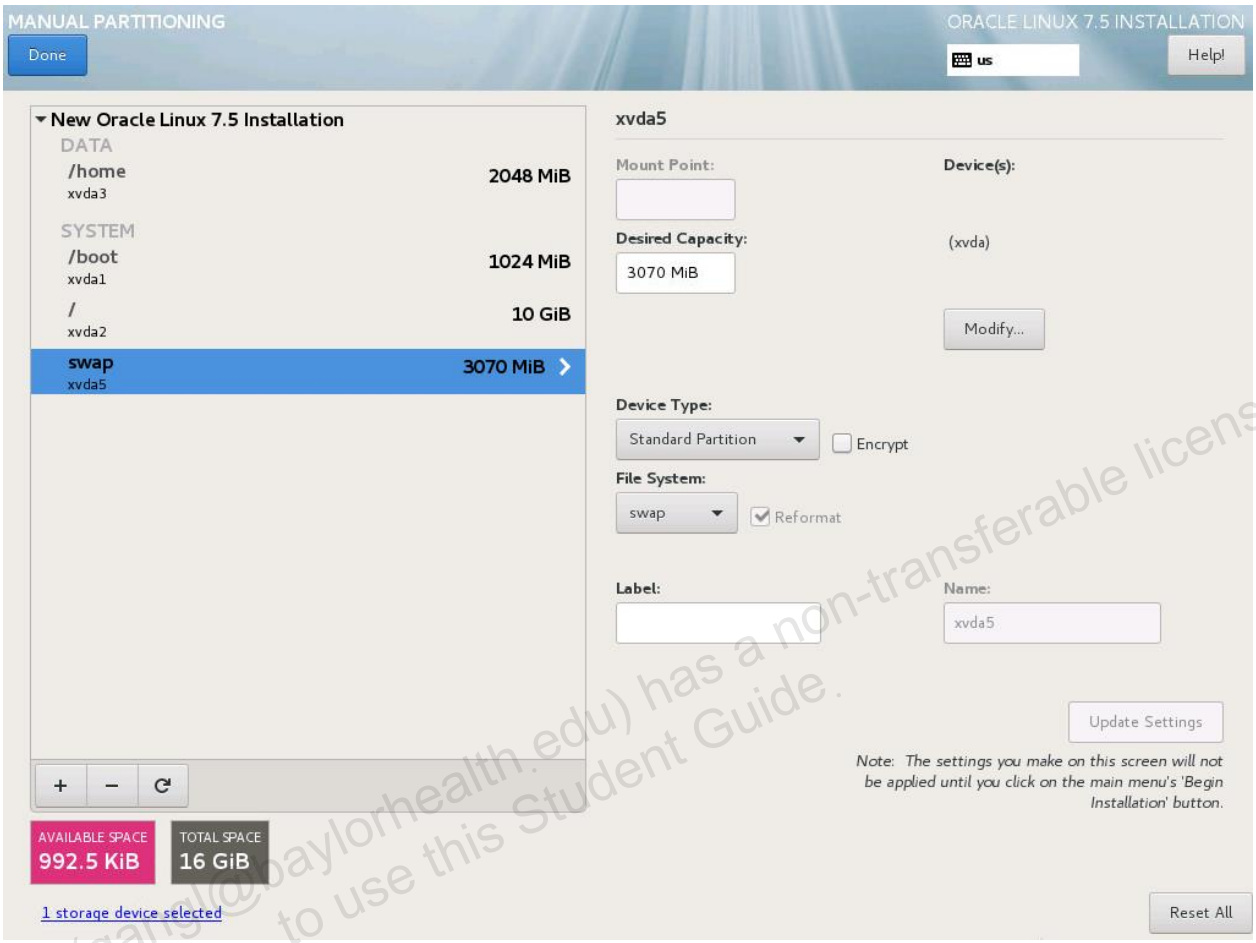


a. Use the `fdisk` command to display the partition table.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 17.2 GB, 17179869184 bytes, 33554432 sectors
/dev/xvda1   *        2048     2099199     1048576   83  Linux
/dev/xvda2         2099200    23070719    10485760   83  Linux
/dev/xvda3        23070720    27265023     2097152   83  Linux
/dev/xvda4        27265024    33554431     3144704    5
Extended
/dev/xvda5        27267072    33554431     3143680   82  Linux
swap / Solaris
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 17 GB in size (16 GiB in binary units)
  - `/dev/xvdb`, approximately 5 GB in size (5 GiB in binary units)
  - `/dev/xvdd`, approximately 5 GB in size (5 GiB in binary units)
- 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.

3. Relate the mounted partitions to selections made during installation.
   - Here is the final partition table created during installation.



   a. Use the `df` command to list the mounted partitions. You might see `/dev/sr0` listed in the middle of the output shown, the order of the devices might be different, and some numbers might vary.

```
# df -h
Filesystem      Size  Used  Avail  Use%  Mounted on
...
/dev/xvda2      9.8G  4.0G  5.3G    44% /
/dev/xvda1      976M  168M  742M    19% /boot
/dev/xvda3      2.0G   38M  1.8G     3% /home
...
```

      - The first partition on hard drive xvda (`xvda1`) contains boot files and is mounted on `/boot`.
      - The second partition on hard drive xvda (`xvda2`) is the root file system and is mounted on `/`.
      - The third partition on hard drive xvda (`xvda3`) is for user home directories and is mounted on `/home`.

4. Relate the swap space to selections made during installation. Your output might vary.
   a. Use the `swapon` command to display the swap space.

```
# swapon -s
Filename                        Type            Size    Used    Priority
/dev/xvda5                      partition       3143676 0       -1
```

   - The fifth partition (the first logical partition) on hard drive xvda (`xvda5`) is swap space.

   b. Use the `cat` command to show that the `/proc/swaps` file provides the same information.

```
# cat /proc/swaps
Filename                        Type            Size    Used    Priority
/dev/xvda5                      partition       3143676 0       -1
```

Practice 12-2: 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 that if the file system is greater than 2 TB, you cannot use `fdisk`; you must use `parted`.

Assumptions

You are the `root` user on the **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: unrecognised 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 "`unrecognised 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. Exit the `parted` utility.

```
(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`. Your output might vary.

```
# cat /proc/partitions
major minor  #blocks  name

 202        0   16777216 xvda
 202        1    1048576 xvda1
 202        2   10485760 xvda2
 202        3    2097152 xvda3
 202        4          1 xvda4
 202        5    3143680 xvda5
 202       16    5242880 xvdb
 202       17    1048976 xvdb1
 202       48    5242880 xvdd
 202       49    1048559 xvdd1
  11        0    4336640 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 12-3: Creating `ext3` and `ext4` 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 the **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: done
```

   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/xvdb1: LABEL="Dev"  UUID="..." SEC_TYPE="ext2" TYPE="ext3"
/dev/xvdd1: LABEL="Test" UUID="..." TYPE="ext4" PARTUUID="..."
/dev/xvda1: UUID="..." TYPE="ext4"
/dev/xvda2: UUID="..." TYPE="ext4"
/dev/xvda3: UUID="..." TYPE="ext4"
/dev/xvda5: UUID="..." TYPE="swap"
...
```

      - 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. Your output might vary.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/xvda2       9.8G  4.0G  5.3G  44% /
/dev/xvda1       976M  168M  742M  19% /boot
/dev/xvda3       2.0G   38M  1.8G   3% /home
...
/dev/xvdb1       976M  1.3M  924M   1% /Dev
/dev/xvdd1       992M  2.6M  923M   1% /Test
```

   e. Use the `mount` command to display the mounted file systems. Your output might be ordered differently.

```
# mount
...
/dev/xvda2 on / type ext4 (rw,relatime,seclabel,data=ordered)
...
/dev/xvda1 on /boot type ext4 (rw,relatime,seclabel,data=...)
```

```
/dev/xvda3 on /home 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
sysfs /sys sysfs rw,seclabel,nosuid,nodev,noexec,relatime 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.

a. 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.
- You can also run the `mount -a` command to mount all file systems in `/etc/fstab`.

6. Mount all file systems in `/etc/fstab`.

a. Use the `umount` command to unmount the file systems on `/Dev` and `/Test`.

```
# umount /Dev /Test
```

b. Use the `df` command to display the mounted file systems. Your output might vary.

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
...
/dev/xvda2      9.8G  4.0G  5.3G  44% /
/dev/xvda1      976M  168M  742M  19% /boot
/dev/xvda3      2.0G   38M  1.8G   3% /home
...
```

- Note that `/Dev` and `/Test` are not listed.

c. Use the `mount -a` command to mount all file systems in `/etc/fstab`.

```
# mount -a
```

d. Use the `df` command to display the mounted file systems. Your output might vary.

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
...
/dev/xvda2      9.8G  4.0G  5.3G  44% /
```

```
/dev/xvda1       976M  168M  742M  19% /boot
/dev/xvda3       2.0G   38M  1.8G   3% /home
...
/dev/xvdb1       976M  1.3M  924M   1% /Dev
/dev/xvdd1       992M  2.6M  923M   1% /Test
```

- Note that `/Dev` and `/Test` are now listed.

7. Create files on the new file systems.
   - This task verifies that the new file systems are usable.

   a. Use the `vi` command to create a file on `/Dev`.
      - Name the file whatever you want. This example names the file `dev_file`.

```
# vi /Dev/dev_file
insert any text you want
```

   b. Use the `ls` command to list the contents of `/Dev`.
      - The `lost+found` directory is automatically created when you make an `ext` file system. This directory is used by the `fsck` command to repair a damaged file system. You do not need to manually interact with the `lost+found` directory.

```
# ls -la /Dev
total 28
drwxr-xr-x.  3 root root  4096 <date_time> .
dr-xr-xr-x. 22 root root  4096 <date_time> ..
-rw-r--r--.  1 root root    11 <date_time> dev_file
drwx------.  2 root root 16384 <date_time> lost+found
```

      - Note that `dev_file` exists on `/Dev`.

   c. Use the `cp` command to copy the `/Dev/dev_file` file to `/Test/test_file`.

```
# cp /Dev/dev_file /Test/test_file
```

   d. Use the `ls` command to list the contents of `/Test`.

```
# ls -la /Test
total 28
drwxr-xr-x.  3 root root  4096 <date_time> .
dr-xr-xr-x. 22 root root  4096 <date_time> ..
drwx------.  2 root root 16384 <date_time> lost+found
-rw-r--r--.  1 root root    11 <date_time> test_file
```

      - Note that `test_file` exists on `/Test`.

Practice 12-4: Implementing Access Control Lists

Overview

In this practice, you set ACLs on a directory.

Assumptions

Ensure that you are using `vncviewer` to connect to **host03** and not using `ssh`.

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

You switch between the `root` user and the `oracle` user for this practice.

Tasks

1. Enable ACL support on a file system.
   a. Use the `umount` command to unmount the file system on `/Dev`.

```
# umount /Dev
```

   b. Use the `df` command to display the mounted file systems. Your output might vary.

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
...
/dev/xvda2      9.8G  4.0G  5.3G  44% /
/dev/xvda1      976M  168M  741M  19% /boot
/dev/xvda3      2.0G   38M  1.8G   3% /home
...
/dev/xvdd1      992M  2.6M  923M   1% /Test
```

   - Note that `/Dev` is not listed.

2. Update `/etc/fstab` to add ACL support for the file system mounted on `/Dev`.
   a. Use the `vi` editor to add the `acl` option to the `/Dev` entry in place of `defaults`, as shown.

```
# vi /etc/fstab
LABEL=Dev   /Dev   ext3  defaults  0  0 (old entry)
LABEL=Dev   /Dev   ext3  acl       0  0 (new entry)
...
```

   b. Use the `mount -a` command to mount the file systems in `/etc/fstab`.

```
# mount -a
```

   c. Use the `df` command to display the mounted file systems. Your output might vary.

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
...
/dev/xvda2      9.8G  4.0G  5.3G  44% /
/dev/xvda1      976M  168M  741M  19% /boot
```

```
/dev/xvda3       2.0G   38M  1.8G   3% /home
...
/dev/xvdd1       992M  2.6M  923M   1% /Test
/dev/xvdb1       976M  1.3M  924M   1% /Dev
```

- Note that both `/Dev` and `/Test` are now listed.

d. Use the `mount` command to view the `/dev/xvdb1` partition.

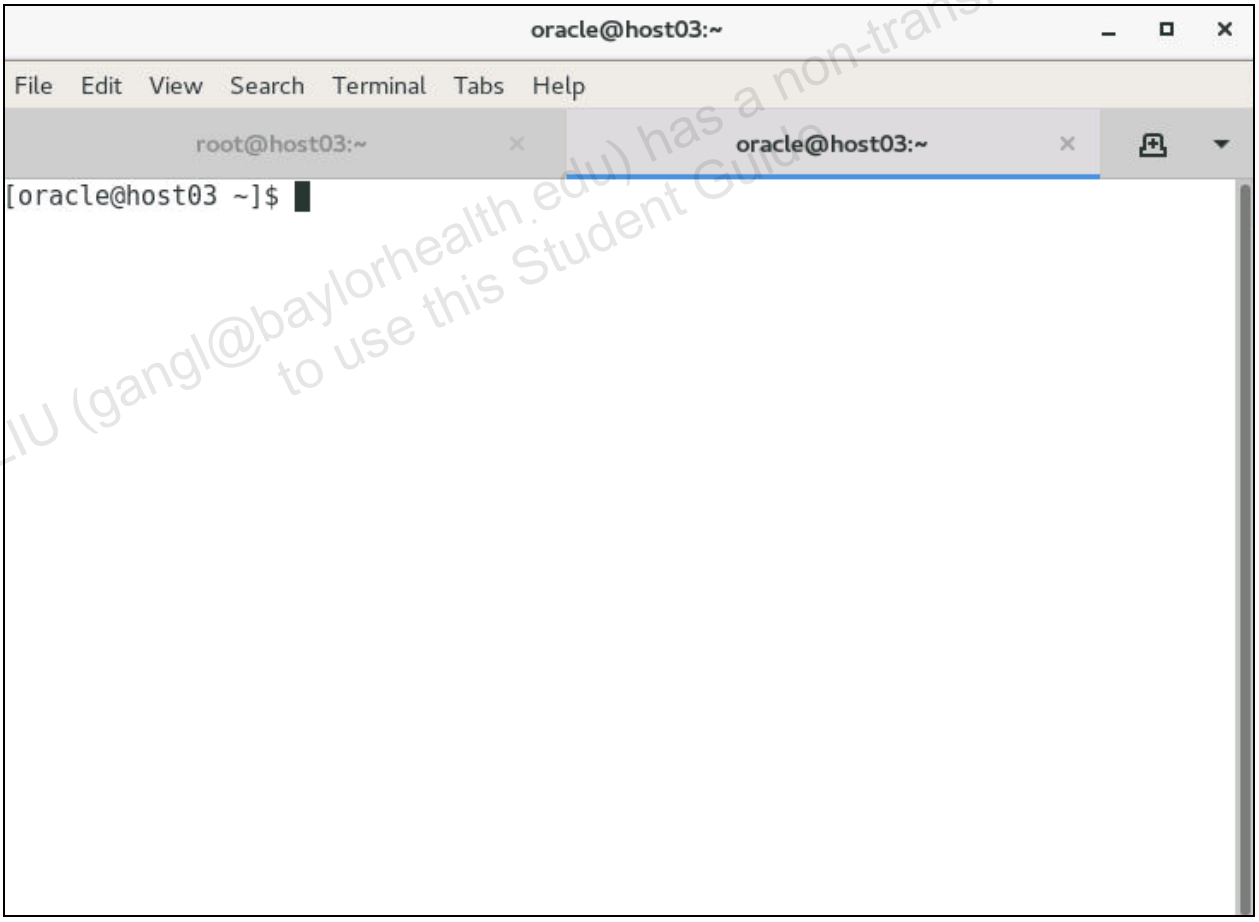
```
# mount | grep /dev/xvdb1
/dev/xvdb1 on /Dev type ext3 (rw,relatime,seclabel,...,acl,...)
```

- Note the `acl` option is in effect.

3. Open a tab in the current window.

a. From the terminal window menu bar, open another tab by selecting File > Open Tab, or pressing Shift + Ctrl + T.

- Your window looks like the following screenshot.
- You are the `root` user in one tab and you are the `oracle` user in the other.



4. As the `oracle` user, use the `touch` command to create the `test` file in the `/Dev` directory.

```
[oracle@host03]$ touch /Dev/test
touch: cannot touch 'Dev/test': Permission denied
```

- Note that you do not have permission to create files in the `/Dev` directory.

5. As the `root` user, use the `getfacl` command to display the `/Dev` directory's ACL.
   - Click the **root@host03** tab to enter commands as the `root` user.

```
[root@host03]# getfacl /Dev
getfacl: Removing leading '/' from absolute path names
# file: Dev
# owner: root
# group: root
user::rwx
group::r-x
other::r-x
```

6. As the `root` user, use the `setfacl` command to add a rule to the ACL giving the `oracle` user read, write, and execute permissions to the `/Dev` directory.

```
[root@host03]# setfacl –m u:oracle:rwx /Dev
```

7. As the `root` user, use the `getfacl` command to display the `/Dev` directory's ACL.

```
[root@host03]# getfacl /Dev
getfacl: Removing leading '/' from absolute path names
# file: Dev
# owner: root
# group: root
user::rwx
user:oracle:rwx
group::r-x
mask::rwx
other::r-x
```

   - Note the new `user:oracle:rwx` line in the output of the `getfacl` command.

8. As the `root` user, use the `ls –ld` command to display the permissions for the `/Dev` directory.

```
[root@host03]# ls -ld /Dev
drwxrwxr-x+ ... /Dev
```

   - Note the plus sign (+), indicating that the directory has an ACL.

9. As the `oracle` user, use the `touch` command to create the `test` file in the `/Dev` directory.
   - Click the **oracle@host03** tab to enter commands as the `oracle` user.

```
[oracle@host03]$ touch /Dev/test
```

   - Note that the command succeeded this time.

10. As the `oracle` user, use the `ls` command to display a long listing of the `/Dev` directory.

```
[oracle@host03]$ ls -l /Dev
-rw-r--r--. 1 root    root ...   dev_file
```

```
drwx------. 2 root    root ...   lost+found
-rw-rw-r--. 1 oracle  oracle ... test
```

- Note that the test file is owned by the oracle user

11. In the terminal window, click the **x** in the **oracle@host03** tab to close this tab.
    - You are now at the **root@host03** prompt in the terminal window.

Practice 12-5: 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 the **host03** VM.

Tasks

1. Display the current amount of swap space. Your output might vary.
   a. Use the `swapon` command to display the current amount of swap space.

```
# swapon -s
Filename                Type            Size    Used  Priority
/dev/xvda5              partition       3143676 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:      3143676 kB
SwapFree:       3143676 kB
```

   c. Use the `free` command to display the amount of swap space.

```
# free
              total        used        free      ...
...
Swap         3143676           0     3143676
```

2. Create and initialize a swap file.
   a. Use the `dd` command to create a 1 GiB swap file, `/swapfile`.
      - Ensure that the `count` argument has six zeros and not seven zeros, to avoid filling up the `root` partition.

```
# dd if=/dev/zero of=/swapfile bs=1024 count=1000000
1000000+0 records in
1000000+0 records out
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 permissions on the file.

   b. Display the updated amount of swap space. Your output might vary.

```
# swapon -s
Filename                Type            Size    Used  Priority
/dev/xvda5              partition       3143676 7328  -1
/swapfile               file            999996  0     -2

# grep -i swap /proc/meminfo
SwapCached:           44 kB
SwapTotal:      4143672 kB
SwapFree:       4136344 kB

# free
              total        used        free      ...
...
Swap:       4143672        7324     4136348
```

Practice 12-6: Removing Partitions and Additional Swap Space

Overview

In this practice, you delete the partitions on the `/dev/xvdb` and `/dev/xvdd` devices, and remove the additional swap file in preparation for the next lesson.

Assumptions

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

Tasks

1. Use the `df` command to display the mounted file systems. Your output might vary.

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
...
/dev/xvda2      9.8G  5.0G  4.3G  54% /
/dev/xvda1      976M  168M  742M  19% /boot
/dev/xvda3      2.0G   38M  1.8G   3% /home
...
/dev/xvdd1      992M  2.6M  923M   1% /Test
/dev/xvdb1      976M  1.3M  924M   1% /Dev
```

   - Note that `/dev/xvdd1` is mounted on `/Test` and `/dev/xvdb1` is mounted on `/Dev`.

2. Use the `umount` command to unmount `/Dev` and `/Test`. Use the `cd` command first, to ensure you are in the `root` user's home directory.
   - You cannot delete a partition when a file system is mounted on it.

```
# cd
# umount /Dev /Test
```

3. Use the `fdisk` command to delete the `/dev/xvdb1` partition.
   - Use the `d` command to delete partition1.
   - Use the `p` command to print the partition table and confirm there are no partitions.
   - Use the `w` command to save the partition table and exit the `fdisk` utility.

```
# fdisk /dev/xvdb
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): p
```

```
Disk /dev/xvdb: 5368 MB, 5368709120 bytes ...
...
    Device Boot      Start         End      Blocks   Id  System

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

4. Use the `fdisk` command to delete the `/dev/xvdd1` partition.
   - Use the `d` command to delete partition 1.
   - Use the `p` command to print the partition table and confirm there are no partitions.
   - Use the `w` command to save the partition table and exit the `fdisk` utility.

```
# fdisk /dev/xvdd
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): p

Disk /dev/xvdd: 5368 MB, 5368709120 bytes ...
...
Disk label type: gpt
...
#         Start          End    Size  Type            Name

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

5. Use `parted` to change the label on `/dev/xvdd` from `gpt` to `msdos`
   - Use the `mklabel` command to create an `msdos` label.
   - Use the `print` command to show the partition table to confirm the label change.
   - Use the `quit` command to exit the `parted` utility.

```
# parted /dev/xvdd
GNU Parted 3.1
Using /dev/xvdd
Welcome to GNU Parted! Type 'help' to view a list of commands.
```

```
(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) print
Model: Xen Virtual Block Device (xvd)
Disk /dev/xvdd: 5369MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start  End  Size  Type  File system  Flags

(parted) quit
Information: You may need to update /etc/fstab.
```

6. View the `xvd` devices in the `/proc/partitions` file to ensure that the `xvdb1` and `xvdd1` entries no longer exist.

```
# grep xvd /proc/partitions
 202        0   16777216 xvda
 202        1    1048576 xvda1
 202        2   10485760 xvda2
 202        3    2097152 xvda3
 202        4          1 xvda4
 202        5    3143680 xvda5
 202       16    5242880 xvdb
 202       48    5242880 xvdd
```

7. Use the `vi` command to remove the `Dev` and `Test` entries from the `/etc/fstab` file.

```
# vi /etc/fstab
LABEL=Dev       /Dev    ext3    acl             0 0
LABEL=Test      /Test   ext4    defaults        0 0
```

8. Use the `rmdir` command to remove the `/Dev` and `/Test` mount points.

```
# rmdir /Dev /Test
```

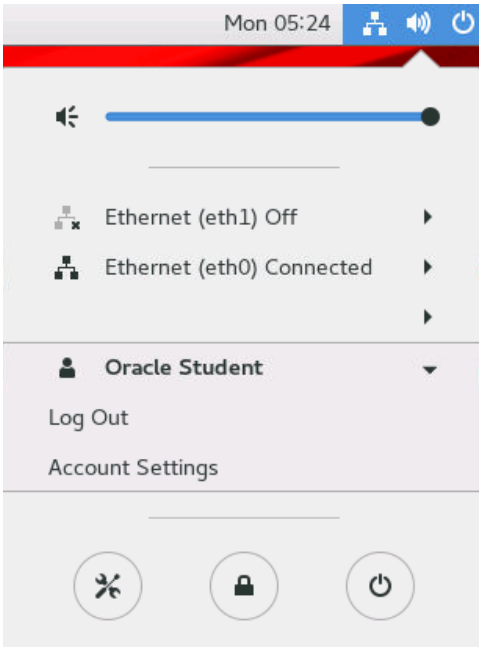
9. Use the `swapoff` command to disable swapping to /`swapfile`.

```
# swapoff /swapfile
```

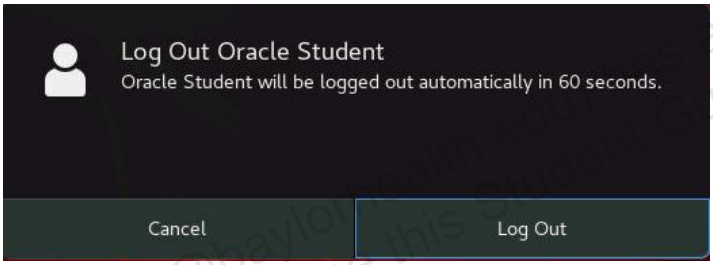
10. Use the `rm` command to remove /`swapfile`.

```
# rm /swapfile
rm: remove regular file '/swapfile'? y
```

11. Log off from **host03** in preparation for the next practice.
    a. Click the Power icon in the upper right of the GNOME screen > select Oracle Student.



1) Click Log Out. The following pop-up appears:



2) Click Log Out.
3) Click the **X** in the top-right corner of the GNOME login window to close the window.

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

Practices for Lesson 13: Network Configuration

Practices for Lesson 13: Overview

Practices Overview

In these practices, you:

- Configure the `eth1` network interface by editing network interface configuration files
- Use NetworkManager with the GNOME GUI to configure network interfaces
- Use the Network Connection editor to configure networking properties
- Use the `nmcli` utility to configure networking properties
- Use the `nmtui` text-based utility to configure network interfaces
- Use the `ip` utility to manage network links, addresses, and the ARP cache

Practice 13-1: Configuring the `eth1` Network Interface

Overview

In this practice, you:

- Display the available network interfaces on your system
- View the network interface configuration files
- Configure a static IP address for the `eth1` network interface
- Update your `/etc/hosts` file
- Display your route table
- Ensure connectivity to **dom0** and the other VM guests

Assumptions

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

Tasks

1. Log in to **host03**.
   a. Connect to the **host03** guest by using the `xm vncviewer host03&` command.

```
# xm vncviewer host03&
```

   The GNOME login window appears.
   b. Select Oracle Student from the GNOME login window; enter the password.
   c. Right-click the GNOME desktop and select **Open Terminal** from the shortcut menu.
   d. In the terminal window, become the `root` user by entering the `su -` command and providing the `root` password.

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

2. Use the `ip addr` command to display your available network interfaces.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.103/24 brd 192.0.2.255 scope global ...
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
...
```

- Note that you have two Ethernet interfaces (`eth0` and `eth1`) and the loopback interface (`lo`).
- The `eth0` Ethernet interface has an IP address, but `eth1` does not.

3. View the network interface configuration files.
   a. Use the `cd` command to change to the /`etc/sysconfig/network-scripts` directory.

```
# cd /etc/sysconfig/network-scripts
```

   b. Use the `ls` command to view the contents of this directory.

```
# ls
ifcfg-eth0  ifdown-post    ifup-bnep  ifup-routes
ifcfg-eth1  ifdown-ppp     ifup-eth   ifup-sit
ifcfg-lo    ifdown-routes  ifup-ippp  ifup-Team
...
```

   - Note that you have a configuration file for `eth0`, `ifcfg-eth0`.
   - Observe that you have a configuration file for `eth1`, `ifcfg-eth1`.
   - Note that you have a configuration file for the loopback interface, `ifcfg-lo`.
   - Several interface control scripts exist in this directory to activate and deactivate network interfaces.

   c. Use the `cat` command to view the contents of the `ifcfg-eth0` file.

```
# cat ifcfg-eth0
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=eth0
UUID=...
DEVICE=eth0
ONBOOT=yes
IPADDR=192.0.2.103
PREFIX=24
GATEWAY=192.0.2.1
IPV6_PRIVACY=no
```

   - Note that this Ethernet interface is configured with a static IPv4 address:
     - `BOOTPROTO=none`

- IPADDR=192.0.2.103

- Some network interface configuration file parameters are described in https://docs.oracle.com/cd/E52668_01/E54669/html/ol7-about-netconf.html.

4. Configure eth1 with a static IP address of 192.168.1.103.

- If you were unable to complete the OS install on **host03** in the Installation practices, and you have been using **host02** in the practices, use the IP address of 192.168.1.102 for the eth1 interface if configuring **host02**.
- If you are using **host03** for the practices, use the IP address of 192.168.1.103 for the eth1 interface if configuring **host03**.

a. Make a copy of the original ifcfg-eth1 file.

```
# cp ifcfg-eth1 ifcfg-eth1_SAV
```

b. Use the vi editor to edit the ifcfg-eth1 file as follows: (The required changes are listed in the following bullets and are in **bold** font.)

- Change BOOTPROTO=dhcp to **BOOTPROTO=none**
- Change DEFROUTE=yes to **DEFROUTE=no**
- Change ONBOOT=no to **ONBOOT=yes**
- Add **IPADDR=192.168.1.103**
- Add **PREFIX=24**
- Add **PEERDNS=no**
- Add **PEERROUTES=no**

```
# vi ifcfg-eth1
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=no
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=eth1
UUID=...
DEVICE=eth1
ONBOOT=yes
IPADDR=192.168.1.103
PREFIX=24
PEERDNS=no
```

```
PEERROUTES=no
```

5. Make a copy of the original `/etc/hosts` file.

```
# cp /etc/hosts /etc/hosts_SAV
```

6. Use the `vi` editor to edit the `/etc/hosts` file to look as follows:

```
# vi /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
```

7. Use the `systemctl` command to restart the network service.

```
# systemctl restart network
```

8. Use the `ip addr` command to display the status of the interfaces.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.103/24 brd 192.0.2.255 scope global ...
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
...
```

- Note that both `eth0` and `eth1` now have IP addresses.

9. Use network interface control scripts to stop and start a specific network interface.

   a. Use the `ifdown` script to stop the `eth1` interface.

```
# ifdown eth1
Device 'eth1' successfully disconnected.
```

   b. Use the `ip addr` command to display the status of the interfaces.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
...
```

- Note that the `eth1` interface does not have IP addresses.

   c. Use the `ifup` script to start the `eth1` interface.

```
# ifup eth1
Connection successfully activated (D-Bus active path: ...)
```

d. Use the `ip addr` command to display the status of the interfaces.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
...
```

- Note that the `eth1` interface now has IP addresses.

10. Display the route table.

a. Use the `netstat -r` command (or `route`) to display the route table.

```
# netstat -r
Kernel IP routing table
Destination   Gateway      Genmask        Flags ...  Iface
default       example.com  0.0.0.0        UG         eth0
192.0.2.0     0.0.0.0      255.255.255.0  U          eth0
192.168.1.0   0.0.0.0      255.255.255.0  U          eth1
192.168.122.0 0.0.0.0      255.255.255.0  U          vibr0
```

- Note that all packets destined for the `192.168.1` subnet use the `eth1` interface.
- Observe that all packets destined for the `192.0.2` subnet use the `eth0` interface.
- Note that all other packets are routed through the `192.0.2.1` default gateway (`example.com`), via `eth0`.

b. You can also use the `ip route` command to display the route table in a different format.

```
# ip route
default via 192.0.2.1 dev eth0 proto static metric 100
192.0.2.0/24 dev eth0 proto kernel ... 192.0.2.103 metric 100
192.168.1.0/24 dev eth1 proto ... 192.168.1.103 metric 101
192.168.122.0/24 dev virbr0 proto kernel ... 192.168.122.1
```

11. Use the `ping` command to verify that you can communicate to **dom0** and the other VM guests.

- Press **Ctrl + C** to kill the `ping` command.

```
# ping dom0
64 bytes from example.com (192.0.2.1)...
CTRL-C
# ping host01
64 bytes from host01.example.com (192.0.2.101)...
CTRL-C
# ping host02
```

```
64 bytes from host02.example.com (192.0.2.102)...
CTRL-C
```

Practice 13-2: Using NetworkManager with the GNOME GUI

Overview

In this practice, you:

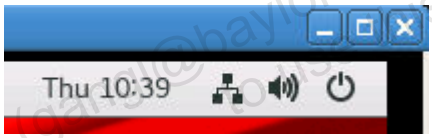
- Ensure that the NetworkManager software package is installed
- Use the NetworkManager GUI to view network status
- Use the Network Settings window to:
  - Disable and enable a network connection
  - View network configuration settings
  - Add a connection profile
  - Select a different connection profile
- View the network interface configuration file for the new connection profile

Assumptions

- You are connected to the **host03** VM by using vncviewer.
- You are the `root` user on the **host03** VM.

Tasks

1. Install and start NetworkManager if necessary.
   - NetworkManager is installed and running if you see the network icon in the upper right of your GNOME window as follows:



   a. Use the `rpm` command to verify that the NetworkManager package is installed.

```
# rpm -qa | grep -i networkmanager
...
NetworkManager-config-server--...
NetworkManager-ppp-...
...
NetworkManager-...
```

   - In this example, NetworkManager is installed.

   b. If NetworkManager is not installed, use the `yum` command to install the package.

```
# yum install NetworkManager
...
```

   c. Use the `systemctl` command to verify that NetworkManager is running.

```
# systemctl status NetworkManager
NetworkManager.service - Network Manager
   Loaded: loaded (/usr/lib/systemd/system/NetworkManager...)
   Active: active (running) since ...
```

```
. . .
```

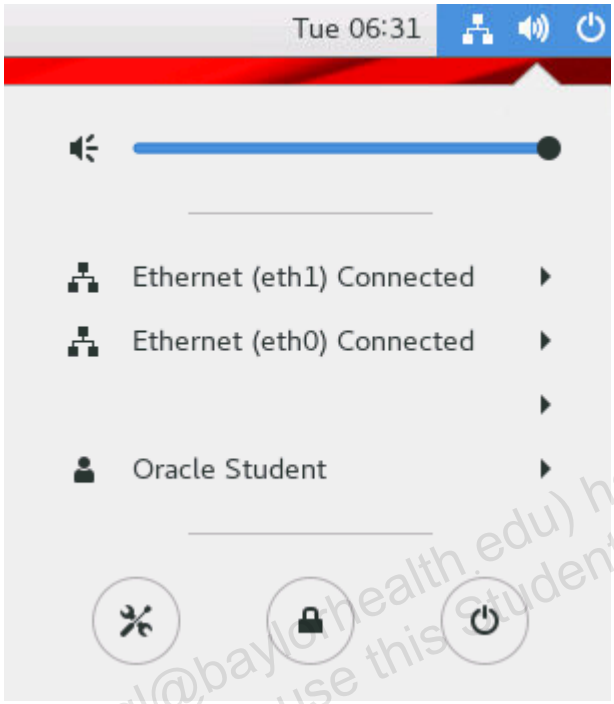
- In this example, NetworkManager is running.

d. If NetworkManager is not running, use the `systemctl` command to start it.

```
# systemctl start NetworkManager
```

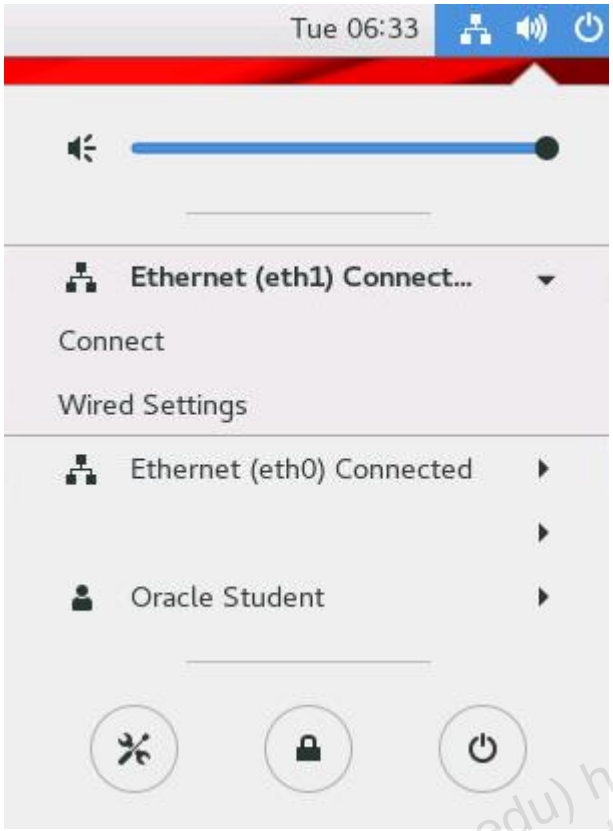
2. Use NetworkManager to view network status and disable and enable a network interface.

a. Select the network icon to display the following:



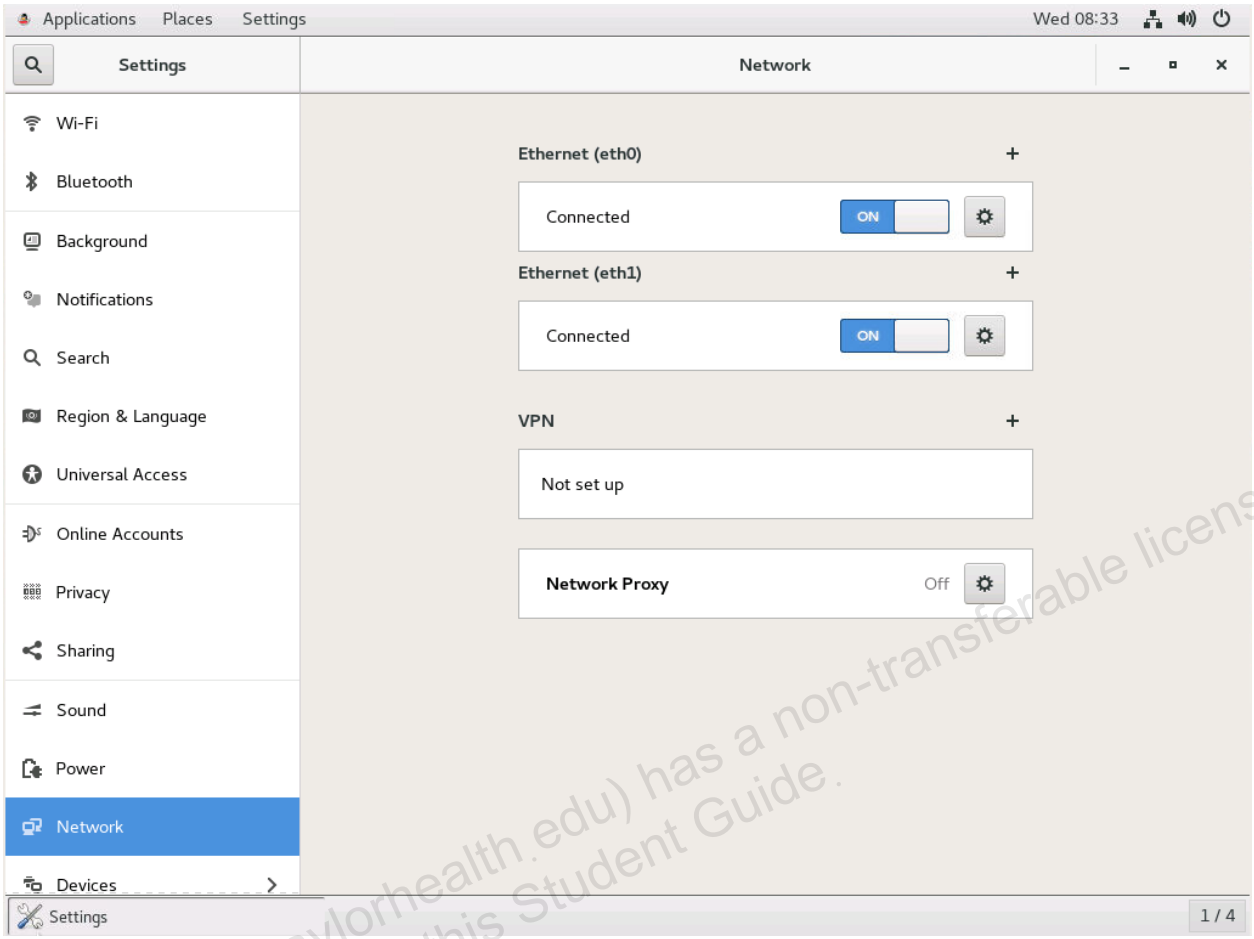
- In this example, there are two Ethernet interfaces.
- Both network interfaces are **Connected**.

b. Select the **Ethernet (eth1)** entry.



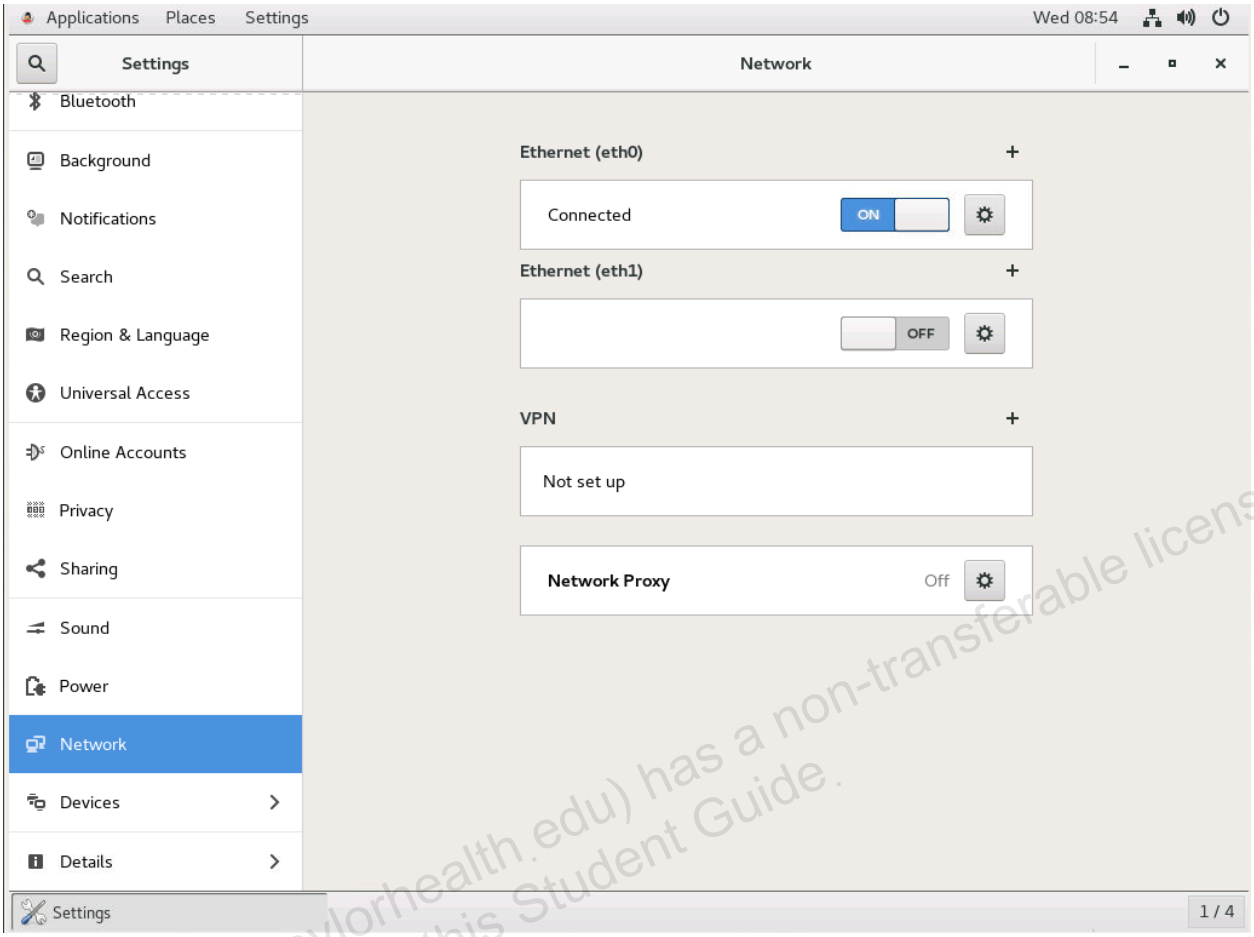
- Note that you can connect an interface from this screen or select **Wired Settings**.

c. Select **Wired Settings** to display the following screen. Network settings are part of the GNOME Settings interface.



d. For the **Ethernet (eth1)** entry, toggle the **ON/OFF** switch to **OFF**.

- The window is displayed as shown.



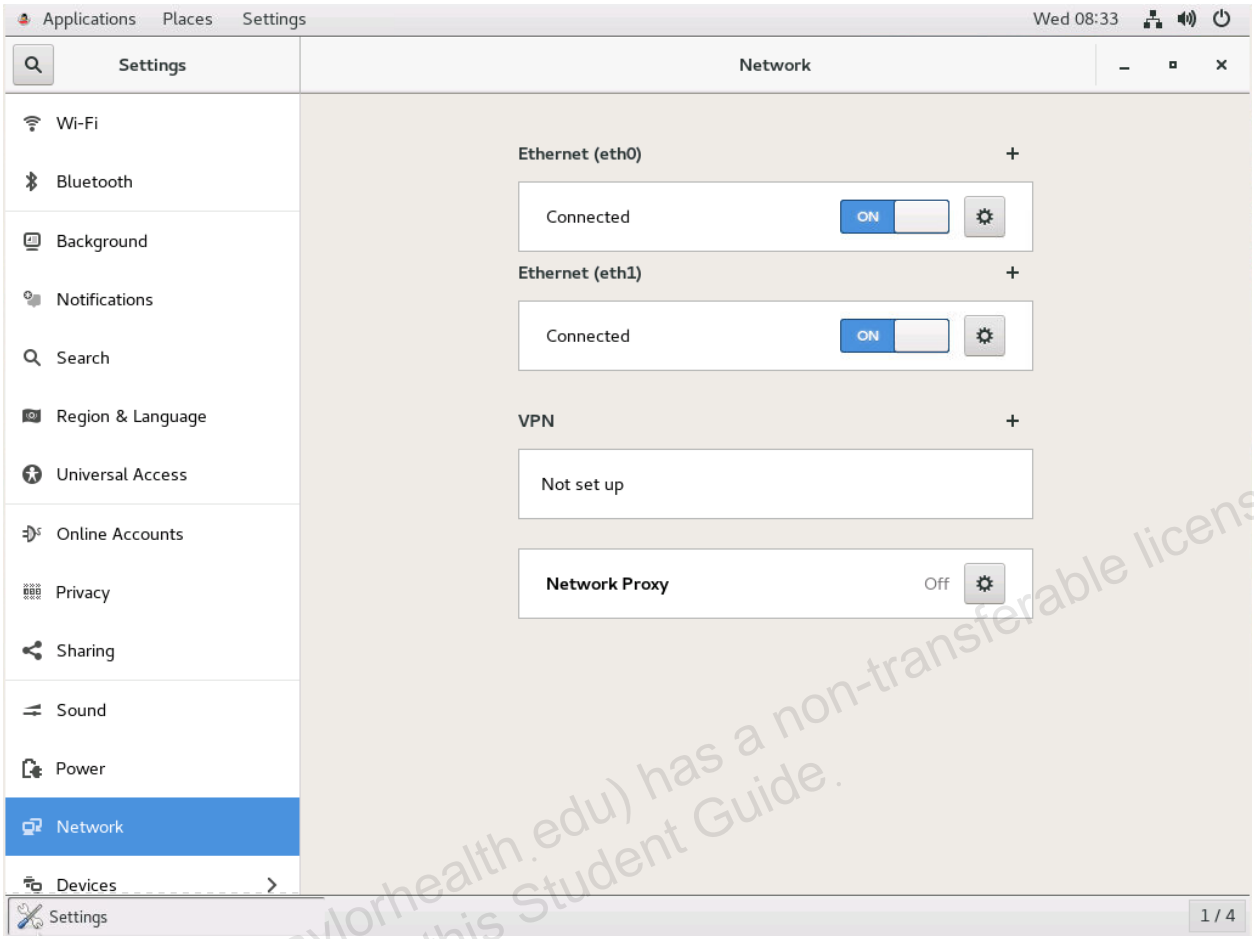
e. From a terminal window, use the `ip addr` command to display the status of the interfaces.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
...
```

- Note that the `eth1` interface does not have IP addresses.

f. For the **Ethernet (eth1)** entry, toggle the **ON/OFF** switch to **ON**.

- The window is displayed as shown.



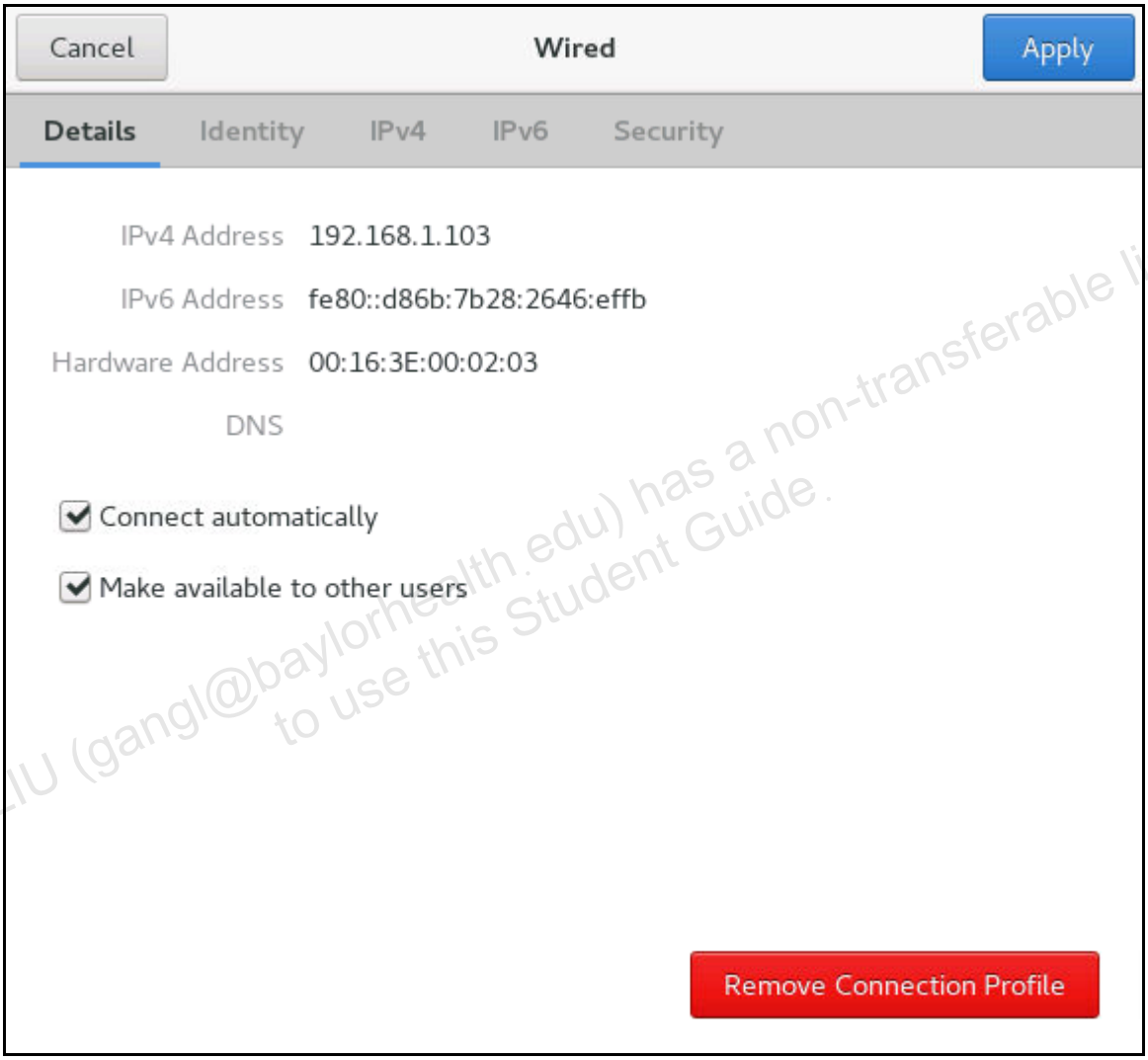
g. From a terminal window, use the `ip addr` command to display the status of the interfaces.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
...
```

- Note that the `eth1` interface now has IP addresses.

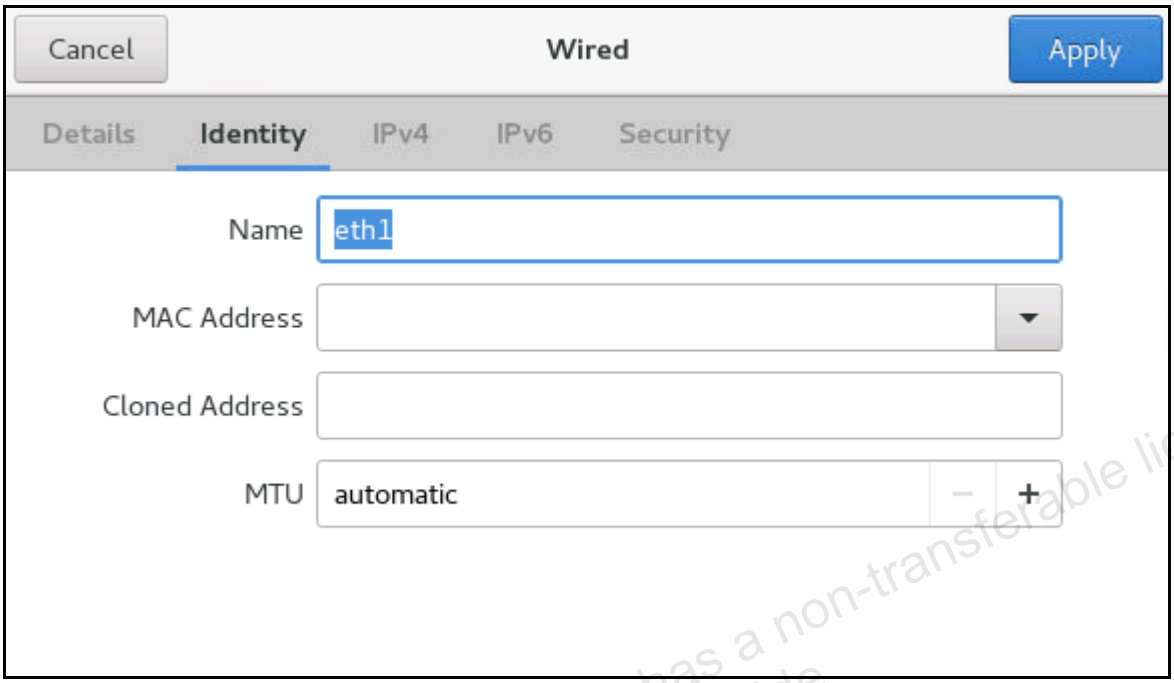
3. Use the **Network** window to view network configuration settings.
   a. For the **Ethernet (eth1)** entry, select the gear icon on the right side of the entry as shown:

- The window is displayed as shown:



b. Select the **Identity** entry along the top.

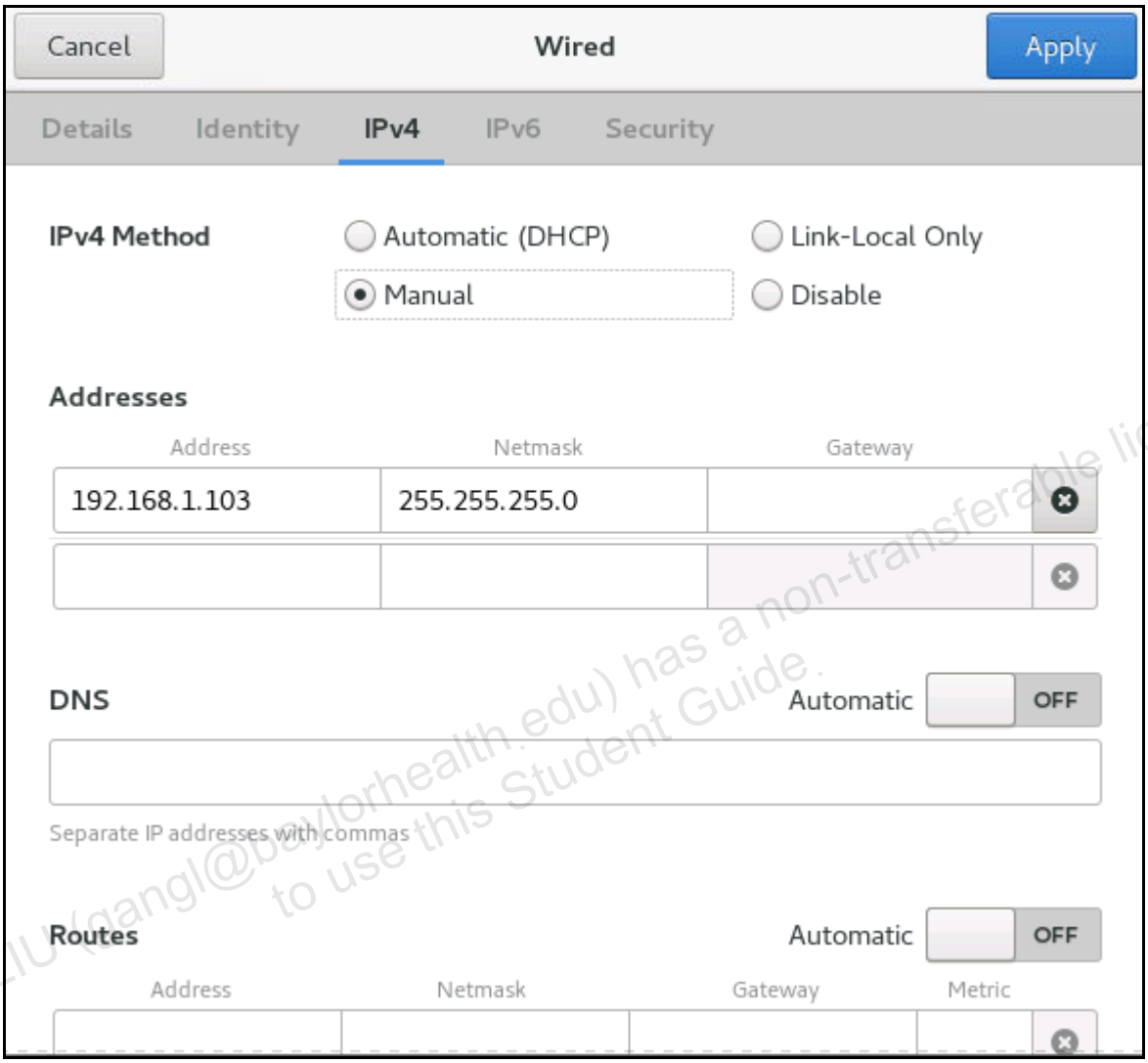
- The window is displayed as shown:



- Note the available settings in this window.

c. Select the **IPv4** entry along the top.

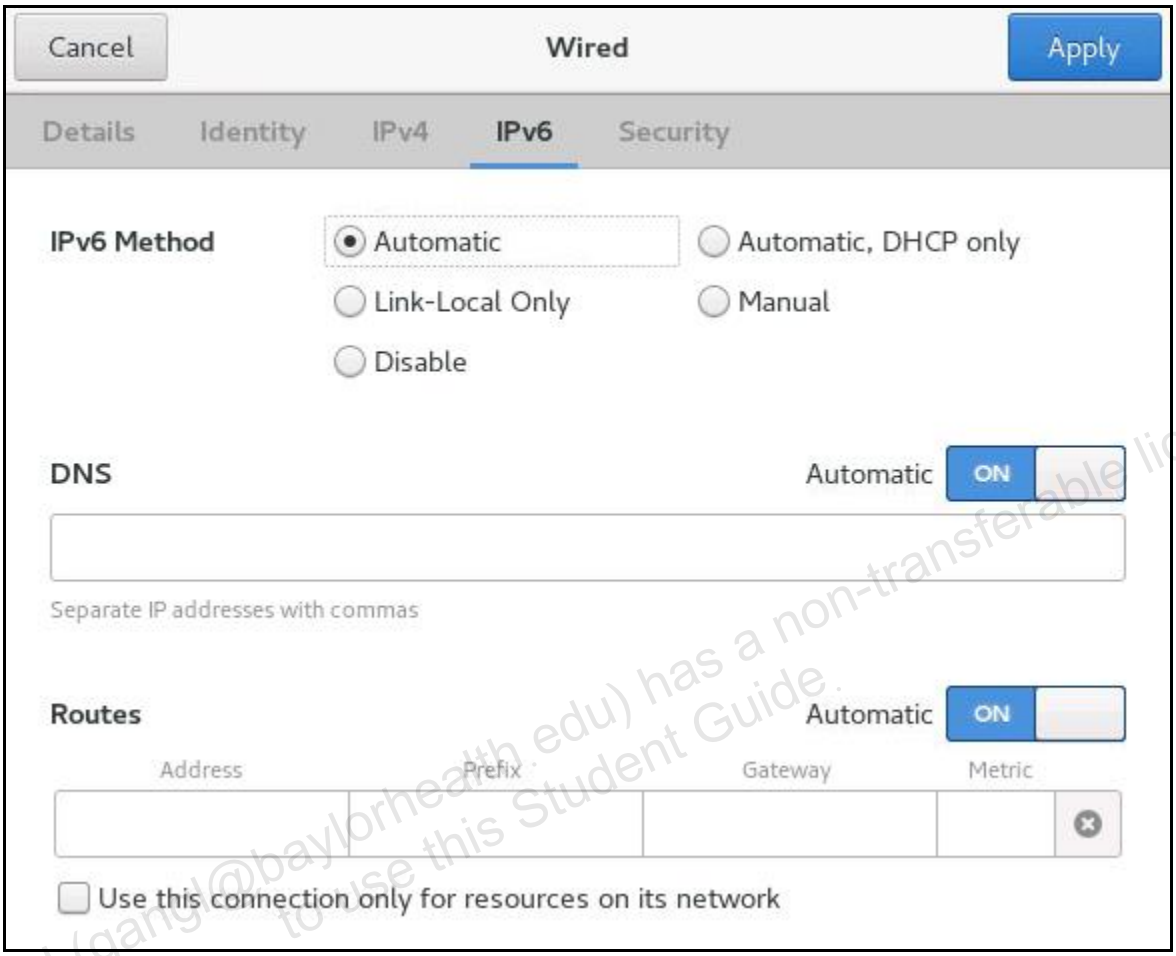
- The window is displayed as shown:



- Observe the available settings in this window.
- Scroll down to view all settings.

d. Select the **IPv6** entry along the top.

- The window is displayed as shown:



- Note the available settings in this window.

e. Select the **Security** entry along the top.

- The window is displayed as shown:

f. Toggle the **ON/OFF** switch to **ON** to enable 802.1x Security.

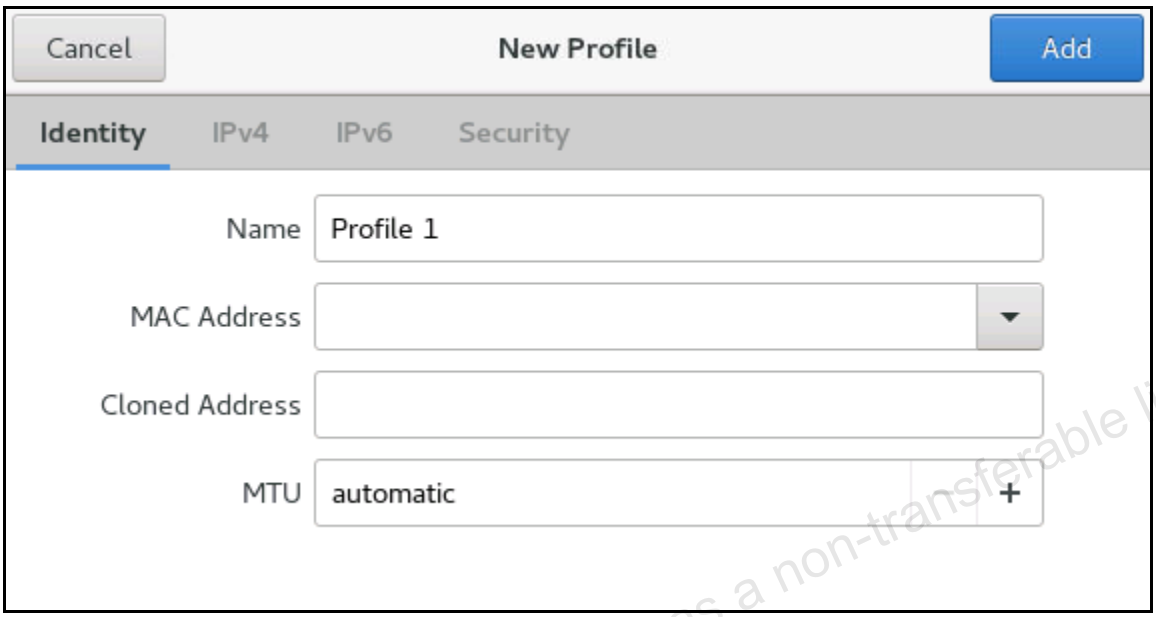
g. Select each Authentication option and view the configuration setting options:

- **MD5** (message-digest algorithm)
- **TLS** (Transport Layer Security)
- **PWD** (Shared Password)
- **FAST** (Flexible Authentication via Secure Tunneling)
- **Tunneled TLS**
- **Protected EAP (PEAP)** (Protected Extensible Authentication Protocol)

h. Toggle the **ON/OFF** switch to **OFF** to disable 802.1x Security.

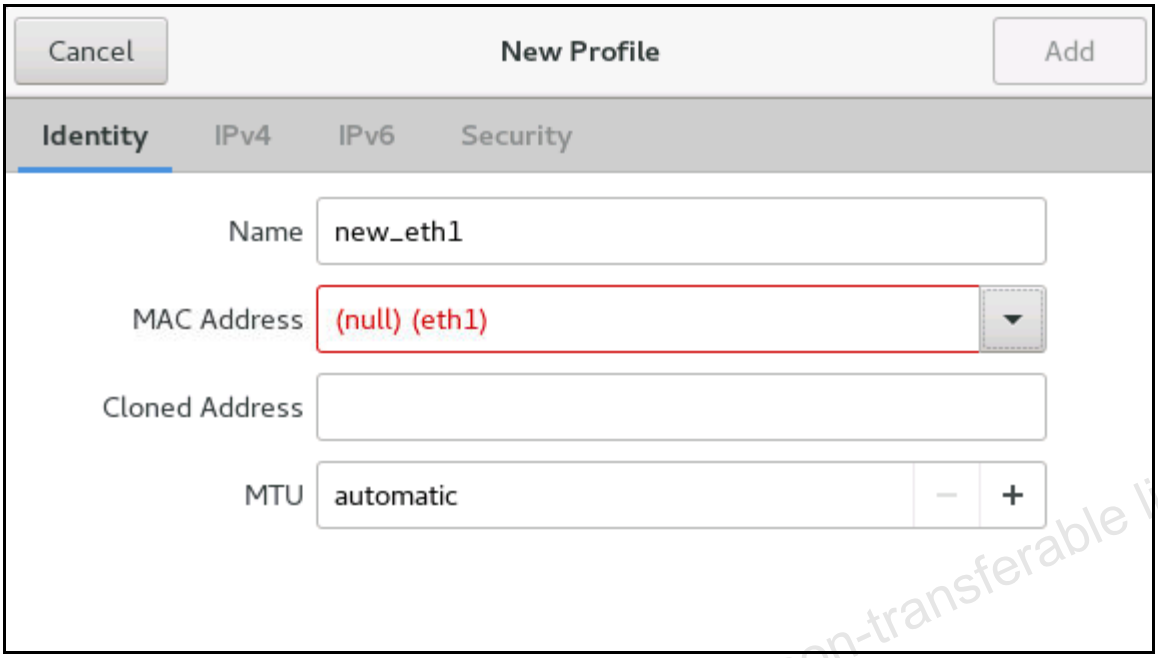
i. Select **Cancel** to close the window and return to the main **Network Settings** window.

4. Use the **Network** window to add a profile for the `eth1` interface.
   a. Click the "+" to the right of **Ethernet (eth1).**
      - The window is displayed as shown:



   b. Provide the following Identity information:
      - **Name**: `new_eth1`
      - **MAC Address**: Select `eth1` from the drop-down list.
      - Leave the MTU setting as **automatic**.

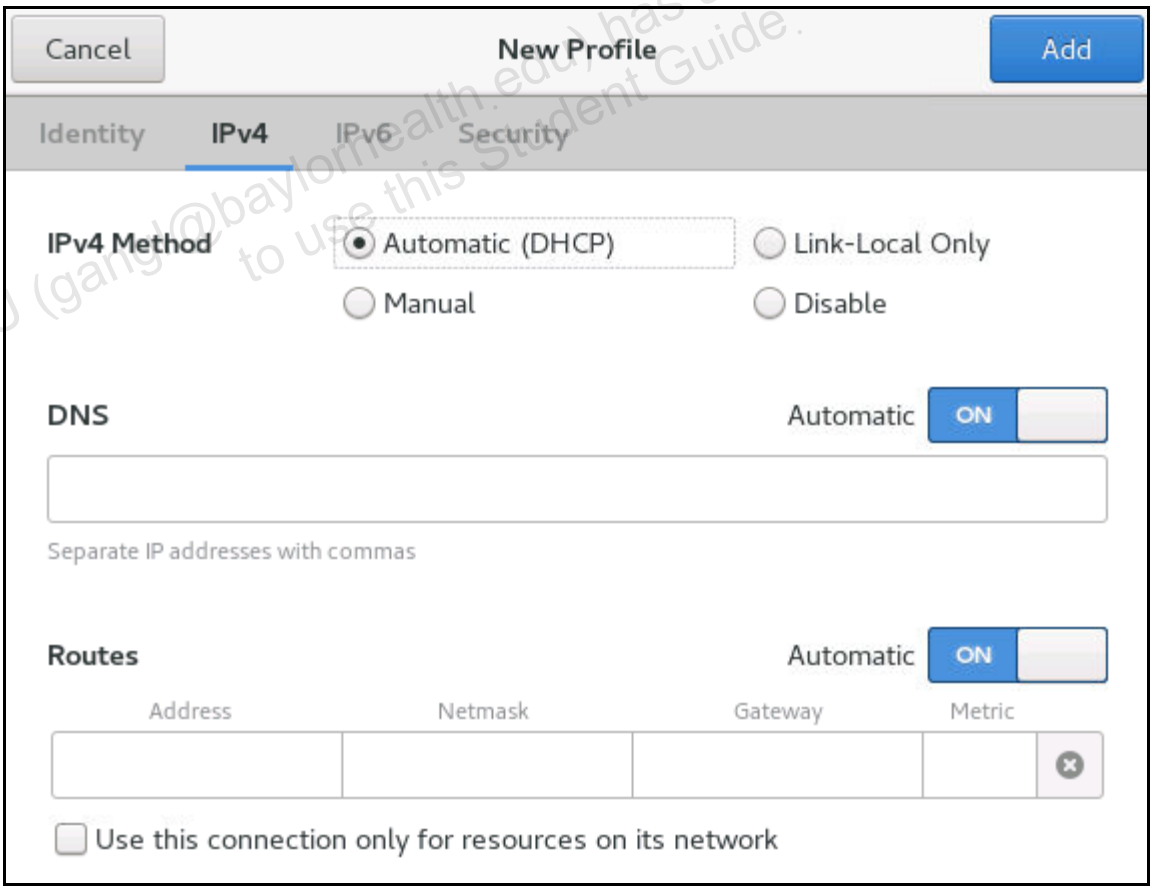
- Your window appears as shown:



c. Enter the following MAC address for host03 - **`00:16:3E:00:02:03`** - replacing the "(null)" characters so that your entry looks like this:

d. Select the **IPv4** entry along the top.

- The window is displayed as shown:



- Do not make any changes in this window.
- Use DHCP to obtain an IP address from the DHCP server, **dom0**.

- Dynamic Host Configuration Protocol (DHCP) is covered in another course.

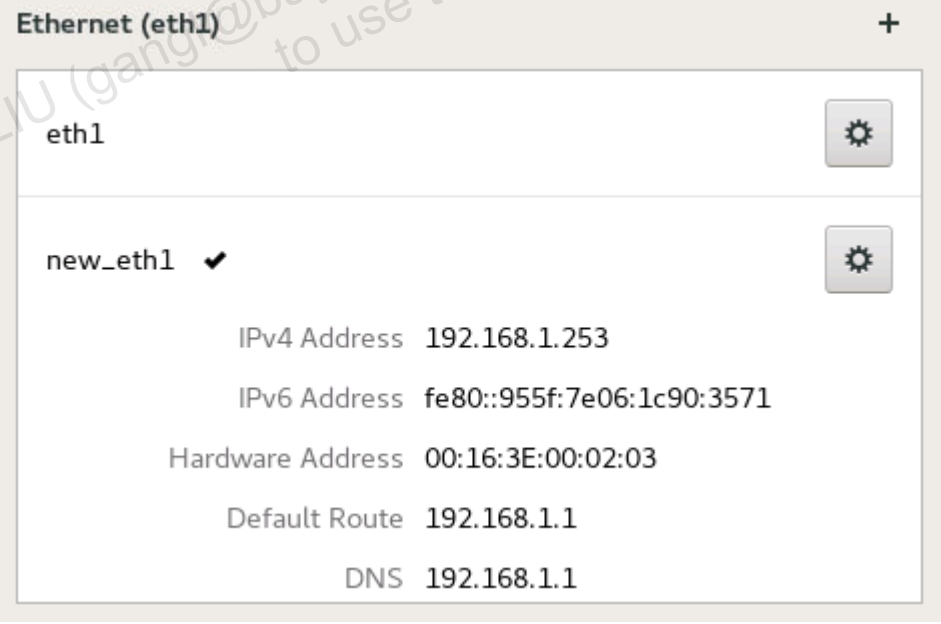
e. Click **Add**.

- The **New Profile** window closes and the **Network** window is displayed.
- The **Ethernet (eth1)** entry should look like this:

- Note that `eth1` has a check mark, meaning it is currently the selected profile.

5. Use NetworkManager to select a different connection profile.

a. Select `new_eth1` from the **Network** window. The **Ethernet (eth1)** entry should now look like the following:



b. From a terminal window, use the `ip addr` command to display the status of the interfaces.

```
# ip addr
...
```

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.253/24 brd 192.168.1.255 scope global ...
...
```

- Note that the `eth1` interface now has a different IPv4 address.
- In this example, the IPv4 address is `192.168.1.253`.
- This address might be different on your system because it was obtained from the DHCP server running on **dom0**.

6. View the network interface configuration file for the new connection profile.
   a. From a terminal window, use the `cd` command to change to the `/etc/sysconfig/network-scripts` directory.

```
# cd /etc/sysconfig/network-scripts
```

   b. Use the `ls` command to display files beginning with "`ifcfg`".

```
# ls ifcfg*
ifcfg-eth0  ifcfg-eth1  ifcfg-eth1_SAV ifcfg-lo  ifcfg-new_eth1
```

   - Note that the `ifcfg-new_eth1` file exists for the new connection profile.

   c. Use the `cat` command to view the contents of the `ifcfg-new_eth1` file.

```
# cat ifcfg-new_eth1
HWADDR=00:16:3E:00:02:03
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=new_eth1
UUID=...
ONBOOT=yes
```

   - Note that this Ethernet interface uses DHCP to obtain an IPv4 address:
     - `BOOTPROTO=dhcp`

7. Use NetworkManager to select the `eth1` connection profile.
   a. Select `eth1` from the **Network** window. The **Ethernet (eth1)** entry should now look like the following:

b. From a terminal window, use the `ip addr` command to display the status of the interfaces.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
...
```

- Note that the `eth1` interface IPv4 address is now set to `192.168.1.103`.

8. Close the GNOME **Settings** window.
   a. Click the **X** in the upper-right corner of the **Settings** window.

Practice 13-3: Using the Network Connection Editor

Overview

In this practice, you:

- Ensure that the Network Connections package is installed
- Run the Network Connection editor and view configuration settings for `eth0`

Assumptions

- You are connected to the **host03** VM by using vncviewer.
- You are the `root` user on the **host03** VM.

Tasks

1. Install the Network Connections package if necessary.
   - The Network Connections package name is `nm-connection-editor`.

   a. Use the `rpm` command to verify that the `nm-connection-editor` package is installed.

```
# rpm -q nm-connection-editor
nm-connection-editor-...
```

   - In this example, the package is installed.

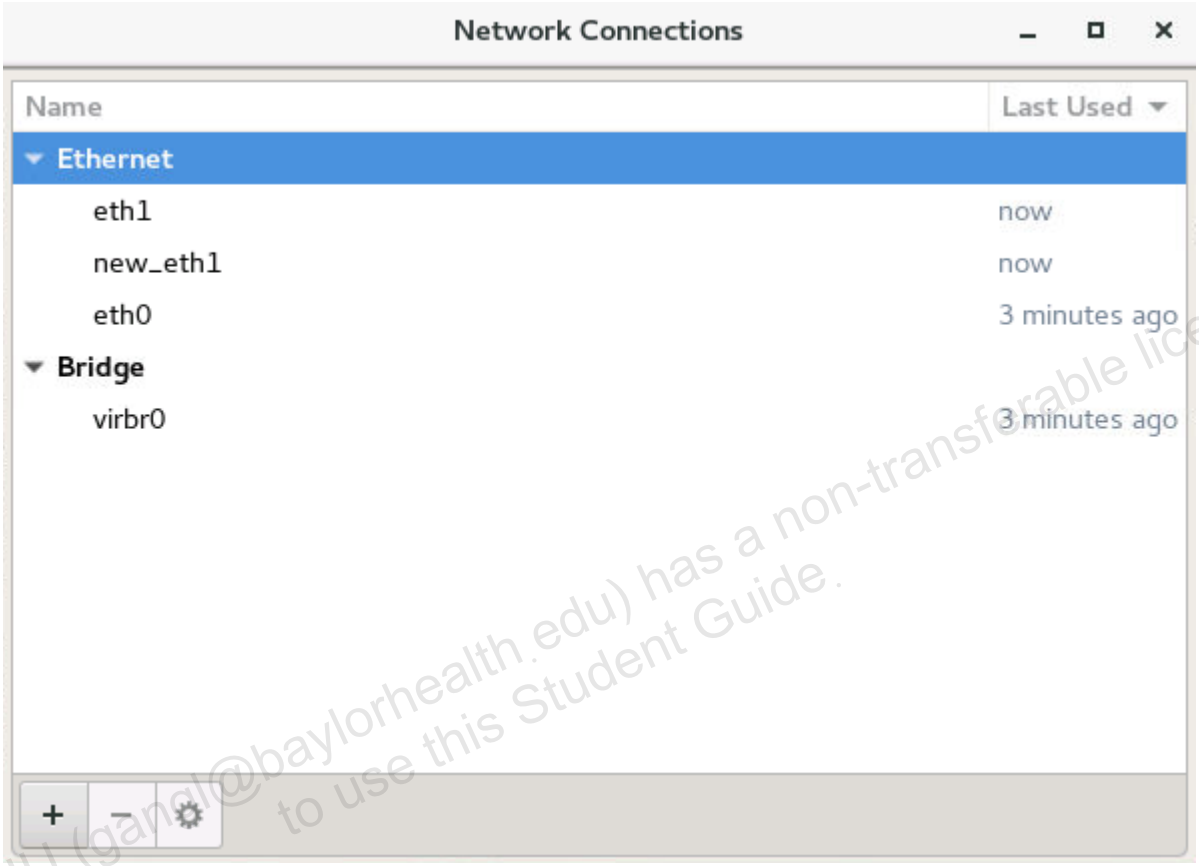
   b. If the Network Connections package is not installed, use the `yum` command to install it.

```
# yum install nm-connection-editor
...
```

2. Run the Network Connection editor.
   a. Run the `nm-connection-editor` command.

```
# nm-connection-editor
```

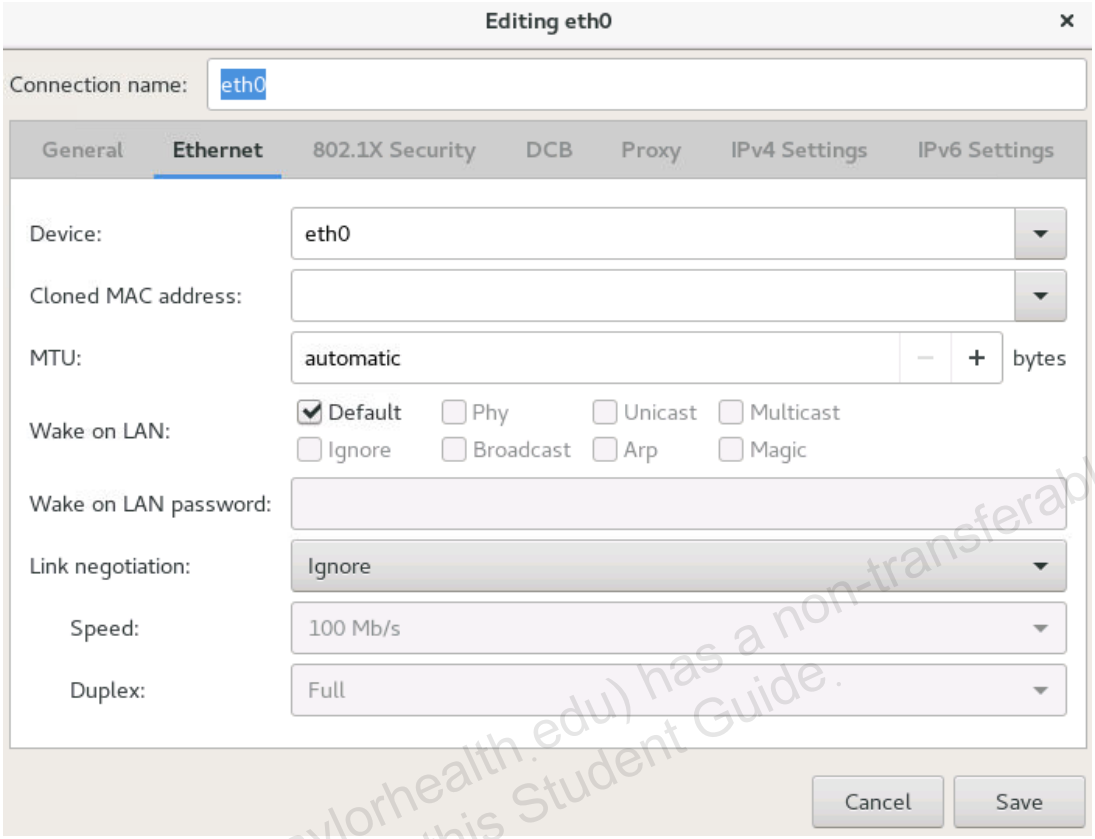
   - The window is displayed as shown:



   - Note the three Ethernet connections, including the `new_eth1` connection profile.
   - Observe that the icons on the lower left allow you to add new connection (+), delete a connection (−), or edit an existing connection (gear icon).

b. Select the `eth0` connection and then select the gear icon to edit it.

- The window appears as shown:



- Note that this is the same window that appeared when you configured the network during the installation of Oracle Linux 7.

c. Select the remaining tabs to view the configuration options for each window.

- **General**
- **802.1x Security**
- **DCB**
- **Proxy**
- **IPv4 Settings**
- **IPv6 Settings**

d. Select **Cancel** to close the editing window.

e. Click the **X** in the upper-right corner to close the Network Connections editor window.

Practice 13-4: Using the `nmcli` Utility

Overview

In this practice, you use the `nmcli` command-line utility to view network status information, change the system host name, change the logging level, disable and enable networking, view connection information, add, edit and delete a connection profile, and view device status information.

Assumptions

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

Tasks

1. Run the `nmcli help` command.

```
# nmcli help
Usage: nmlci [OPTIONS] OBJECT { COMMAND | help }
OPTIONS
  -t[erse]                    terse output
  -p[retty]                   pretty output
  -m[ode] tabular|multiline   output mode
...
OBJECT
  g[eneral]     NetworkManager's general status and operations
  n[etworking]  overall networking control
  r[adio]       NetworkManager radio switches
  c[onnection]  NetworkManager's connections
  d[evice]      devices managed by NetworkManager
  a[gent]       NetworkManager secret agent or polkit agent
  m[onitor]     monitor NetworkManager changes
```

   - Note that several options are available:
     - `-t|--terse`: This mode is designed and suitable for script processing.
     - `-p|--pretty`: This mode produces easily readable output with a header.
     - `-m|--mode tabular|multiline`: Produces output in table format or on multiple lines.
   - Refer to the `nmcli(1)` man page for a description of all options.
   - Note that there are seven different objects for the `nmcli` command.
2. Run the `nmcli general` object commands.
   a. Run the `nmcli general help` command.

```
# nmcli general help
Usage: nmcli general { COMMAND | help }
```

```
COMMAND := { status | hostname | permissions | logging }
...
```

- Note that the `nmcli general` object provides four commands.

b. Run the `nmcli general status` command.

- Note that "`status`" is the default. That is, you can omit this argument.

```
# nmcli general status
STATE      CONNECTIVITY  WIFI-HW  WIFI     WWAN-HW  WWAN
connected  full          enabled  enabled  enabled  enabled
```

- Note that the network status is "`connected`" with "`full`" connectivity.
- Full connectivity means the host is connected to a network and has full access to the Internet

c. Use the `systemctl` command to stop the NetworkManager service.

```
# systemctl stop NetworkManager
```

d. Run the `nmcli general status` command.

```
# nmcli general status
Error: NetworkManager is not running.
```

- With the NetworkManager service stopped, an error is returned.

e. Use the `systemctl` command to start the NetworkManager service.

```
# systemctl start NetworkManager
```

f. Run the `nmcli general hostname` command.

```
# nmcli general hostname
host03.example.com
```

- This command reports the host name when the NetworkManager service is running.
- You can also use this `nmcli general hostname` command to change the host name.

g. Run the `nmcli general hostname` command and change the host name to "`myhost`".

```
# nmcli general hostname myhost
```

- The host name is stored in the `/etc/hostname` file.

h. Use the `cat` command to view the contents of the `/etc/hostname` file.

```
# cat /etc/hostname
myhost
```

i. Run the `nmcli general hostname` command and change the host name back to "`host03.example.com`".

```
# nmcli general hostname host03.example.com
```

j. Use the `cat` command to view the contents of the `/etc/hostname` file.

```
# cat /etc/hostname
```

```
host03.example.com
```

k. Run the `nmcli general permissions` command.

- In this example, all permissions are set to `yes`, which means that you can enable and disable networking and modify all connections and settings.

```
# nmcli general permissions
PERMISSION                                                VALUE
org.freedesktop.NetworkManager.enable-disable-network     yes
org.freedesktop.NetworkManager.enable-disable-wifi        yes
org.freedesktop.NetworkManager.enable-disable-wwan        yes
org.freedesktop.NetworkManager.enable-disable-wimax       yes
org.freedesktop.NetworkManager.sleep-wake                 yes
org.freedesktop.NetworkManager.network-control            yes
org.freedesktop.NetworkManager.wifi.share.protected       yes
org.freedesktop.NetworkManager.wifi.share.open            yes
org.freedesktop.NetworkManager.settings.modify.system     yes
org.freedesktop.NetworkManager.settings.modify.own        yes
org.freedesktop.NetworkManager.settings.modify.hostname yes
...
```

l. Run the `nmcli general logging` command.

- With no arguments, this command shows the current logging level by domain.

```
# nmcli general logging
LEVEL  DOMAINS

INFO   PLATFORM,RFKILL,ETHER,WIFI,BT,MB,DHCP4,DHCP6,PPP,IP4,IP6,
AUTOIP4,DNS,VPN,SHARING,SUPPLICANT,AGENTS,SETTINGS,SUSPEND,CORE,
DEVICE,OLPC,INFINIBAND,FIREWALL,ADSL,BOND,VLAN,BRIDGE,TEAM,
CONCHECK,DCB,DISPATCH,AUDIT,SYSTEMD,PROXY
```

- In this example, the logging level is `INFO` for all domains.
- Refer to the `NetworkManager.conf(5)` man page for information about logging levels and domain descriptions.

m. Use the `nmcli general logging` command to change the logging level to `DEBUG` for the `IP4` domain.

```
# nmcli general logging level DEBUG domains IP4
```

n. Run the `nmcli general logging` command to show the current logging level.

```
# nmcli general logging
LEVEL  DOMAINS
DEBUG  IP4
```

o. Use the `nmcli general logging` command to change the logging level to `INFO` for the `ALL` domains.

- This command returns the logging level to the default setting.

```
# nmcli general logging level INFO domains ALL
```

3. Run the `nmcli networking` object commands.

a. Run the `nmcli networking help` command.

```
# nmcli networking help
Usage: nmcli networking { COMMAND | help }

COMMAND := { [ on | off | connectivity ] }
...
```

- Note that the `nmcli networking` object provides three commands.

b. Run the `nmcli networking` command with no options or arguments to show the networking status.

```
# nmcli networking
enabled
```

- Note that the status is `enabled`.

c. Run the `nmcli networking off` command to disable networking.

```
# nmcli networking off
```

d. Run the `nmcli networking` command with no options or arguments to show the networking status.

```
# nmcli networking
disabled
```

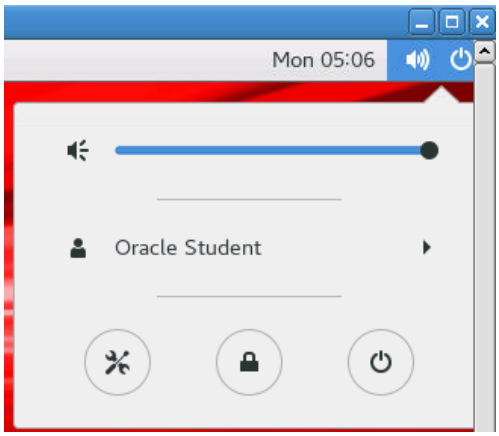
- Note that the status is `disabled`.

e. Use the `ip addr` command to display your available network interfaces.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
...
2: eth0: <BROADCAST,MULTICAST> mtu 1500 qdisc ... DOWN ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc ... DOWN ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
...
```

- Note that both Ethernet interfaces are **DOWN** and have no IP addresses.

f. Select the power icon on the GNOME notification area to show that networking is disabled because no network interfaces are shown.



g. Run the `nmcli networking on` command to enable networking.

```
# nmcli networking on
```

h. Run the `nmcli networking` command with no options or arguments to show the networking status.

```
# nmcli networking
enabled
```

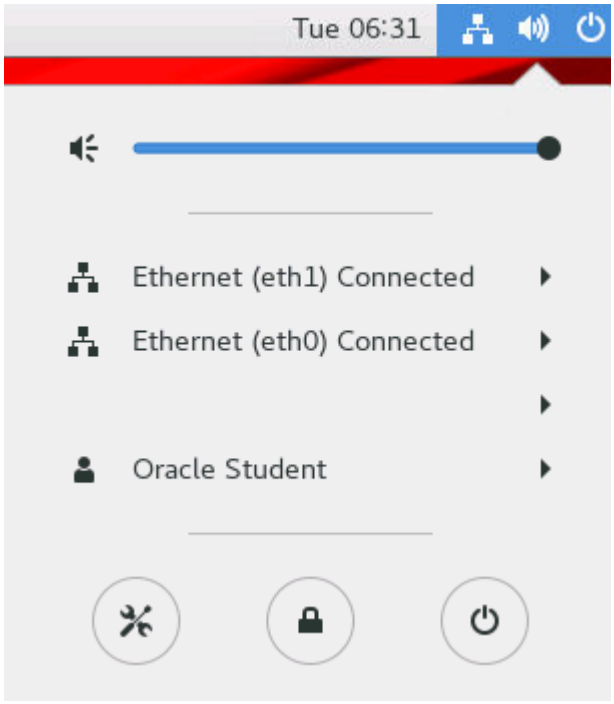
- Note that the status is now `enabled`.

i. Use the `ip addr` command to display the status of the interfaces.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.103/24 brd 192.0.2.255 scope global ...
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
...
```

- Note that both `eth0` and `eth1` are now **UP** and have IP addresses.

j. Select the network icon on the GNOME notification area to show that the network interfaces are connected:



k. Run the `nmcli networking connectivity` command to show the network connectivity state.

- Include the `check` argument.
- Without the `check` argument, the command displays the most recent known connectivity state without re-checking.

```
# nmcli networking connectivity check
full
```

- Connectivity state `full` means the host is connected to a network and has full access to the Internet.

4. Run the `nmcli radio` object commands.

a. Run the `nmcli radio help` command.

```
# nmcli radio help
Usage: nmcli radio { COMMAND | help }

COMMAND := { [ all | wifi | wwan | }
...
```

b. Run the `nmcli radio` command with no options or arguments to show radio switches status.

```
# nmcli radio
WIFI-HW  WIFI     WWAN-HW  WWAN
enabled  enabled  enabled  enabled
```

- Note that all switches are `enabled`.

c. Run the `nmcli radio wifi off` command to disable the WI-FI radio switch.

```
# nmcli radio wifi off
```

d. Run the `nmcli radio` command to show radio switches' status.

```
# nmcli radio
WIFI-HW  WIFI      WWAN-HW  WWAN
enabled  disabled  enabled  enabled
```

- Note that the WI-FI switch is `disabled`.

e. Run the `nmcli radio wifi on` command to enable the WI-FI radio switch.

- Run the `nmcli radio` command to show radio switches' status.

```
# nmcli radio wifi on
# nmcli radio
WIFI-HW  WIFI     WWAN-HW  WWAN
enabled  enabled  enabled  enabled
```

- Note that all switches are `enabled`.

5. Run the `nmcli connection` object commands.

a. Run the `nmcli connection help` command.

```
# nmcli connection help
Usage: nmcli connection { COMMAND | help }

COMMAND := { show | up | down | add | modify | clone | edit |
delete | monitor | reload | load | import | export }
...
```

- Note that the `nmcli connection` object provides 13 commands.

b. Run the `nmcli connection show` command.

- This command lists all the connection profiles.

```
# nmcli connection show
NAME         UUID    TYPE              DEVICE
eth0         ...     ethernet          eth0
eth1         ...     ethernet          eth1
virbr0       ...     bridge            virbr0
new_eth1     ...     ethernet          --
```

c. Run the `nmcli connection show` command with the `--active` argument.

- This command lists only active profiles.

```
# nmcli connection show --active
NAME         UUID    TYPE              DEVICE
eth0         ...     ethernet          eth0
eth1         ...     ethernet          eth1
virbr0       ...     bridge            virbr0
```

d. Run the `nmcli connection show id eth0` command.

- This command shows detailed information for a specific connection, `eth0`.

- Only a partial output is shown.

```
# nmcli connection show id eth0
connection.id:                  eth0
connection.uuid:                ...
...
connection.type:                802-3-ethernet
connection.interface-name:      eth0
connection.autoconnect:         yes
...
802-3-ethernet.mtu              auto
...
ipv4.method:                    manual
...
ipv4.addresses:                 192.0.2.103/24
ipv4.gateway:                   192.0.2.1
...
```

e. Run the `nmcli connection up id new_eth1` command.

- This command activates a specific connection, `new_eth1`.

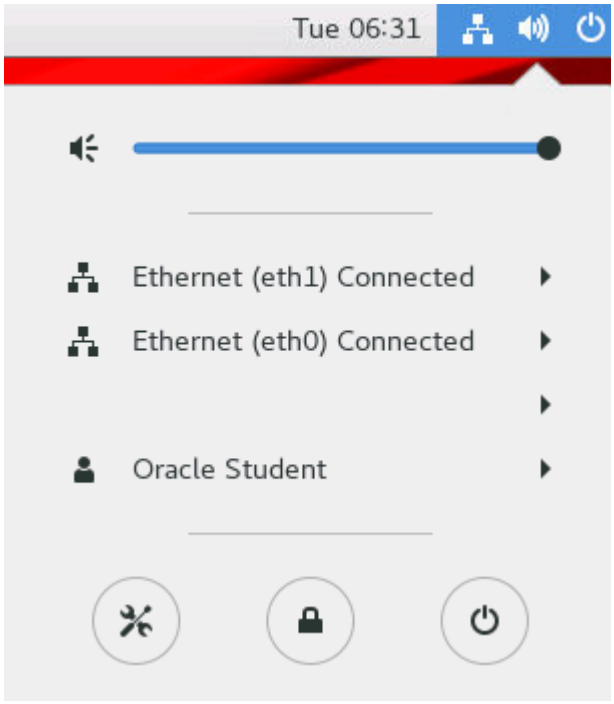
```
# nmcli connection up id new_eth1
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/6)
```

f. Run the `nmcli connection show` command.

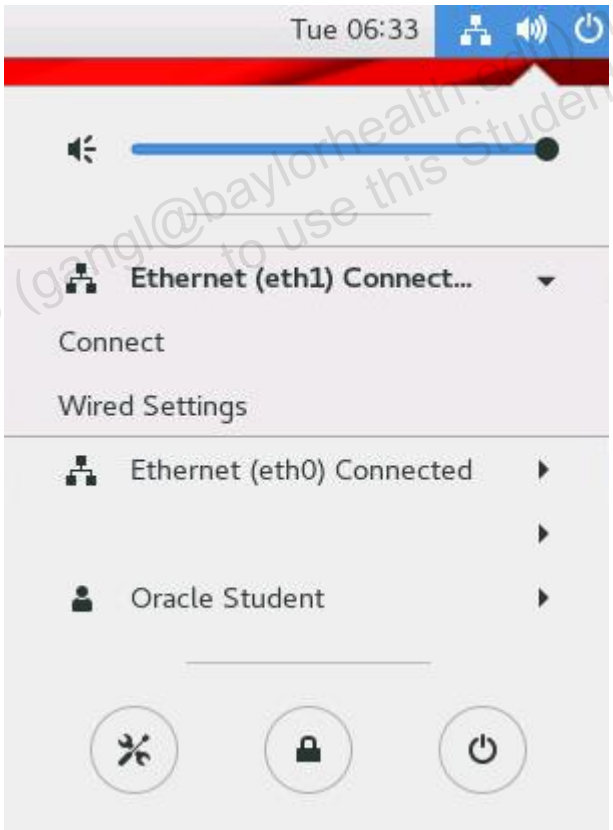
```
# nmcli connection show
NAME        UUID    TYPE              DEVICE
eth0        ...     ethernet          eth0
new_eth1    ...     ethernet          eth1
virbr0      ...     bridge            virbr0
eth1        ...     ethernet          --
```

- Note that the `new_eth1` connection profile is now active and `eth1` is not.

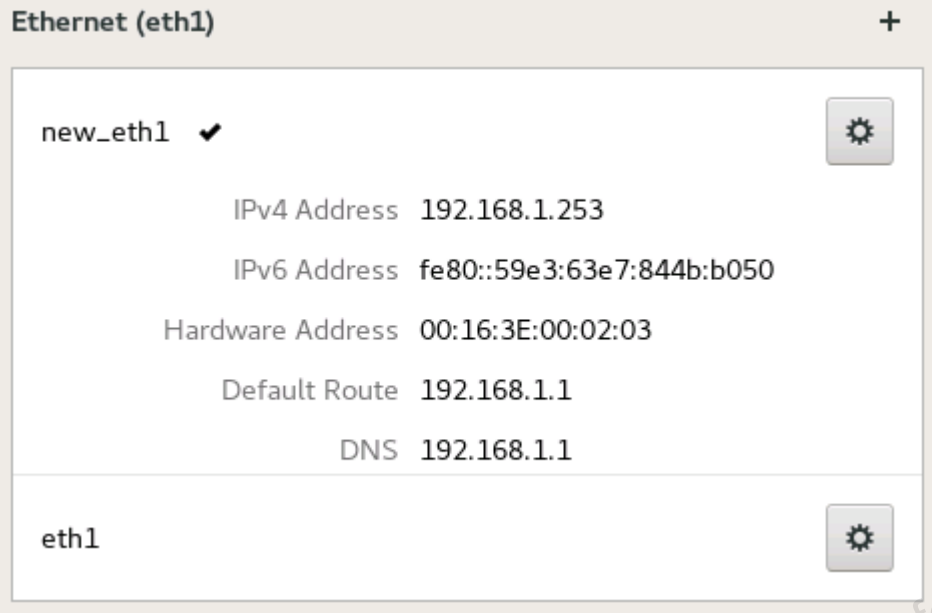
g. Select the network icon in the GNOME notification area to display the following:



h. Select **`Ethernet (eth1) Connected`** to display the `Wired Settings` option:



i. Click the **`Wired Settings`** option to see that `new_eth1` has a check mark next to it, indicating it is currently selected.



j. Click the **X** in the upper-right corner of the **Settings** window to close it.

k. Run the `nmcli connection down id new_eth1` command.

- This command deactivates a specific connection, `new_eth1`.

```
# nmcli connection down id new_eth1
Connection 'new_eth1' successfully deactivated (D-Bus active
path: /org/freedesktop/NetworkManager/ActiveConnection/6)
```

l. Run the `nmcli connection show` command.

```
# nmcli connection show
NAME        UUID    TYPE              DEVICE
eth0        ...     ethernet          eth0
eth1        ...     ethernet          eth1
virbr0      ...     bridge            virbr0
new_eth1    ...     ethernet          --
```

- Note that the `new_eth1` connection profile is no longer active and `eth1` is active.
- The `eth1` connection automatically starts because the `autoconnect` parameter is set to "`yes`".

m. Run the `nmcli connection show id eth1` command.

- Pipe the output to `grep` and search for the "`autoconnect`" string.

```
# nmcli connection show id eth1 | grep autoconnect
connection.autoconnect:      yes
...
```

- Note that the `autoconnect` parameter is set to "`yes`".

6. Use the `nmcli connection` object to add, edit, and delete a connection profile.

a. Run the `nmcli connection add` command to add a new connection profile. Use the following parameters:

- Connection name (`con-name`): `new_eth0`
- Interface name (`ifname`): `eth0`
- Type (`type`): `ethernet`
- IPv4 address (`ip4`): `192.0.2.111/24`
- IPv4 Gateway (`gw4`): `192.0.2.1`

```
# nmcli connection add con-name new_eth0 ifname eth0 type
ethernet ip4 192.0.2.111/24 gw4 192.0.2.1
Connection 'new_eth0' (...) successfully added.
```

b. Run the `nmcli connection show` command.

```
# nmcli connection show
NAME        UUID    TYPE              DEVICE
eth0        ...     ethernet          eth0
eth1        ...     ethernet          eth1
virbr0      ...     bridge            virbr0
new_eth0    ...     ethernet          --
new_eth1    ...     ethernet          --
```

- Note that the `new_eth0` connection now exists.

c. Use the `ls` command to view network interface configuration files in the `/etc/sysconfig/network-scripts` directory.

```
# ls /etc/sysconfig/network-scripts/ifcfg*
/etc/sysconfig/network-scripts/ifcfg-eth0
/etc/sysconfig/network-scripts/ifcfg-eth1
/etc/sysconfig/network-scripts/ifcfg-eth1_SAV
/etc/sysconfig/network-scripts/ifcfg-lo
/etc/sysconfig/network-scripts/ifcfg-new_eth0
/etc/sysconfig/network-scripts/ifcfg-new_eth1
```

- Note that the `nmcli connection add` command created a network interface configuration file, `ifcfg-new_eth0`, for the new connection profile, `new_eth0`.

d. Run the `nmcli connection edit` command to edit connection parameters for the `new_eth0` connection.

- This command uses an interactive editor.

```
# nmcli connection edit new_eth0

===| nmcli interactive connection editor |===

Editing existing '802-3-ethernet' connection: 'new_eth0'

Type 'help' or '?' for available commands.
Type 'describe [<settings>.<prop>]' for detailed property
description.
```

```
You may edit the following settings: connection, 802-3-ethernet
(ethernet), 802-1x, dcb, ipv4, ipv6, tc, proxy
nmcli>
```

e. Enter `help` at the `nmcli>` prompt to list the available commands.

- Only partial output is shown.

```
nmcli> help
----------------------------------------------------------
---[ Main menu ]---
goto     [<setting> | <prop>]         :: go to a setting ...
remove   <setting>[.<prop>] | <prop>  :: remove setting or ...
set      [<setting>.<prop> <value>]   :: set property value
describe [<setting>.<prop>]           :: describe property
print    [all | <setting>[.<prop>]]   :: print the connection
...
```

f. Enter `print` at the `nmcli>` prompt to print the connection profile details.

- Only partial output is shown.
- Note that the output is similar to the "`nmcli connection show id new_eth0`" command output.

```
nmcli> print
==========================================================
          Connection profile details (new_eth0)
==========================================================
connection.id:                new_eth0
connection.uuid:              ...
...
connection.type:              802-3-ethernet
connection.interface-name:    eth0
connection.autoconnect:       yes
...
----------------------------------------------------------
...
802-3-ethernet.mac-address:   --
...
802-3-ethernet.mtu            auto
...
----------------------------------------------------------
ipv4.method:                  manual
ipv4.dns:                     --
ipv4.dns-search:              --
...
```

```
ipv4.addresses:                 192.0.2.111/24
ipv4.gateway:                   192.0.2.1
...
```

- Note that the `connection.autoconnect` parameter is set to "`yes`".

g. Use the `set` command from the `nmcli>` prompt to change the following parameter:

- `connection.autoconnect=no`

```
nmcli> set connection.autoconnect no
```

h. Use the `quit` command from the `nmcli>` prompt to exit the interactive editor.

- Answer **`n`** when prompted. Do not exit the `nmcli` interactive editor.

```
nmcli> quit
The connection is not saved. Do you really want to quit?
(yes/no) [no] n
```

i. Use the `save` command from the `nmcli>` prompt to save your change.

```
nmcli> save
Connection 'new_eth0' (...) successfully updated.
```

j. Use the `quit` command from the `nmcli>` prompt to exit the interactive editor.

```
nmcli> quit
```

k. Run the `nmcli connection show id new_eth0` command.

- Pipe the output to `grep` and search for the "`autoconnect`" string.

```
# nmcli connection show id new_eth0 | grep autoconnect
connection.autoconnect:       no
...
```

- Note that the `connection.autoconnect` parameter is set to "`no`".

l. Run the `nmcli connection modify` command to modify a connection parameter for the `new_eth0` connection.

- This command does not use an interactive editor.
- Set the `ipv4.dns` parameter to `152.68.154.3`.

```
# nmcli connection modify new_eth0 ipv4.dns 152.68.154.3
```

m. Run the `nmcli connection show id new_eth0` command.

- Pipe the output to `grep` and search for the "`ipv4.dns`" string.

```
# nmcli connection show id new_eth0 | grep ipv4.dns
ipv4.dns:                       152.68.154.3
ipv4.dns-search:                --
...
```

- Note that the `ipv4.dns` parameter is set to "`152.68.154.3`".

n. Run the `nmcli connection delete` command to delete the `new_eth0` connection profile.

```
# nmcli connection delete new_eth0
Connection 'new_eth0' (...) successfully deleted.
```

o. Run the `nmcli connection show` command.

```
# nmcli connection show
NAME        UUID    TYPE              DEVICE
eth0        ...     ethernet          eth0
eth1        ...     ethernet          eth1
virbr0      ...     bridge            virbr0
new_eth1    ...     ethernet          --
```

- Note that the `new_eth0` connection no longer exists.

p. Use the `ls` command to view network interface configuration files in the `/etc/sysconfig/network-scripts` directory.

```
# ls /etc/sysconfig/network-scripts/ifcfg*
/etc/sysconfig/network-scripts/ifcfg-eth0
/etc/sysconfig/network-scripts/ifcfg-eth1
/etc/sysconfig/network-scripts/ifcfg-eth1_SAV
/etc/sysconfig/network-scripts/ifcfg-lo
/etc/sysconfig/network-scripts/ifcfg-new_eth1
```

- Note that the network interface configuration file for the `new_eth0` connection profile no longer exists.

q. Run the `nmcli connection delete` command to delete the `new_eth1` connection profile.

```
# nmcli connection delete new_eth1
Connection 'new_eth1' (...) successfully deleted.
```

7. Run the `nmcli device` object commands.

a. Run the `nmcli device help` command.

```
# nmcli device help
Usage: nmcli device { COMMAND | help }

COMMAND := { status | show | set | connect | reapply | modify |
disconnect | delete | monitor | wifi | lldp }
...
```

- Note that the `nmcli device` object provides 11 commands.

b. Run the `nmcli device status` command.

- This command displays the status of all the devices.

```
# nmcli device status
DEVICE      TYPE          STATE         CONNECTION
eth0        ethernet      connected     eth0
eth1        ethernet      connected     eth1
virbr0      bridge        connected     virbr0
```

```
lo           loopback     unmanaged    --
virbr0-nic   tun          unmanaged    --
```

c. Run the `nmcli device disconnect eth1` command.

```
# nmcli device disconnect eth1
Device 'eth1' successfully disconnected.
```

d. Run the `nmcli device status` command.

- The `status` command is the default for the `nmcli device` object and so it is not required.

```
# nmcli device
DEVICE      TYPE         STATE           CONNECTION
eth0        ethernet     connected       eth0
virbr0      bridge       connected       virbr0
eth1        ethernet     disconnected    --
lo          loopback     unmanaged       --
virbr0-nic  tun          unmanaged       --
```

- Note the change in the `eth1` device.

e. Run the `nmcli device connect eth1` command to reconnect the `eth1` device.

```
# nmcli device connect eth1
Device 'eth1' successfully activated with '...'.
```

Practice 13-5: Using the `nmtui` Utility

Overview

In this practice, you use the `nmtui` text-based utility to view network connections.

Assumptions

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

Tasks

1. Install `NetworkManager-tui` if necessary.
   - The `numtui` utility is provided by the `NetworkManager-tui` software package.
   a. Use the `rpm` command to verify that the `NetworkManager-tui` package is installed.

```
# rpm -qa | grep NetworkManager-tui
NetworkManager-tui-...
```

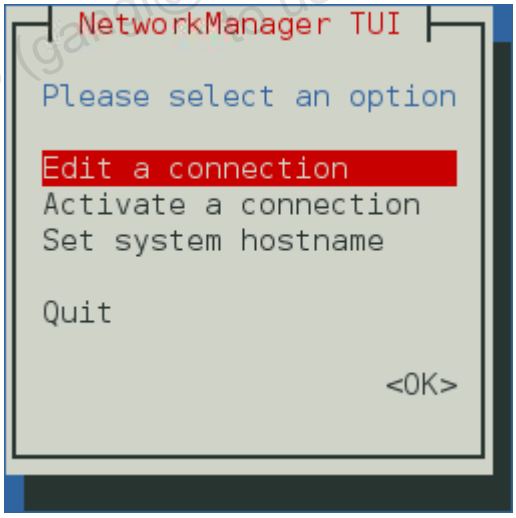
   b. If the utility is not installed, use the `yum` command to install the package.

```
# yum install NetworkManager-tui
...
```

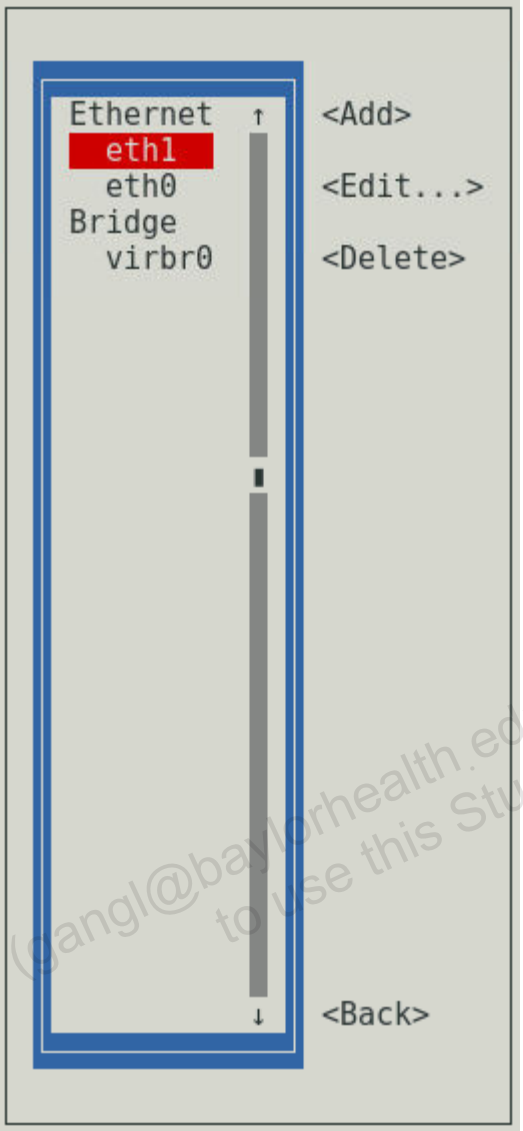
2. Use `nmtui` to configure network interfaces.
   a. Enter `nmtui` from the command line.

```
# nmtui
```

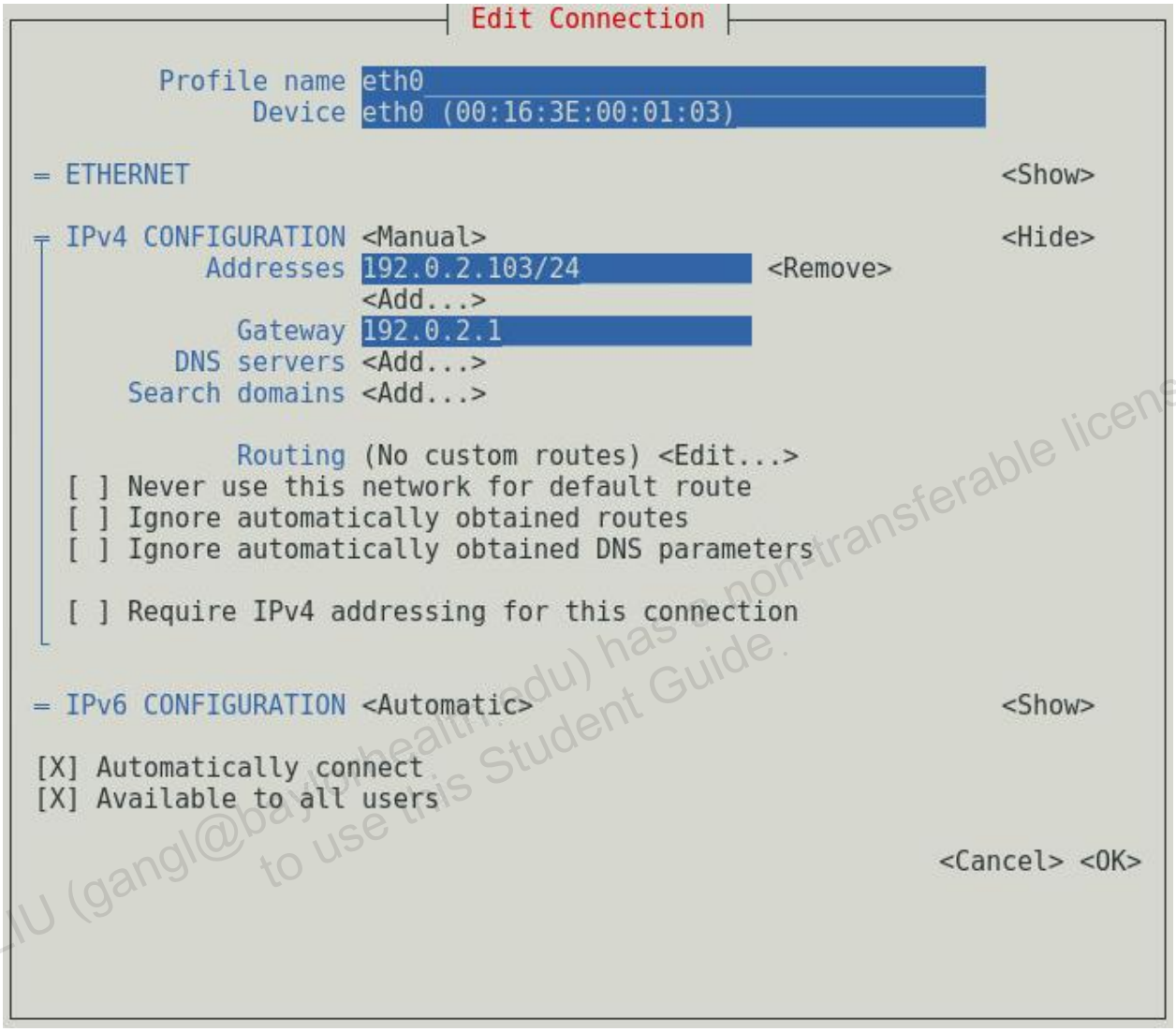
   - The screen appears as follows:



b. Use the up/down arrows to select **Edit a connection**. Use the Tab key to select **<OK>** and press Enter. The screen appears as follows:

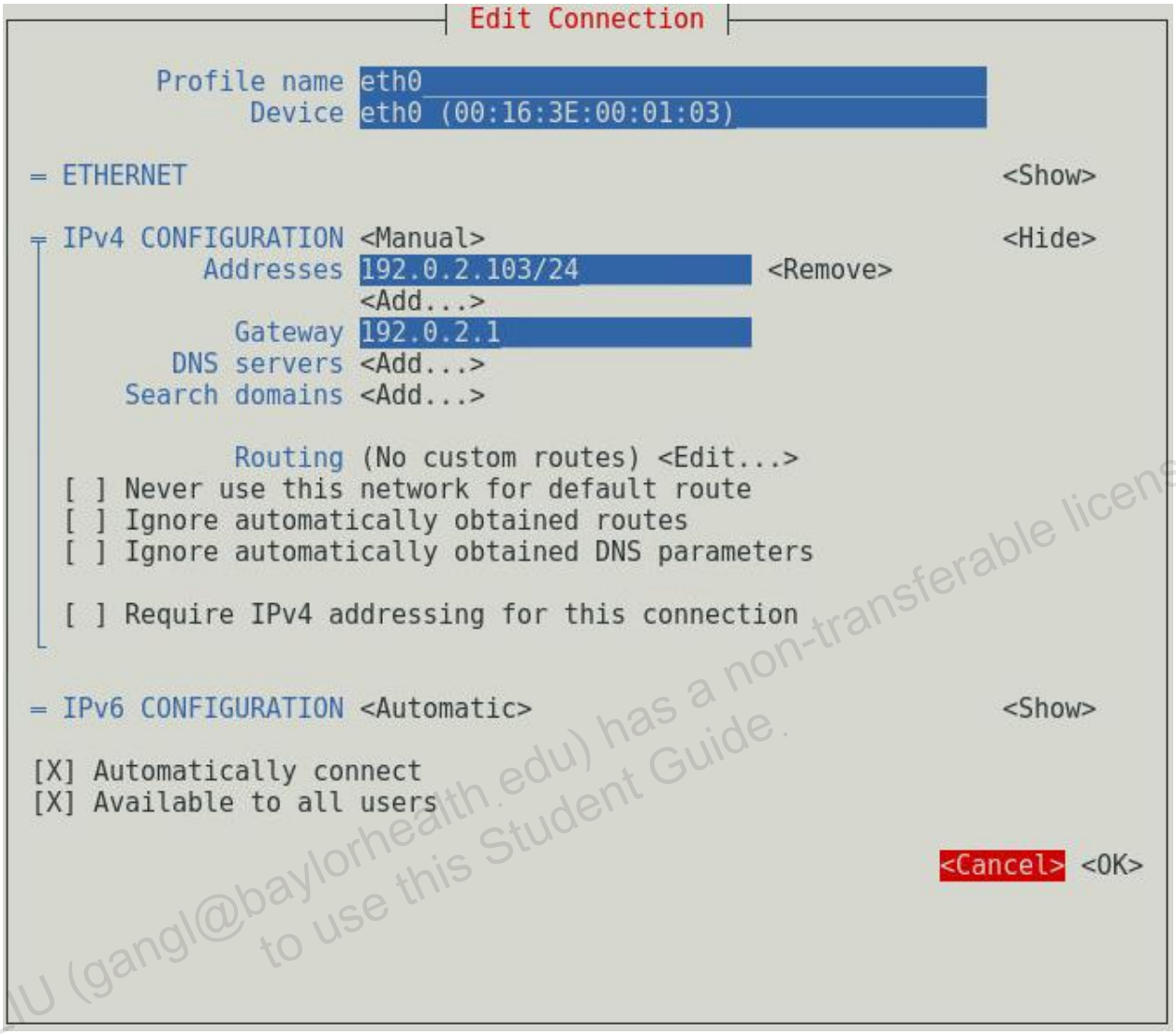


c. Use the up/down arrows to select **eth0**. Use the Tab key to select **<Edit…>** and press Enter. The screen appears as follows:



- Note that the information displayed (except the MAC address) is included in the configuration file, `/etc/sysconfig/network-scripts/ifcfg-eth0`.
- Any updates made from this screen are written to the configuration file.

d. Do not make any changes; use the down arrow key to select **<Cancel>** as shown.



e. With **<Cancel>** selected as shown, press Enter.

f. Use the Tab key to select **<Back>** and then press Enter.

g. Use the Tab key and then the down arrow key to select **<Quit>** and press Enter to exit the `nmtui` utility.

Practice 13-6: Using the `ip` Utility

Overview

In this practice, you use the `ip` utility to view network device information; add, edit, and delete a link; view IP addresses; add and delete an IPv4 address to a network device; and view and flush the ARP cache.

Assumptions

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

Tasks

1. Run the `ip` command without any options or arguments.
   - Only partial output is shown.

```
# ip
Usage: ip [OPTIONS] OBJECT {COMMAND | help}
       ip [ -force ] -batch filename
where OBJECT := { link | address | addrlabel | route | rule |
                  neigh | ntable | tunnel | tuntap | maddress |
                  mroute | mrule | monitor | xfrm | netns |
                  l2tp | fou | macsec | tcp_metrics | token |
                  netconf | ila | vrf}
...
```

   - Note that several options are available:
     - `-b, -batch <FILENAME>`: Read and invoke commands from `<FILENAME>`.
     - `-V, -Version`: Print the version of the `ip` utility.
     - `-s, -stats, -statistics`: Output more information.
   - Refer to the `ip(8)` man page for a description of all options.
   - Note that there are 23 different objects for the `ip` command.
2. Run the `ip link` object commands.
   - A “link” is a network device.
   - The “`ip link`” command is used to display and configure network devices.

   a. Run the `ip link help` command.
      - Only partial output is shown.

```
# ip link help
Usage: ip link add [link DEV] [ name ] NAME
       ...
       ip link delete { DEVICE | dev DEVICE | ... } ... [ ARGS ]
       ...
```

```
       ip link set { DEVICE | dev DEVICE | ... }
                          [ { up ... } ]
       ...
       ip link show [ DEVICE | group GROUP ] [up] ...

       ip link xstats type TYPE [ ARGS ]

       ip link afstats [ dev DEVICE ]
...
```

- Note that the `ip link` object provides six commands:
  - The six commands are `add`, `delete`, `set`, `show`, `xstats`, and `afstats`.

b. Run the `ip link show` command.

- This command shows the existing network devices.
- The `show` command is the default for the `ip link` object so it is not required.

```
# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 ...
    link/ether 52:54:00:ba:89:13 brd ff:ff:ff:ff:ff:ff
5: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc ...
    link/ether 52:54:00:ba:89:13 brd ff:ff:ff:ff:ff:ff
```

- Note that you have five network devices; two Ethernet devices (`eth0` and `eth1`), one loopback device (`lo`), and two virtual bridge devices (`vibr0` and `virbr0-nic`). The details for `vibr0` and `virbr0-nic` might vary.
- These are the same devices that are listed by the `nmcli device status` command.

c. Run the `nmcli device status` command.

- This command lists the same devices as the `ip link show` command.

```
# nmcli device
DEVICE      TYPE          STATE        CONNECTION
eth0        ethernet      connected    eth0
eth1        ethernet      connected    eth1
virbr0      ethernet      connected    virbr0
lo          loopback      unmanaged    --
virbr0-nic  tun           unmanaged    --
```

3. Use the `ip link` utility to add, edit, and delete a link.
   - The `ip link add` command adds a virtual link.
   - The following are examples of virtual link types: `vlan`, `veth`, `vcan`, `dummy`, `ifb`, `macvlan`, `vcan`, `bridge`, `ipoib`, `ip6tnl`, `ipip`, `sit`, or `vxlan`.
   - Refer to the `ip-link(8)` man page for more information about link types.
   - VLANs and advanced networking topics are covered in the *Oracle Linux System Administration II* course.

   a. Run the `ip link add` command to add a new link. Use the following parameters:
      - Physical device to operate on (`link`): `eth0`
      - Name (`name`): `eth0.10`
      - Type (`type`): `vlan`
      - VLAN ID (`id`): `10`

```
# ip link add link eth0 name eth0.10 type vlan id 10
```

   b. Run the `ip link show` command.

```
# ip link
...
6: eth0.10@eth0: <BROADCAST,MULTICAST> mtu 1500 ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
```

      - Note that you now have six network devices, including the new link `eth0.10`.

   c. Run the `nmcli device status` command.

```
# nmcli device
DEVICE      TYPE          STATE         CONNECTION
eth0        ethernet      connected     eth0
eth1        ethernet      connected     eth1
virbr0      ethernet      connected     virbr0
lo          loopback      unmanaged     --
virbr0-nic  tun           unmanaged     --
eth0.10     vlan          unmanaged     --
```

      - Note the `eth0.10` device is listed.

   d. Run the `ip link set` command to change device attributes.
      - Change the MTU for the `eth0.10` device to `1400`.

```
# ip link set eth0.10 mtu 1400
```

   e. Run the `ip link show` command to show the `eth0.10` device.
      - The `show` command is required when specifying a device as an argument.

```
# ip link show eth0.10
6: eth0.10@eth0: <BROADCAST,MULTICAST> mtu 1400 ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
```

      - Note that the MTU is `1400`.

f. Run the `ip link delete` command to delete the `eth0.10` device.

```
# ip link delete eth0.10
```

g. Run the `ip link show` command.

```
# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 ...
    link/ether 52:54:00:ba:89:13 brd ff:ff:ff:ff:ff:ff
5: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc ...
    link/ether 52:54:00:ba:89:13 brd ff:ff:ff:ff:ff:ff
```

- Note that the `eth0.10` device no longer exists.

4. Run the `ip addr` object commands.

- The "`ip addr`" command is used to display and manage IP addresses on network devices.

a. Run the `ip addr help` command.

- Only partial output is shown.

```
# ip addr help
Usage: ip address {add|change|replace} IFADDR dev IFNAME ...
       ip address del IFADDR dev IFNAME [mngtmpaddr]
       ip address {save|flush} [ dev IFNAME ] [ scope ... ] ...
       ip address [ show [ dev IFNAME ] [ scope SCOPE-ID ] ...
       ip address {showdump|restore}
IFADDR := PREFIX | ADDR peer PREFIX
...
```

- Note that the `ip addr` object provides nine commands:
  - The commands are `add`, `change`, `replace`, `del`, `show`, `save`, `flush`, `showdump`, and `restore`.
  - Refer to the `ip-address(8)` man page for more information on using `ip addr` commands.

b. Run the `ip addr show` command.

- This command shows the same information as `ip link` but also shows IP address information.
- The `show` command is the default for the `ip addr` object so it is not required.
- Only partial output is shown.

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

```
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft ...
    inet6 ...
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.103/24 brd 192.0.2.255 scope global ...
        valid_lft ...
    inet6 ...
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
       valid_lft ...
    inet6 ...
...
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 ...
    link/ether 52:54:00:ba:89:13 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope ...
       valid_lft ...
5: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc ...
    link/ether 52:54:00:ba:89:13 brd ff:ff:ff:ff:ff:ff
```

- Note the IP address information that is provided. Details for `virbr0` and `virbr0-nic` may vary.

c. Run the `ip addr add` command to add a second IP address to `eth1`.

- Use `10.1.1.1/24` as the second IPv4 address.

```
# ip addr add 10.1.1.1/24 dev eth1
```

d. Run the `ip addr show` command to show the `eth1` device.

- The `show` command is required when specifying a device as an argument.

```
# ip addr show eth1
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
       valid_lft ...
    inet 10.1.1.1/24 scope global eth1
       valid_lft ...
    inet6 ...
...
```

- Note that the device now has two IPv4 addresses.

e. Run the `ip addr del` command to delete an IP address from `eth1`.

- Delete address `10.1.1.1/24`.

```
# ip addr del 10.1.1.1/24 dev eth1
```

f. Run the `ip addr show` command to show the `eth1` device.

```
# ip addr show eth1
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global ...
       valid_lft ...
    inet6 ...
...
```

- Note that the `10.1.1.1/24` address no longer exists.

5. Run the `ip neigh` object commands.

- The "`ip neigh`" command is used to display and manage ARP cache entries.

a. Run the `ip neigh help` command.

- Only a partial output is shown.

```
# ip neigh help
Usage: ip neigh { add | del | change | replace }
                { ADDR ...
       ip neigh { show | flush } [ proxy ] [ to PREFIX ] ...
...
```

- Note that the `ip neigh` object provides six commands:
  - The commands are `add`, `del`, `change`, `replace`, `show`, and `flush`.

b. Run the `ip neigh show` command.

- The `show` command is the default for the `ip neigh` object so is not required.

```
# ip neigh
192.0.2.1 dev eth0 lladdr fe:ff:ff:ff:ff:ff STALE
```

c. If no output is produced from the `ip neigh show` command, use the `ping` command to communicate to **dom0** and the other VM guests.

- Press **Ctrl + C** to kill the `ping` command.

```
# ping dom0
64 bytes from example.com (192.0.2.1)...
CTRL-C
# ping host01
64 bytes from host01.example.com (192.0.2.101)...
CTRL-C
# ping host02
64 bytes from host02.example.com (192.0.2.102)...
CTRL-C
```

d. Run the `ip neigh show` command again to list ARP cache entries. Your output might vary.

```
# ip neigh
192.0.2.101 dev eth0 lladdr 00:16:3e:00:01:01 REACHABLE
192.0.2.1 dev eth0 lladdr fe:ff:ff:ff:ff:ff REACHABLE
192.0.2.102 dev eth0 lladdr 00:16:3e:00:01:02 REACHABLE
```

e. Run the `ip neigh flush` command to clear all entries in the ARP cache.

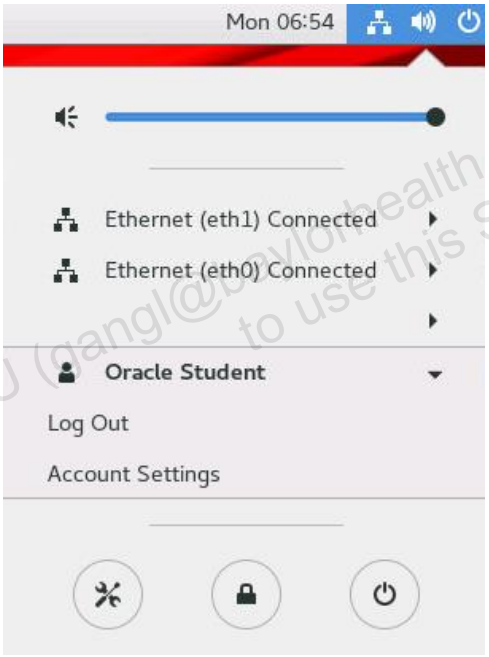
```
# ip neigh flush all
```

f. Run the `ip neigh show` command. Your output might vary.

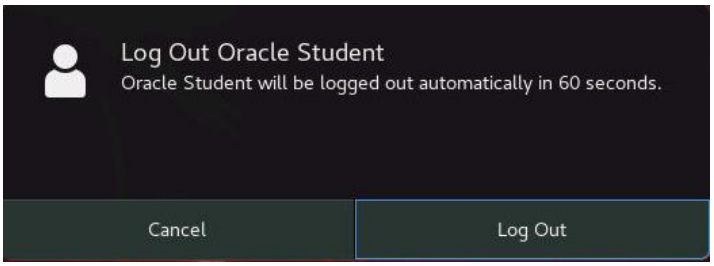
```
# ip neigh
192.0.2.1 dev eth0 FAILED
192.0.2.101 dev eth0  FAILED
192.0.2.102 dev eth0  FAILED
```

6. Log off from **host03** in preparation for the next practice.

a. Select the Power icon in the upper right of the GNOME screen > select Oracle Student.



1) Click Log Out. The following pop-up appears:



2) Click Log Out.

3) Click the **X** in the top-right corner of the GNOME login window to close the window.

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

Practices for Lesson 14: IPv6

Practices for Lesson 14: Overview

Practices Overview

In these practices, you:

- View IPv6 addresses and address types
- Make changes to an IPv6 interface configuration file setting and view the resulting IPv6 address changes

Practice 14-1: Using IPv6

Overview

In this practice, you use the `ip` utility to view IPv6 addresses. Address types and interface IDs will be examined. Some IPv6 settings in the `ifcfg-eth1` interface configuration file will be viewed. A setting will be changed, and the resulting IPv6 address changes will be noted.

Assumptions

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

Tasks

1. Log in to **host03**.
   a. Use `ssh` to log in to `host03`, providing the root password.

```
# ssh host03
root@host03's password:
Last login: ...
```

2. View some IPv6 addresses on network interfaces.
   a. Run the `ip addr` command to show IPv4 and IPv6 address information.
      - Partial output is shown. Aside from the loopback address, your IPv6 addresses will be different. Map the points made about these addresses to your command output.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state
UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP group default qlen 1000
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.103/24 brd 192.0.2.255 scope global
noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::ba72:373a:52de:9b90/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP group default qlen 1000
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global
noprefixroute eth1
       valid_lft forever preferred_lft forever
```

```
    inet6 fe80::5689:8e:77ab:2e31/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
...
```

- Note that IPv6 addresses begin with `inet6`.
- The first interface in the list is the loopback interface. The IPv6 form of the loopback specification is `::1/128`. This compressed address format represents 127 zero bits followed by a 1 bit. The scope is internal to the local host.
- `eth0` is next, with an IPv6 address of `fe80::ba72:373a:52de:9b90/64` in this example. This is a link-local unicast address, as seen by the 64-bit prefix of `fe80::` at the beginning. This prefix is followed by a 64-bit interface ID of `ba72:373a:52de:9b90`. This interface ID is not in modified EUI-64 format based on the MAC address. We can see this because the MAC address of this interface is `00:16:3e:00:01:03`. The modified EUI-64 format would be created by inserting hexadecimal `ff fe` between the first three and last three octets of this MAC address and toggling the seventh bit (from the left), giving `0216:3eff:fe00:0103` for the interface ID. Instead, the interface ID is randomly generated. The scope of this address is link-local.
- The last interface shown is `eth1`. It has an IPv6 address of `fe80::5689:8e:77ab:2e31/64`. This also is a link-local address with a randomly generated interface ID for the same reasons as described for `eth0`.

b. Run the `ip -6 addr` command to restrict output to IPv6 address information.

```
# ip -6 addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qlen 1000
    inet6 fe80::ba72:373a:52de:9b90/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qlen 1000
    inet6 fe80::5689:8e:77ab:2e31/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

- Note that IPv4 and MAC address information is not included in the output.

3. Make changes to an IPv6 interface configuration file setting and view the resulting IPv6 address changes.

a. View the interface configuration file for `eth1` using the `cat` command. Use the `cd` command to change to the `/etc/sysconfig/network-scripts` directory first.

```
# cd /etc/sysconfig/network-scripts
# cat ifcfg-eth1
TYPE=Ethernet
```

```
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=no
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=eth1
UUID=...
DEVICE=eth1
ONBOOT=yes
IPADDR=192.168.1.103
PREFIX=24
PEERDNS=no
PEERROUTES=no
```

- Note that addresses are being generated with interface IDs created following RFC 7217 with `IPV6_ADDR_GEN_MODE=stable-privacy`. This provides randomly generated interface IDs that remain stable within the local subnet, while still addressing privacy concerns.

b. Change the `IPV6_ADDR_GEN_MODE` to `eui64` and view the change to the interface ID. This setting causes interface IDs to be generated in modified EUI-64 format. Use `vi` to edit the `ifcfg-eth1` interface configuration file to make this change, which is highlighted in bold.

```
# vi ifcfg-eth1
...
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=eui64
...
```

c. Restart the network to put this change into effect.

```
# systemctl restart network
```

d. View the `eth1` network interface.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP group default qlen 1000
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.103/24 brd 192.168.1.255 scope global
noprefixroute eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::216:3eff:fe00:203/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
...
```

- The `eth1` IPv6 address, `fe80::216:3eff:fe00:203/64`, has an interface ID that is now in modified EUI-64 format, created from the MAC address. Here, the MAC address is `00:16:3e:00:02:03`. Hexadecimal `ff fe` was inserted between the first three and last three octets of the MAC address, and the seventh bit of the MAC address was toggled (from the left, binary `0000 0010` gives `02` in hexadecimal), giving `0216:3eff:fe00:0203` for the interface ID. The full address is `fe80::216:3eff:fe00:203` in compressed form.

e. Change the `IPV6_ADDR_GEN_MODE` back to `stable-privacy` and view the change to the interface ID. Use `vi` to edit the `ifcfg-eth1` interface configuration file to make this change, which is highlighted in bold.

```
# vi ifcfg-eth1
...
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
...
```

f. Restart the network to put this change into effect.

```
# systemctl restart network
```

g. View the `eth1` network interface.

```
# ip addr
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP group default qlen 1000
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
```

```
    inet 192.168.1.103/24 brd 192.168.1.255 scope global
noprefixroute eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::5689:8e:77ab:2e31/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
...
```

- The `eth1` IPv6 address has a link-local interface ID that is now in stable-privacy format: `fe80::5689:8e:77ab:2e31/64`.
- You might see a state of `tentative` after `scope link` initially. This is the state of the address until Duplicate Address Detection (DAD) is completed.

4. Log off from **host03** in preparation for the next practice.

   Use the `exit` command to log off from **host03**.

```
# exit
logout
Connection to host03 closed.
```

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

Practices for Lesson 15: OpenSSH

Practices for Lesson 15: Overview

Practices Overview

In these practices, you do the following:

- You verify that the OpenSSH packages are installed and that the `sshd` service is running.
- You use the `ssh` and `scp` utilities.
- You use the `ssh-keygen` utility to generate keys enabling connectivity without supplying a password.

Assumptions

You updated the `/etc/hosts` file in the section “Configuring the eth1 Network Interface” in the practices for the lesson titled “Network Configuration.”

Practice 15-1: Connecting to a Remote Server by Using `ssh`

Overview

In this practice, you verify that the OpenSSH packages are installed, verify that the `sshd` service is started on the server, and use the `ssh` utility to establish a connection and execute a command on a remote system.

Assumptions

- You are logged in to **dom0** as the `root` user.
- This practice is performed on the **host01** and **host03** VMs.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. Log in to **host03**.

   Use the `ssh` command to log in to **host03** as the `root` user.

   - Provide the `root` password when prompted.

```
[dom0]# ssh host03
root@host03's password:
Last login: ...
```

2. Verify that the `sshd` service is running on **host03**.

   a. Use the `rpm` command to verify that the `openssh` packages are installed.

```
[host03]# rpm -qa | grep openssh
openssh-clients-...
openssh-server-...
openssh-...
```

   - In this example, the packages are already installed.

   b. Use the `systemctl` command to verify that the `sshd` service is started.

```
[host03]# systemctl status sshd
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service;
enabled; ...
   Active: active (running) since ...
...
```

   - In this example, the service is enabled and running.

3. Log in to **host01**.

   Do not log off from **host03**.

   a. Open a second terminal window on **dom0**.

b. In the second terminal window on **dom0**, become the `root` user by entering the `su -` command and provide the `root` password.

```
[dom0]$ su -
Password:
[dom0]# whoami
root
```

c. From the second terminal window on **dom0**, use the `ssh` command to log in to **host01** as the `root` user.

- Provide the `root` password when prompted.

```
[dom0]# ssh host01
root@host01's password:
Last login: ...
```

4. Log in to remote **host03** from **host01**.

a. On **host01**, use the `rpm` command to verify that the `openssh` packages are installed.

```
[host01]# rpm -qa | grep openssh
openssh-server-...
openssh-clients-...
openssh-...
```

- In this example, the packages are already installed.

b. Use the `exit` command to log off as `root` on **host01**.

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

c. From **dom0**, use the `ssh` command to log back in to **host01** as user `oracle`.

- Provide the password when prompted.

```
[dom0]# ssh oracle@host01
oracle@host01's password:
Last login:
[oracle@host01 ~]$
```

d. Use the `ls` command to display a long listing of all files in the home directory of user `oracle`.

```
[oracle@host01 ~]$ ls -la
...
```

- Notice that there is no `~/.ssh` directory.

e. Perform a remote login to **host03** by using the `ssh` command.

- Answer "**yes**" when asked, "Are you sure ..."
- Provide the password when prompted.

```
[oracle@host01 ~]$ ssh host03
```

```
The authenticity of host 'host03 (192.0.2.103)' can't be
established.
ECDSA key fingerprint is SHA ...
ECDSA key fingerprint is MD5 ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host03,192.0.2.103' (ECDSA) to the
list of known hosts.
oracle@host03's password:
Last login: ...
[oracle@host03 ~]$
```

f. Use the `hostname` command to display the host name to confirm that you successfully logged in to **host03**.

```
[oracle@host03 ~]$ hostname
host03.example.com
```

g. Use the `logout` command to close the `ssh` connection to **host03**. Use the `hostname` command to confirm that you are back to **host01**.

```
[oracle@host03 ~]$ logout
Connection to host03 closed.
[oracle@host01 ~]$ hostname
host01.example.com
```

5. View the `.ssh` directory in the `oracle` user's home directory.

a. Use the `ls` command to display a long listing of all files in the home directory of user `oracle`.

```
[oracle@host01 ~]$ ls -la
...
drwx------. 2 oracle oracle 4096 <date_time> .ssh
```

- Notice that there is now a `~/.ssh` directory.

b. Use the `cd` command to change to the `~/.ssh` directory and then use `ls` to view the contents of the directory.

```
[oracle@host01 ~]$ cd .ssh
[oracle@host01 .ssh]$ ls
known_hosts
```

- Notice that the `known_hosts` file was created.

c. Use the `cat` command to view the contents of the `known_hosts` file.

```
[oracle@host01 .ssh]$ cat known_hosts
host03,192.0.2.103 ecdsa-sha2-nistp256 ...
```

- Notice that **host03** is now a "known host".

6. Log in to remote **host03** from **host01**.

a. Perform a remote login to **host03** using the `ssh` command.

- Provide the password when prompted.

```
[oracle@host01 .ssh]$ ssh host03
oracle@host03's password:
Last login...
[oracle@host03 ~]$
```

- Notice that you still need to provide a password, but are not asked to confirm this time, because the `known_hosts` file identifies **host03** as a "known host."

b. Use the `logout` command to close the `ssh` connection to **host03**.

- Use the `hostname` command to confirm that you are back to **host01**.
- Use the `cd` command to change back to the user's home directory.

```
[oracle@host03 ~]$ logout
Connection to host03 closed.
[oracle@host01 .ssh]$ hostname
host01.example.com
[oracle@host01 .ssh]$ cd
[oracle@host01 ~]$
```

c. Log in to **host03** as user `root` and run the `ls` command with a single `ssh` command.

- Provide the password when prompted.

```
[oracle@host01 ~]$ ssh root@host03 ls
root@host03's password:
anaconda-ks.cfg
initial-setup-ks.cfg
[oracle@host01 ~]$
```

- Note that the `ls` command ran on the remote system displaying the contents of the remote directory, and then the remote connection closed.

d. Use the `hostname` command to confirm that you are back to **host01**.

```
[oracle@host01 ~]$ hostname
host01.example.com
```

e. Use the `logout` command to log off from **host01**.

```
[oracle@host1 ~]$ logout
Connection to host01 closed.
```

Practice 15-2: Configuring OpenSSH to Connect Without a Password

Overview

In this practice, you use the `ssh-keygen` command to generate an RSA key pair and configure OpenSSH to connect to a remote system without supplying a password. You also use the `scp` command in this practice.

Assumptions

- This practice is performed on the **host01** and **host03** VMs.
- You are logged in to **host03** as the `root` user.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. Use the `ssh-keygen` command to create the public and private parts of an RSA key.
   a. As the `root` user on **host03**, use the `su` command to become the `oracle` user. Use the `whoami` and `pwd` commands to confirm you are logged in as the `oracle` user.

```
[root@host03 ~]# su - oracle
Last login:
[oracle@host03 ~]$ whoami
oracle
[oracle@host03 ~]$ pwd
/home/oracle
```

   b. Use the `ls` command to view the contents of the `~/.ssh` directory.

```
[oracle@host03 ~]$ ls ~/.ssh
ls: cannot access /home/oracle/.ssh: No such file or directory
```

   - Notice that the directory does not exist.

   c. Use the `ssh-keygen -t rsa` command to create the public and private parts of an RSA key.
   - Accept all the defaults.

```
[oracle@host03 ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/oracle/.ssh/id_rsa):
ENTER
Created directory '/home/oracle/.ssh'.
Enter passphrase (empty for no passphrase): ENTER
Enter same passphrase again: ENTER
Your identification has been saved in /home/oracle/.ssh/id_rsa.
Your public key has been saved in /home/oracle/.ssh/id_rsa.pub.
The key fingerprint is:
... oracle@host03.example.com
```

```
The key's randomart image is:
...
```

d. Use the `ls` command to view the contents of the `~/.ssh` directory.

```
[oracle@host03 ~]$ ls ~/.ssh
id_rsa     id_rsa.pub
```

- Note that the `ssh-keygen` command generated two keys.

2. Use the `scp` command to copy `~/.ssh/id_rsa.pub` on the local system (**host03**) to `~/.ssh/authorized_keys` on the remote system (**host01**).

- Answer "**yes**" to continue.
- Provide the password when prompted.

```
[oracle@host03 ~]$ scp ~/.ssh/id_rsa.pub
host01:~/.ssh/authorized_keys
The authenticity of host 'host01 (192.0.2.101)' can't be
established.
ECDSA key fingerprint is SHA256: ...
ECDSA key fingerprint is MD5: ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host01,192.0.2.101' (ECDSA) to the
list of known hosts.
oracle@host01's password:
id_rsa.pub                                    100%  407   1.2MB/s   00:00
[oracle@host03 ~]$
```

- Because you are connecting to this OpenSSH server for the first time, you are asked to confirm the connection.
- Note that a password is required to make the connection.
- Note that the file is copied, but you are still connected to the local system (**host03**).

3. Log in to remote **host01** from **host03**.

a. Perform a remote login to **host01** by using the `ssh` command.

```
[oracle@host03 ~]$ ssh host01
Last login:...
```

- Note that you no longer need to enter a password.

b. Use the `hostname` command to confirm that you successfully logged in to **host01**.

```
[oracle@host01 ~]$ hostname
host01.example.com
```

c. Use the `ls` command to view the contents of the `~/.ssh` directory.

```
[oracle@host01 ~]$ ls ~/.ssh
authorized_keys   known_hosts
```

- Note the existence of the `authorized_keys` file, which allowed you to connect without supplying a password.

4. Remove files and log off.

a. Use the `rm` command to remove the `authorized_keys` file on **host01**.

```
[oracle@host01 ~]$ rm ~/.ssh/authorized_keys
```

b. Use the `logout` command to close the connection to **host01**.

- Use the `hostname` command to confirm that you are back to **host03**.

```
[oracle@host01 ~]$ logout
Connection to host01 closed.
[oracle@host03 ~]$ hostname
host03.example.com
```

c. Use the `rm` command to remove the `id_rsa` and `id_rsa.pub` files on **host03**.

```
[oracle@host03 ~]$ rm ~/.ssh/id_rsa
[oracle@host03 ~]$ rm ~/.ssh/id_rsa.pub
```

d. Use the `exit` command to log out as the `oracle` user and return to the `root` logon.

- Use the `whoami` command to confirm that you are logged in as `root`.

```
[oracle@host03 ~]$ exit
logout
[root@host03 ~]# whoami
root
```

e. Log off from **host03** in preparation for the next practice.

- Use the `exit` command to close the `ssh` connection to **host03**.

```
[root@host03 ~]# exit
logout
Connection to host03 closed.
```

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

Practices for Lesson 16: Security Administration

Practices for Lesson 16: Overview

Practices Overview

In these practices, you:

- Configure a `chroot` jail
- Configure a `chroot` jail for `ftp` users
- Explore and configure `firewalld`
- Configure `iptables`
- Configure a TCP wrapper

Assumptions

- You updated the `/etc/hosts` file in the practices for the lesson titled “Network Configuration.”
- You created a local Yum repository in the practices for the lesson titled “Package Management.”

Practice 16-1: Configuring a `chroot` Jail

Overview

In this practice, you configure a `chroot` jail and copy all files required to run the `/bin/bash` shell in the `chroot` jail.

Assumptions

- You are logged in to **dom0** as the `root` user.
- This practice is performed on **host03**.

Tasks

1. Log in to **host03**.
   a. Connect to the **host03** guest by using the `xm vncviewer host03&` command.

```
# xm vncviewer host03&
```

   The GNOME login window appears.
   b. Select Oracle Student from the GNOME login window; enter the password.
   c. Right-click on the GNOME desktop and select **Open Terminal** from the short-cut menu.
2. Attempt to create a `chroot` jail.
   a. Use the `cd` and `pwd` commands to ensure that you are in the `oracle` user's home directory.

```
[oracle@host03 ~]$ cd
[oracle@host03 ~]$ pwd
/home/oracle
```

   b. As the `oracle` user, use the `mkdir` command to make a `jail` directory in the current directory.

```
[oracle@host03 ~]$ mkdir jail
```

   c. Use the `su` command to become the `root` user.
      - Use the `whoami` command to confirm that you are the `root` user.

```
[oracle@host03 ~]$ su -
Password:
Last login: ...
[root@host03 ~]# whoami
root
```

   d. Use the `echo` command to display the value of the `SHELL` variable.

```
[root@host03 ~]# echo $SHELL
/bin/bash
```

      - In this example, `SHELL=/bin/bash`.

e. As the `root` user, use the `chroot` command to create a `chroot` jail in the `/home/oracle/jail` directory.

```
[root@host03 ~]# chroot /home/oracle/jail
chroot: failed to run command '/bin/bash': No such file or
directory
```

- Note that if you do not specify a command as an argument, `chroot` attempts to run the value of the `SHELL` variable, `/bin/bash`, in the `chroot` jail directory, `/home/oracle/jail`.
- The command failed because `/bin/bash` was not found in `/home/oracle/jail`.

3. As the `oracle` user, copy `/bin/bash` and other necessary files into the `~/jail` directory.

a. Use the `exit` command to log off as `root` and return to the `oracle` user login.

```
[root@host03 ~]# exit
logout
[oracle@host03 ~]$
```

b. Use the `cd` command to change to the ~/`jail` directory.

- Use the `mkdir` command to make a `bin` directory.

```
[oracle@host03 ~]$ cd ~/jail
[oracle@host03 jail]$ mkdir bin
```

c. Use the `cp` command to copy `/bin/bash` into `~/jail/bin`.

```
[oracle@host03 jail]$ cp /bin/bash ~/jail/bin
```

d. Use the `ldd` command to determine which shared libraries are required by `/bin/bash`.

```
[oracle@host03 jail]$ ldd /bin/bash
linux-vdso.so.1 => (0x0000...)
libtinfo.so.5 => /lib64/libtinfo.so.5 (0x0000...)
libdl.so.2 => /lib64/libdl.so.2 (0x0000...)
libc.so.6 => /lib64/libc.so.6 (0x0000...)
/lib64/ld-linux-x86-64.so.2 (0x0000...)
```

- In this example, there are four shared library files in `/lib64` used by `/bin/bash`.

e. In the `/home/oracle/jail` directory, use the `mkdir` command to make a `lib64` directory.

```
[oracle@host03 jail]$ mkdir lib64
```

f. Use the `cp` command to copy the four shared library files required for `/bin/bash` from `/lib64` to `~/jail/lib64`.

```
[oracle@host03 jail]$ cp /lib64/libtinfo.so.5 ~/jail/lib64
[oracle@host03 jail]$ cp /lib64/libdl.so.2 ~/jail/lib64
[oracle@host03 jail]$ cp /lib64/libc.so.6 ~/jail/lib64
[oracle@host03 jail]$ cp /lib64/ld-linux-x86-64.so.2
~/jail/lib64
```

4. Create a `chroot` jail successfully.

   a. Use the `su` command to become the `root` user.

```
[oracle@host03 jail]$ su -
Password:
Last login: ...
[root@host03 ~]#
```

   b. As the `root` user, use the `chroot` command to create a `chroot` jail in the `/home/oracle/jail` directory.

```
[root@host03 ~]# chroot /home/oracle/jail
bash-4.2#
```

   - Note that the `chroot` command was successful—no errors occurred and the `/bin/bash` program executed.

   c. Use the `pwd` command to display the current directory.

```
bash-4.2# pwd
/
```

   - Note that the output indicates that the current directory is the root-level directory, even though the actual directory is `/home/oracle/jail`.

   d. Use the `exit` command to exit the `chroot` jail.

```
bash-4.2# exit
exit
[root@host03 ~]#
```

Practice 16-2: Configuring a `chroot` Jail for `ftp` Users

Overview

In this practice, you:

- Confirm that `anonymous ftp` users are placed in a `chroot` jail on a `vsftpd` server by default
- Configure local `ftp` users to be placed in a `chroot` jail

Assumptions

- You are the `root` user on **host03**.
- This practice is performed on **host01** and **host03** VMs.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. Configure **host03** as an FTP server.
   a. Use the `rpm` command to check whether the `vsftpd` package is installed on **host03**.

```
[host03]# rpm -q vsftpd
vsftpd-...
```

   - You can see that the `vsftpd` package is already installed.

   b. If the `vsftpd` package is not already installed, install it using the `yum` command.

```
[host03]# yum install vsftpd
...
Complete!
```

   c. Use the `systemctl` command to check the status of the `vsftpd` service.

```
[host03]# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service;
disabled; vendor preset: disabled)
   Active: inactive (dead)
```

   - Here, the `vsftpd` service is disabled and inactive.

   d. Use the `systemctl` command to start the `vsftpd` service.

```
[host03]# systemctl start vsftpd
```

   e. Use the `systemctl` command to configure `vsftpd` to start at boot time.

```
[host03]# systemctl enable vsftpd
Created symlink from /etc/systemd/system/multi-
user.target.wants/vsftpd.service to
/usr/lib/systemd/system/vsftpd.service.
```

f. Verify that the `vsftpd` service is `enabled` and `running` by using the `systemctl` command.

```
[host03]# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service;
enabled; vendor preset: disabled)
   Active: active (running) since <date_time; time> ago
 Main PID: 16215 (vsftpd)
   CGroup: /system.slice/vsftpd.service
           └─16215 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

<date_time> host03.example.com systemd[1]: Starting Vsftpd ftp
daemon...
<date_time> host03.example.com systemd[1]: Started Vsftpd ftp
daemon.
```

- You can see that the `vsftpd` service is now `enabled` and `running`.

g. Use the `systemctl` command to stop `firewalld`.

```
[host03]# systemctl stop firewalld
```

2. Configure **host01** as an FTP client.

a. From a second terminal window on **dom0**, become the `root` user, if necessary, by entering the `su -` command and providing the `root` password.

- If you already have a second terminal window open on **dom0** in which you are the `root` user, skip to the next step.

```
[dom0]$ su -
Password:
[dom0]# whoami
root
```

b. From the second terminal window on **dom0**, use the `ssh` command to log in to **host01** as the `root` user.

```
[dom0]# ssh host01
root@host01's password:
Last login: ...
```

c. Use the `rpm` command to verify that the `autofs` package is installed.

```
[host01]# rpm -q autofs
autofs-...
```

- In this example, the package is already installed.

d. Use the `systemctl` command to verify that the `autofs` service is running.

```
[host01]# systemctl status autofs
autofs.service - Automounts filesystems on demand
   Loaded: loaded (/usr/lib/systemd/system/autofs.service...
```

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

- In this example, the `autofs` service is disabled and not running.

e. Use the `systemctl` command to start the `autofs` service.

```
[host01]# systemctl start autofs
```

f. Use the `rpm` command to install the `ftp` package on **host01**.

- You cannot use `yum` because the yum repository is not configured on **host01**.
- Use the `cd /misc/cd/Packages` command to automount the virtual `cdrom`, which contains the Oracle Linux `dvd.iso` image.

```
[host01]# cd /misc/cd/Packages
[host01]# rpm -Uvh ftp-0.17-67.el7.x86_64.rpm
Preparing...    ########################################## [100%]
Updating / installing...
   1:ftp...     ########################################## [100%]
```

3. Confirm that `anonymous` ftp users are placed in a `chroot` jail by default.

a. From **host03**, use the `ls` command to list the contents of the `/var/ftp` directory.

```
[host03]# ls -l /var/ftp
...
drwxr-xr-x...  pub
```

- Note that `/var/ftp` contains a single directory, `pub`.

b. Copy the `/root/anaconda-ks.cfg` file to `/var/ftp/pub` and rename it as `test_file`.

```
[host03]# cp /root/anaconda-ks.cfg /var/ftp/pub/test_file
```

c. From **host01**, `ftp` to **host03** as `anonymous` user.

- Press Enter when prompted for a password.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 3.0.2)
Name...: anonymous
331 Please specify the password.
Password: ENTER
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

d. From **host01**, use the `ls` command to list the contents of the current directory.

```
ftp> ls
...
drwxr-xr-x ... pub
226 Directory send OK.
```

```
ftp>
```

- Note that the current directory contains a single directory, `pub`.

e. Use the `ls` command to list the contents of the `pub` directory.

```
ftp> ls pub
...
-rw------- ... test_file
226 Directory send OK.
ftp>
```

- Note that the `pub` directory contains a single file, `test_file`.
- This confirms that the current location of the `anonymous` FTP user is `/var/ftp` on **host03**.

f. Use the `pwd` command to display the current directory.

```
ftp> pwd
257 "/"
ftp>
```

- Note that the output indicates that the current directory is the `root`-level directory, even though the actual directory is `/var/ftp`.
- This confirms that `anonymous` users are placed in a `chroot` jail by default.

g. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

4. Confirm that local `ftp` users are placed in their home directory by default.

a. From **host01**, `ftp` to **host03** as the `oracle` user.

- Provide the password when prompted.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 3.0.2)
Name...: oracle
331 Please specify the password.
Password:
230 Login successful.
...
ftp>
```

b. Use the `pwd` command to display the current directory.

- Use the `ls` command to display the contents of the current directory.

```
ftp> pwd
257 "/home/oracle"
ftp> ls
...
drwxr-xr-x ...  Desktop
```

```
drwxr-xr-x ...  Documents
drwxr-xr-x ...  Downloads
...
ftp>
```

- The output indicates that the `oracle` user was placed in its home directory.
- This is the case for all local users that access a `vsftpd` server; they are placed in their home directory by default, and not in a `chroot` jail.

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

```
ftp> quit
221 Goodbye.
```

5. On **host03**, enable options in the `/etc/vsftpd/vsftpd.conf` file to put local users in a `chroot` jail.

a. Use the `vi` editor to make the following changes:

- Remove the `#` sign to uncomment the `chroot_local_user` directive and ensure it is set to `YES`.
- Add the `allow_writeable_chroot` directive and set to `YES`.
- Ensure the following `chroot` directives are commented out (preceded with a `#` sign, as shown):
  - `#chroot_list_enable=YES`
  - `#chroot_list_file=/etc/vsftpd/chroot_list`

```
[host03]# vi /etc/vsftpd/vsftpd.conf
chroot_local_user=YES
allow_writeable_chroot=YES
#chroot_list_enable=YES
#chroot_list_file=/etc/vsftpd/chroot_list
```

- After making changes to the `vsftpd.conf` file, you must restart the `vsftpd` service.

b. Use the `systemctl` command to restart the `vsftpd` service.

```
[host03]# systemctl restart vsftpd
```

c. On **host03**, use the `setenforce 0` command to change SELinux to “permissive” mode. Use the `getenforce` command before and after to confirm the change.

- SELinux is covered in another course.
- For the purposes of this practice, set SELinux to “permissive” mode.

```
[host03]# getenforce
Enforcing
[host03]# setenforce 0
[host03]# getenforce
Permissive
```

6. Verify that the `chroot` settings are working.

a. From **host01**, `ftp` to **host03** as the `oracle` user.

- Provide the password when prompted.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 3.0.2)
Name...: oracle
331 Please specify the password.
Password

:
230 Login successful.
...
ftp>
```

b. Use the `pwd` command to display the current directory.

- Use the `ls` command to list the contents of the current directory.

```
ftp> pwd
257 "/"
ftp> ls
...
drwxr-xr-x ...  Desktop
drwxr-xr-x ...  Documents
drwxr-xr-x ...  Downloads
...
ftp>
```

- The output indicates that the current directory is the root-level directory even though the actual directory is `/home/oracle` on the `vsftpd` server, **host03**.

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

```
ftp> quit
221 Goodbye.
```

7. Restore `vsftpd.conf` settings on **host03** to their original state.

a. On **host03**, disable options in the `/etc/vsftpd/vsftpd.conf` file to put local users in a `chroot` jail. Use the `vi` editor to make the following changes:

- Insert a `#` sign to comment out the `chroot_local_user` directive.
- Insert a `#` sign to comment out the `allow_writeable_chroot` directive.

```
[host03]# vi /etc/vsftpd/vsftpd.conf
#chroot_local_user=YES
#allow_writeable_chroot=YES
```

- After making changes to the `vsftpd.conf` file, you need to restart the `vsftpd` service.

b. Use the `systemctl` command to restart the `vsftpd` service.

```
[host03]# systemctl restart vsftpd
```

c. On **host03**, use the `setenforce 1` command to change SELinux to “enforcing” mode. Use the `getenforce` command to confirm the change.

```
[host03]# setenforce 1
[host03]# getenforce
Enforcing
```

d. From **host01**, `ftp` to **host03** as the `oracle` user.

- Provide the password when prompted.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 3.0.2)
Name...: oracle
331 Please specify the password.
Password:
230 Login successful.
...
ftp>
```

e. Use the `pwd` command to display the current directory.

```
ftp> pwd
257 "/home/oracle"
ftp>
```

- Note that the `oracle` user is now placed in their home directory and not in a `chroot` jail.

f. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

Practice 16-3: Exploring `firewalld`

Overview

In this practice, you:

- Start the `firewalld` service if necessary
- Start the Firewall Configuration GUI
- Explore `firewalld` zones
- Change the default zone
- Explore **runtime** and **permanent** configuration modes
- Explore `firewalld` services

Assumptions

- You logged in to **host03** by using the `xm vncviewer host03&` command.
- You are the `root` user on **host03** VM.

Tasks

1. If necessary, start the `firewalld` service on **host03**.
   a. Use the `systemctl` command to check if the `firewalld` service is running.

```
# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service...
   Active: inactive (dead) since <date_time; time> ago
...
```

   - In this example, `firewalld` is enabled but is not running.

   b. Use the `firewall-cmd` command-line utility to check if the `firewalld` service is running.
      - This command provides another method to check the status of `firewalld`.

```
# firewall-cmd --state
not running
```

      - Here, the `firewalld` service is not running.

   c. If the `firewalld` service is not running, use the `systemctl` command to start it. If `firewalld` is already running, go to "Start the Firewall Configuration GUI."

```
# systemctl start firewalld
```

   d. Use the `systemctl` command to check if the `firewalld` service is running.

```
# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service...
   Active: active (running) since <date_time; time> ago
...
```

- The `firewalld` service is now running.

e. Use the `firewall-cmd` command-line utility to check if the `firewalld` service is running.

```
# firewall-cmd --state
running
```

- This confirms that the `firewalld` service is running.

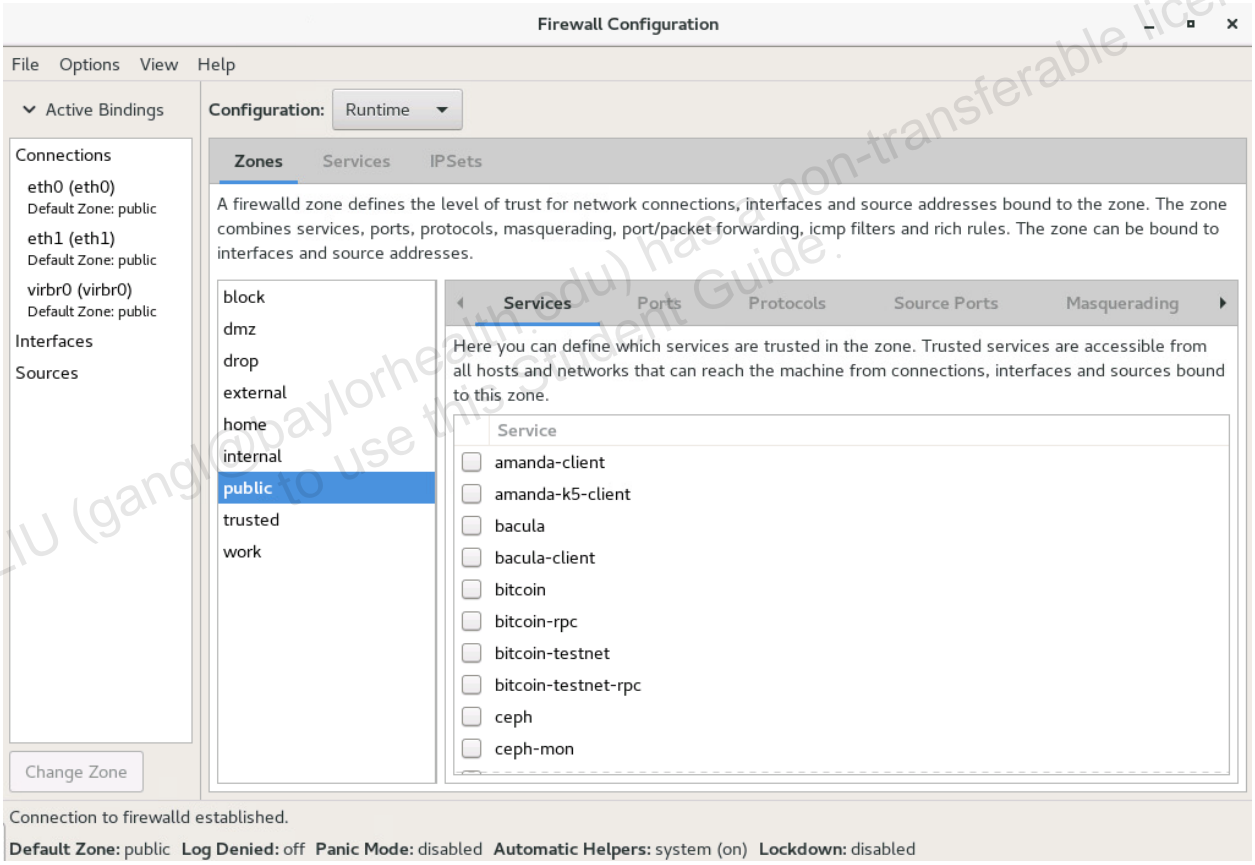
2. Start the Firewall Configuration GUI.

a. Use the `firewall-config` command to start the GUI.

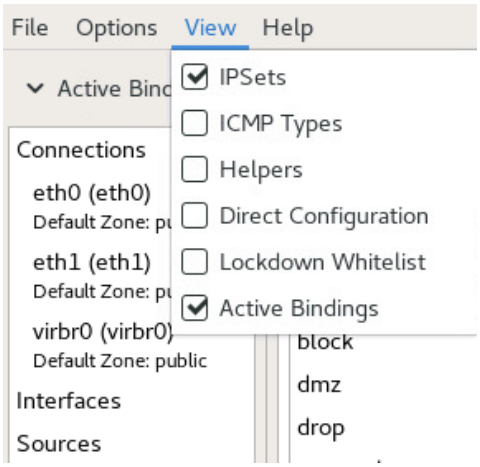
- Run the command in the background so that you can continue to enter commands from the command line.

```
# firewall-config&
```

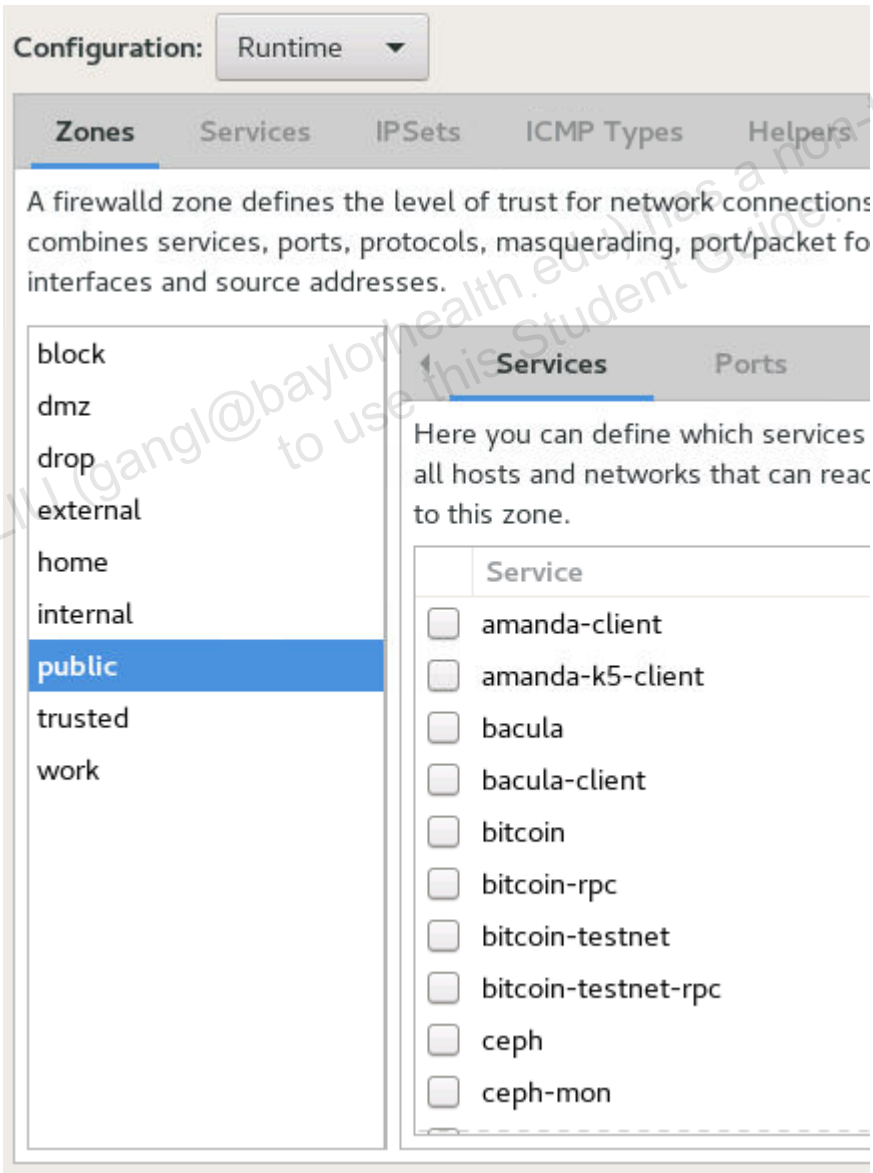
- The GUI appears as shown:



- Notice the message “Connection to firewalld established” at the lower left of the window. This indicates that the `firewalld` service is running.
- If the **ICMP Types**, **Helpers**, **Direct Configuration**, and **Lockdown Whitelist** tabs are not shown, select **View** from the menu bar and select each option.



3. Explore `firewalld` zones.
   - Note that there are nine predefined zones displayed on the GUI as shown:



- Each of these zones is defined by individual files in the `/usr/lib/firewalld/zones` directory.

a. From a command prompt, use the `ls` command to display the contents of the `/usr/lib/firewalld/zones` directory.

```
# ls /usr/lib/firewalld/zones
block.xml  drop.xml      home.xml      public.xml  work.xml
dmz.xml    external.xml  internal.xml  trusted.xml
```

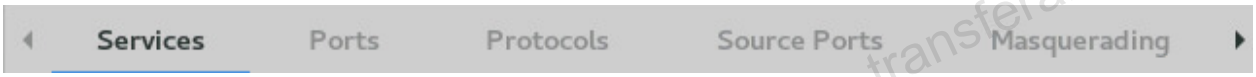
- Note that there is an XML configuration file for each of the nine predefined zones.

b. Use the `firewall-cmd` command to display the available zones.

```
# firewall-cmd --get-zones
block dmz drop external home internal public trusted work
```

- Note that this command shows the same nine predefined zones.

c. From the GUI, note that for each zone you can select the lower row of tabs and configure the associated parameters. Use the scroll arrows to view more tabs.



- Ensure that the **Zones** tab on the upper row of tabs is selected, then select the **Services** tab on the lower row of tabs (not the **Services** tab on the upper row of tabs next to **Zones**).
  - Note that you can select which services are trusted in a specific zone.
- Select the **Ports** tab.
  - Click the **Add** button.
  - Note that you can add additional ports or port ranges that need to be accessible from hosts or networks.
  - Click the **Cancel** button.
- Select the **Protocols** tab.
  - Click the **Add** button.
  - Note that you can allow access for specified protocols.
  - Click the **Cancel** button.
- Select the **Source Ports** tab.
  - Click the **Add** button.
  - Note that you can allow access for specific ports/port ranges by protocol.
  - Click the **Cancel** button.
- Select the **Masquerading** tab.
  - Note that you can enable IPv4 masquerading, which causes hosts on your local network to appear as a single IP address on the Internet.
- Select the **Port Forwarding** tab.
  - Click the **Add** button.
  - Note that you can forward inbound traffic to different ports or different systems.
  - Click the **Cancel** button.
- Select the **ICMP Filter** tab.

  - Note that you can block selected types of ICMP messages.
- Select the **Rich Rules** tab.
  - Click the **Add** button.
  - Note that you can extend existing `firewalld` rules to include additional source and destination addresses and logging and auditing actions.
  - Click the **Cancel** button.
- Select the **Interfaces** tab.
  - Note that the `eth0` and `eth1` interfaces are bound to the `public` zone.
  - From a command prompt, run the following `firewall-cmd` command to display the active zones.

```
# firewall-cmd --get-active-zone
public
  interfaces: eth0 eth1
```

  - Note that the output of this `firewall-cmd` command confirms that the `eth0` and `eth1` interfaces are bound to the `public` zone.
  - From the GUI, select either the `eth0` or `eth1` interface.
  - Click the **Edit** button.
  - Note that you can assign an interface to a different zone.
  - Click the **Close** button.
  - Note that you can also assign an interface to a different zone by selecting **Options > Change Zones of Connections** from the menu bar.
- Click the right arrow, which selects the **Sources** tab.
  - Click the **Add** button.
  - Note that you can bind source addresses to a specific zone.
  - Click the **Cancel** button.
- Click the left arrow until the **Services** tab is selected.

d. From a command prompt, use the `cat` command to view the main `firewalld` configuration file, `/etc/firewalld/firewalld.conf`.

```
# cat /etc/firewalld/firewalld.conf
# firewalld config file

# default zone
# The default zone used if an empty zone string is used.
# Default: public
DefaultZone=public
...
```

- Note that the default zone, `public`, is defined in this configuration file.
- Note that **Default Zone: public** is displayed on the bottom of the GUI as shown:

e. Use the `cat` command to view the contents of the `public.xml` zone file:

```
# cat /usr/lib/firewalld/zones/public.xml
<?xml version="1.0" encoding="utf-8"?>
<zone>
    <short>Public</short>
    <description>For use in public areas. You do not trust the
other computers on networks to not harm your computer. Only
selected incoming connections are accepted.</description>
    <service name="ssh"/>
    <service name="dhcpv6-client"/>
</zone>
```

- Note that the description states, “Only selected incoming connections are accepted.”
- In this example, the selected incoming connections (trusted services) are `ssh` and `dhcpv6-client`.

f. From the Firewall Configuration GUI, scroll down the **Service** window and note that the following two services are trusted for the `public` zone by default:

- `dhcpv6-client`
- `ssh`
- These services correspond with the service entries in the `public.xml` file.

4. Change the default zone.

a. From a command prompt, use the `firewall-cmd` command to display the current default zone.

```
# firewall-cmd --get-default-zone
public
```

- Note that the `public` zone is the default zone.

b. Use the `firewall-cmd` command to change the default zone to the `work` zone.

```
# firewall-cmd --set-default-zone=work
success
```

c. Use the `firewall-cmd` command to display the current default zone.

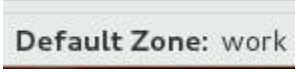
```
# firewall-cmd --get-default-zone
work
```

- Note that the `work` zone is the default zone.

d. Use the `grep -i` command to search for the string “`defaultzone`” in the `firewalld` configuration file.

```
# grep -i defaultzone /etc/firewalld/firewalld.conf
DefaultZone=work
```

- Note that the preceding command, which set the default zone, updated the `DefaultZone` setting in the configuration file.
- Note that the **Default Zone: work** is displayed on the bottom of the GUI as shown:

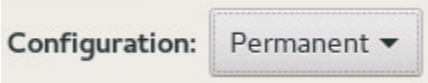


e. Use the `firewall-cmd` command to change the default zone back to the `public` zone.

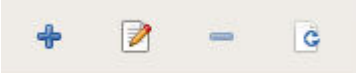
```
# firewall-cmd --set-default-zone=public
success
```

5. Explore **runtime** and **permanent** configuration modes.

a. From the GUI, click the **Configuration** drop-down menu and select **Permanent**.

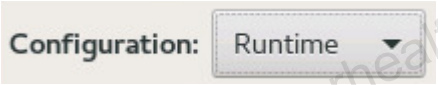


- When **Permanent** is selected, changes are applied when the `firewalld` service restarts.
- Note that you can restart `firewalld` from the GUI by selecting **Options > Reload Firewalld** from the menu bar.
- Note that in **Permanent** configuration mode, the following menu appears under the list of zones:



- These options allow you to add, edit, remove, and load defaults, relative to a selected configuration item.

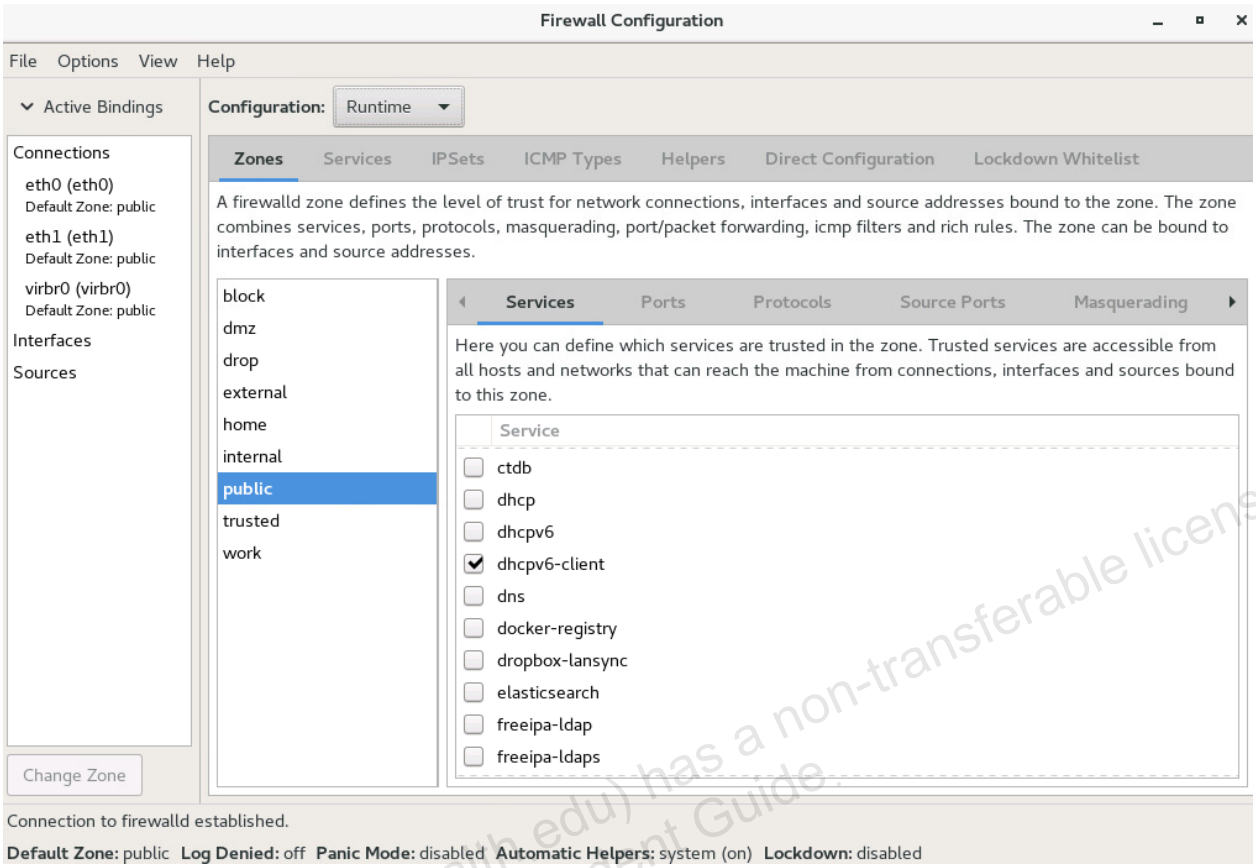
b. Click the **Configuration** drop-down menu and select **Runtime**.



- Notice the menu under the list of zones is no longer displayed.
- When **Runtime** is selected, changes to current firewall settings take effect immediately.

c. Make the following selections from the Firewall Configuration GUI:

- Select the **Runtime** configuration option.
- Select **Zones** from the top row of tabs.
- Select **Services** from the lower row of tabs.
- Select the `public` zone from the list of zones.
- The Firewall Configuration window appears as shown:



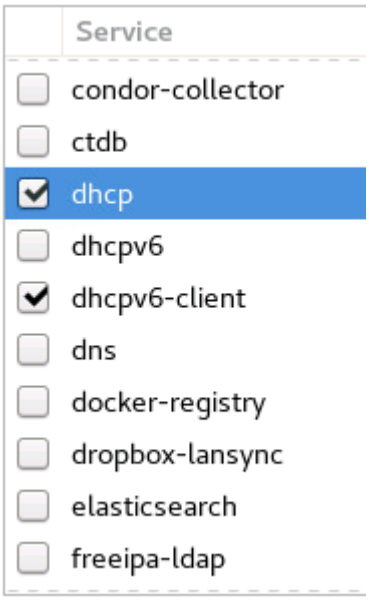
d. From a command prompt, use the `grep` command to search for the "`service`" string in the `/usr/lib/firewalld/zones/public.xml` zone file.

```
# grep service /usr/lib/firewalld/zones/public.xml
    <service name="ssh"/>
    <service name="dhcpv6-client"/>
```

- Note that these services correspond with the trusted services for the `public` zone.

e. In the Firewall Configuration **Service** window, select `dhcp` to trust this service.

- The **Service** window appears as shown:



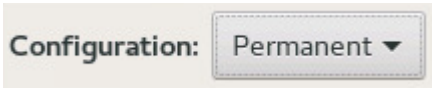
- With the **runtime** configuration selected, this service is trusted immediately by the network connections in the selected zone.
- It is not necessary to restart the `firewalld` service and disrupt the existing network connections and services.

f. Use the `grep` command to search for the string "`service`" in the `/etc/firewalld/zones/public.xml` zone file.

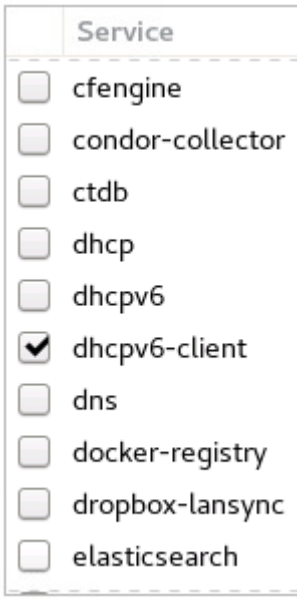
```
# grep service /etc/firewalld/zones/public.xml
    <service name="ssh"/>
    <service name="dhcpv6-client"/>
```

- Note that this file was not updated. There is no entry for the `dhcp` service.
- Configuration files are not updated in **runtime** configuration mode.
- Configuration changes made in **runtime** configuration mode are lost when the `firewalld` service is restarted.

g. From the Firewall Configuration GUI, select the **Permanent** configuration option.

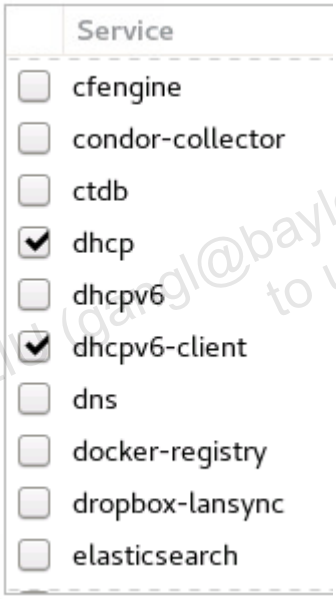


- Note that in the **Service** window, the `dhcp` service is no longer trusted.



h. In the Firewall Configuration **Service** window, select `dhcp` to trust this service.

- The **Service** window appears as shown:



- In the lower-left corner of the GUI, the message “Changes applied” appears briefly. This only means that the configuration file was updated. The change does not take effect until the `firewalld` service is restarted.

i. Use the `grep` command to search for the string “`service`” in the `/usr/lib/firewalld/zones/public.xml` zone file.

```
# grep service /usr/lib/firewalld/zones/public.xml
    <service name="ssh"/>
    <service name="dhcpv6-client"/>
```

- Note that there is no entry for the `dhcp` service.

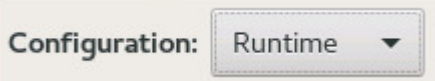
- Configuration files in `/usr/lib/firewalld` are not updated when changes are made.

j. Use the `grep` command to search for the string "`service`" in the `/etc/firewalld/zones/public.xml` zone file.

```
# grep service /etc/firewalld/zones/public.xml
    <service name="ssh"/>
    <service name="dhcpv6-client"/>
    <service name="dhcp"/>
```

- Note that there is an entry for the `dhcp` service.
- The configuration files in `/etc/firewalld` are updated when changes are made.
- In this example, the `dhcp` service is trusted after restarting the `firewalld` service.

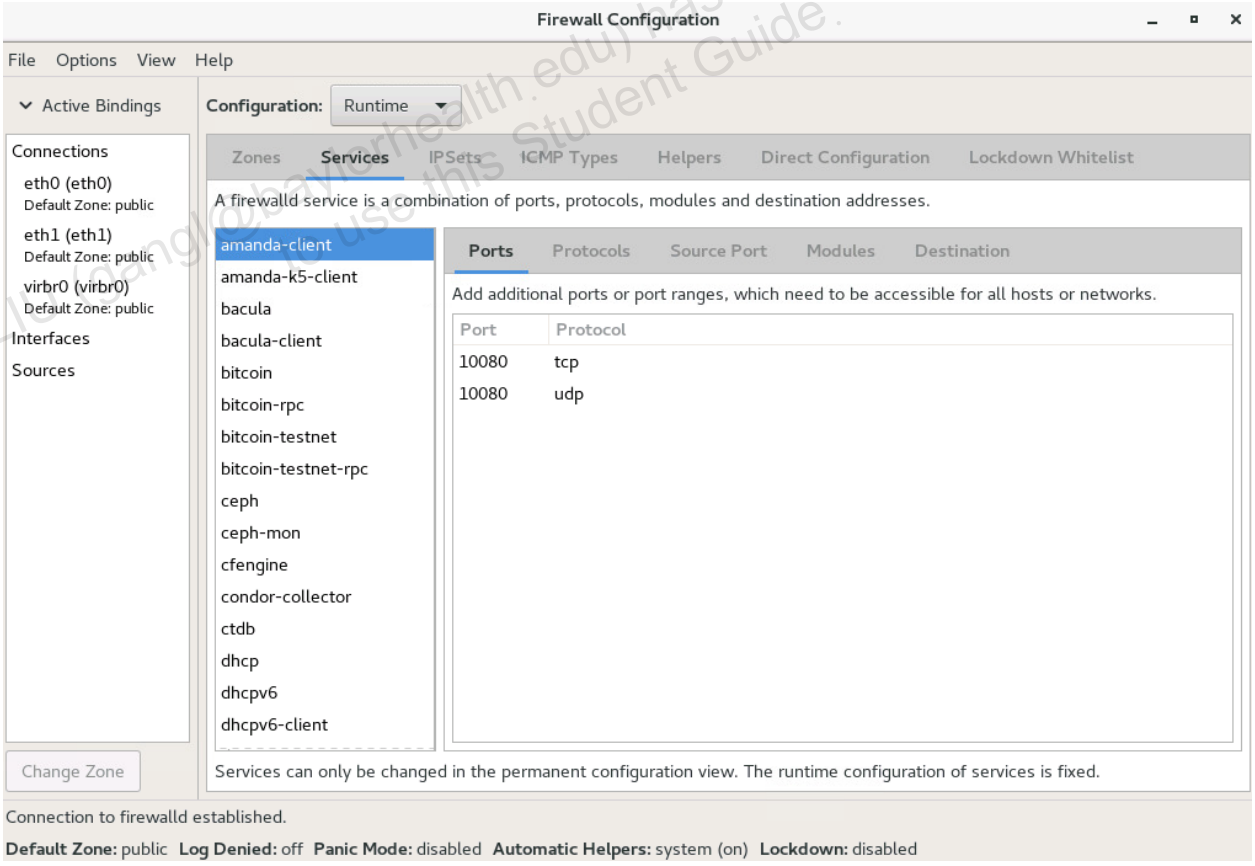
k. Click the **Configuration** drop-down menu and select **Runtime**.



6. Explore `firewalld` services.

a. Select the **Services** tab from the top row of tabs.

- The Firewall Configuration GUI appears as shown:



- Note that there are several predefined services.
- Each of these services is defined by individual files in the `/usr/lib/firewalld/services` directory.

b. Use the `ls` command to display the contents of the `/usr/lib/firewalld/services` directory.

```
# ls /usr/lib/firewalld/services/
amanda-client.xml      high-avail...xml  nfs.xml      sane.xml
amanda-k5-client.xml   https.xml         nrpe.xml     sips.xml
bacula-client.xml      http.xml          ntp.xml      sip.xml
...
```

- Note that there is an XML configuration file for each of the predefined services.

c. Use the `firewall-cmd` command to display the available services.

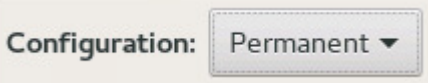
```
# firewall-cmd --get-services
RH-Satellite-6 amanda-client amanda-k5-client bacula bacula-
client bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc
ceph ceph-mon cfengine condor-collector ctdb dhcp dhcpv6 dhcpv6-
client dns ...
```

- Note that this command shows the same predefined services.

d. Use the `cat` command to display the contents of the `/usr/lib/firewalld/services/samba.xml` file.

```
# cat /usr/lib/firewalld/services/samba.xml
...
<service>
  <short>Samba</short>
  <description>This option allows you to access and participate
in Windows file and printer sharing networks. You need the samba
package installed for this option to be useful.</description>
  <port protocol="udp" port="137"/>
  <port protocol="udp" port="138"/>
  <port protocol="tcp" port="139"/>
  <port protocol="tcp" port="445"/>
  <module name="nf_conntrack_netbios_ns"/>
</service>
```

e. From the Firewall Configuration GUI, select the **Permanent** configuration option.



- Services can be changed only in **permanent** configuration mode.

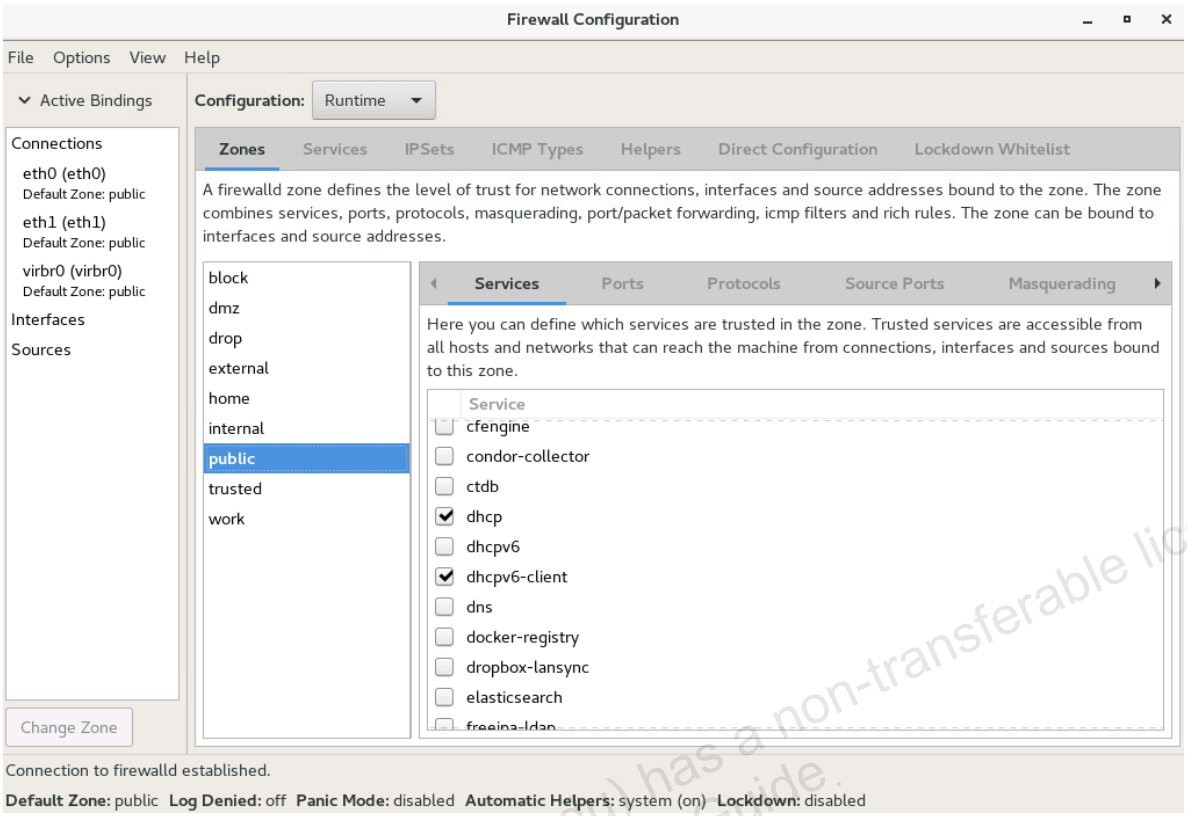
f. From the GUI, note that for each service you can select the lower row of tabs and configure the associated parameters.

- Click the **Ports** tab.
  - Click the **Add** button.
  - Note that you can add, change, or remove ports or port ranges for the selected service.

  - Click the **Cancel** button.
- Click the **Protocols** tab.
  - Click the **Add** button.
  - Note that you can add, change, or remove protocols for the selected service.
  - Click the **Cancel** button.
- Click the **Source Port** tab.
  - Click the **Add** button.
  - Note that you can add, change, or remove source ports or port ranges for the selected service.
  - Click the **Cancel** button.
- Click the **Modules** tab.
  - Click the **Add** button.
  - Note that you can add, change, or remove Netfilter helper modules for the selected service.
  - Click the **Cancel** button.
- Click the **Destination** tab.
  - Note that you can limit traffic to a particular destination address and Internet Protocol (IPv4 or IPv6).

g. To prepare for the next practice, make the following selections from the Firewall Configuration GUI:

- Select the **Runtime** configuration option.
- Select **Zones** from the top row of tabs.
- Select **Services** from the lower row of tabs.
- Select the `public` zone from the list of zones.
- The Firewall Configuration window appears as shown:



Practice 16-4: Configuring `firewalld`

Overview

In this practice, you create a `firewalld` rule to trust NFS.

- You configure **host03** as an NFS server and attempt mounts on NFS client **host01**.
- You create `firewalld` rules to trust NFS so that the NFS client can mount the exported file system with the `firewalld` service running.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- You are logged in to **host01** by using the `ssh` command.
- You are logged in to **host03** by using the `xm vncviewer host03&` command.
- You are the `root` user on **host01** and **host03**.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. Configure **host03** as an NFS server.
   a. Use the `rpm` command to verify that the `rpcbind` package is installed.

```
[host03]# rpm -q rpcbind
rpcbind-...
```

   - In this example, the package is already installed.

   b. If `rpcbind` is not installed, use `yum` to install it.

```
[host03]# yum install rpcbind
```

   c. Use the `systemctl` command to verify whether the `rpcbind` service is running.

```
[host03]# systemctl status rpcbind
● rpcbind.service - RPC bind service
   Loaded: loaded (/usr/lib/systemd/system/rpcbind.service;
enabled; vendor preset: enabled)
   Active: active (running) since <date_time; time> ago
 Main PID: 574 (rpcbind)
   CGroup: /system.slice/rpcbind.service
           └─574 /sbin/rpcbind -w

<date_time> host03.example.com systemd[1]: Starting RPC bind
service...
<date_time> host03.example.com systemd[1]: Started RPC bind
service.
```

   - In this example, the `rpcbind` service is `active (running)`. If your system shows the service as `inactive (dead)`, follow the next step to start it. If your system

shows the `rpcbind` service as `active (running)`, as in this example, skip to the following step, checking installation of `nfs-utils` with `rpm`: "`rpm -q nfs-utils`".

d. Start the `rpcbind` service if it is not started.

```
[host03] # systemctl start rpcbind
```

e. Verify that the `rpcbind` service is `active (running)`.

```
[host03]# systemctl status rpcbind
● rpcbind.service - RPC bind service
   Loaded: loaded (/usr/lib/systemd/system/rpcbind.service;
enabled; vendor preset: enabled)
   Active: active (running) since <date_time; time> ago
  Process: 13855 ExecStart=/sbin/rpcbind -w $RPCBIND_ARGS
(code=exited, status=0/SUCCESS)
 Main PID: 13857 (rpcbind)
   CGroup: /system.slice/rpcbind.service
           └─13857 /sbin/rpcbind -w

<date_time> host02.example.com systemd[1]: Starting RPC bind
service...
<date_time> host02.example.com systemd[1]: Started RPC bind
service.
```

- Note that the `rpcbind` service is now `active (running)`.

f. Use the `rpm` command to verify that the `nfs-utils` package is installed.

```
[host03]# rpm -q nfs-utils
nfs-utils-...
```

- In this example, the package is installed.

g. If `nfs-utils` is not installed, use `yum` to install it.

```
# yum install nfs-utils
```

h. Use the `systemctl` command to verify whether the `nfs` service is started.

```
[host03]# systemctl status nfs
● nfs-server.service - NFS server and services
   Loaded: loaded (/usr/lib/systemd/system/nfs-server.service;
disabled; vendor preset: disabled)
   Active: inactive (dead)
```

- In this example, the `nfs-server` service is `disabled` and not running.

i. Use the `systemctl` command to start the `nfs` service and associated services.

```
[host03]# systemctl start nfs
```

j. Use the `systemctl` command to enable the `nfs-server` service to start at boot time.

```
[host03]# systemctl enable nfs
```

```
Created symlink from /etc/systemd/system/multi-
user.target.wants/nfs-server.service to
/usr/lib/systemd/system/nfs-server.service.
```

k. Use the `systemctl` command to verify that the `nfslock` service is started.

```
[host03]# systemctl status nfslock
● rpc-statd.service - NFS status monitor for NFSv2/3 locking.
   Loaded: loaded (/usr/lib/systemd/system/rpc-statd.service;
static; vendor preset: disabled)
   Active: active (running) since <date_time; time> ago
 Main PID: 3464 (rpc.statd)
   CGroup: /system.slice/rpc-statd.service
           └─3464 /usr/sbin/rpc.statd
...
```

- In this example, the `rpc-statd - NFS status monitor` service is running.

l. If the `rpc-statd` is not `active (running)`, start `nfslock`.

```
# systemctl start nfslock
```

2. Create and export a file system on **host03**.

a. On **host03**, use the `mkfs` command to make an `ext4` file system on `/dev/xvdb`.

```
[host03]# mkfs -t ext4 /dev/xvdb
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
327680 inodes, 1310720 blocks
65536 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1342177280
40 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

b. Use the `mkdir` command to create the `/Dev` mount point.

```
[host03]# mkdir /Dev
```

c. Use the `mount` command to mount `/dev/xvdb` on `/Dev`.

```
[host03]# mount /dev/xvdb /Dev
```

d. Use the `exportfs` command to export `/Dev` to all clients with the `rw` and `no_root_squash` options.

```
[host03]# exportfs -o rw,no_root_squash *:/Dev
```

e. Use the `cat` command to view the `/var/lib/nfs/etab` file on **host03**.

```
[host03]# cat /var/lib/nfs/etab
/Dev *(rw,sync,wdelay,hide,nocrossmnt,secure,no_root_squash ...
```

- The `/Dev` file system is exported to all client systems.
- The `rw` option allows client systems to make changes to the file system.
- The `no_root_squash` option allows `root` users on client systems to retain `root` privileges on the file system.

f. Use the `showmount -e` command to display exported file systems.

```
[host03]# showmount -e
Export list for host03.example.com:
/Dev *
```

- This confirms that the `/Dev` file system is exported to all client systems.

3. Attempt to mount the exported `/Dev` file system on **host01**.

a. On **host01**, use the `mkdir` command to create the `/remote_dev` directory.

```
[host01]# mkdir /remote_dev
```

b. Use the `mount` command to mount the exported file system from **host03**, `/Dev`, with `rw` and `nosuid` options on the local mountpoint, `/remote_dev`.

- The `rw` option mounts the file system with read/write permissions.
- The `nosuid` option does not allow `setuid` or `setgid` bits to take effect.

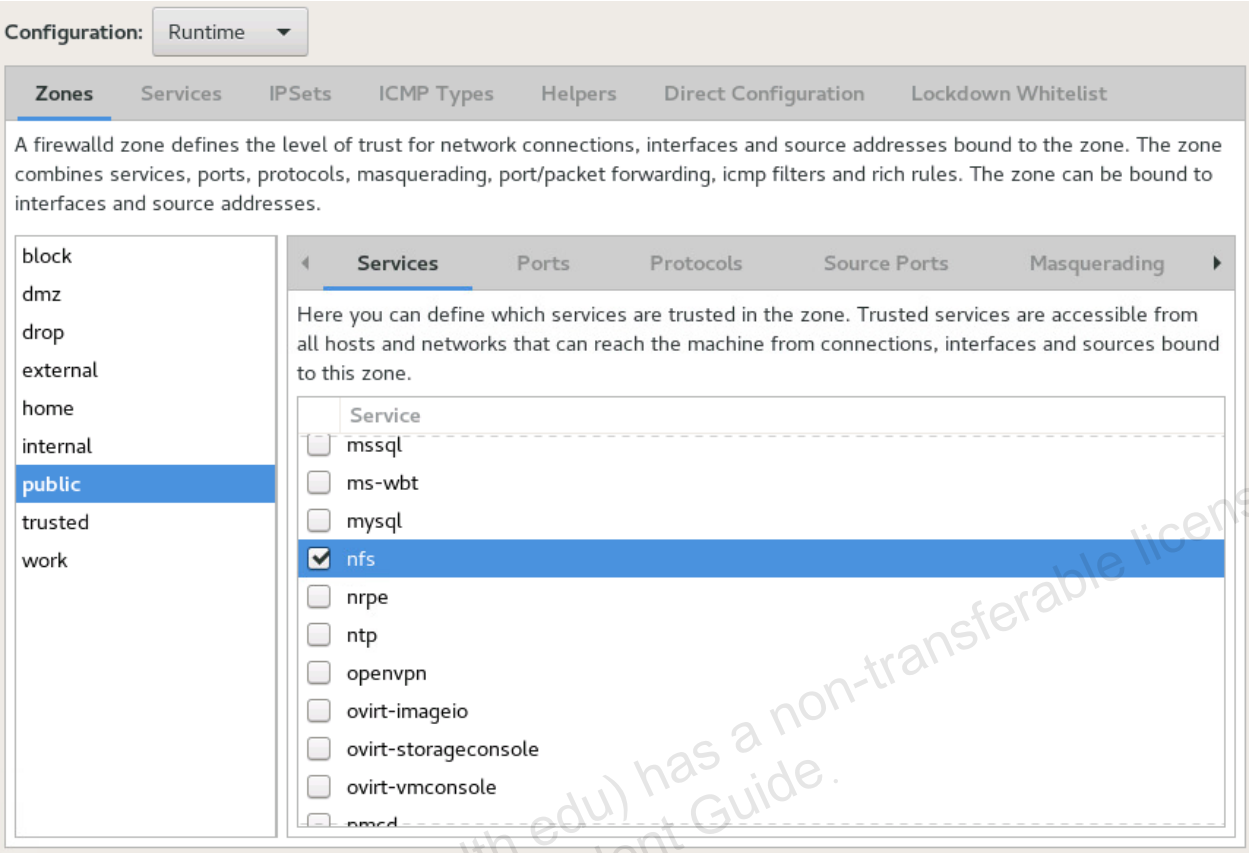
```
[host01]# mount –t nfs –o rw,nosuid host03:/Dev /remote_dev
mount.nfs: No route to host
```

- In this example, the `mount` command fails because of the firewall on **host03**.
- It takes time for the `mount` command to time out. Rather than wait, you can press Ctrl + C to abort the `mount` command.

4. From **host03**, use the Firewall Configuration GUI to trust the `nfs` service.

a. In the Firewall Configuration **Service** window, scroll down if necessary and select `nfs` to trust this service.

- The **Service** window appears as shown:



- In **runtime** configuration mode, this change takes effect immediately.

5. Attempt to mount the exported `/Dev` file system on **host01**.

   a. Re-issue the `mount` command from the previous step on **host01**.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

   - The `mount` command is successful.

   b. On **host01**, use the `df` command to display the mounted file systems.

```
[host01]# df -h
Filesystem      Size  Used  Avail  Use%  Mounted on
...
/dev/xvda2       20G  2.4G    17G   13%  /
/dev/xvda3      2.0G   11M   1.8G    1%  /home
/dev/xvda1      976M  135M   774M   15%  /boot
...
host03:/Dev     4.8G   20M   4.6G    1% /remote_dev
```

   - Note that the `host03:/Dev` file system is mounted on the local file system `/remote_dev`.

6. Unmount the exported file system on **host01**.

   a. Use the `umount` command to unmount `/remote_dev` on **host01**.

```
[host01]# umount /remote_dev
```

b. On **host01**, use the `df` command to display the mounted file systems.

```
[host01]# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/xvda2      20G  2.4G    17G   13%  /
/dev/xvda3     2.0G   11M   1.8G    1%  /home
/dev/xvda1     976M  135M   774M   15%  /boot
...
```

- Note that the `host03:/Dev` file system is no longer mounted on the local file system `/remote_dev`.

7. From **host03**, use the `firewall-cmd` command to permanently trust the `nfs` service.

a. From a command prompt on **host03**, use the `firewall-cmd` command to permanently trust the `nfs` service.

```
[host03]# firewall-cmd --permanent --zone=public --add-
service=nfs
success
```

b. Use the `systemctl` command to restart the `firewalld` service on **host03**.

```
[host03]# systemctl restart firewalld
```

8. Attempt to mount the exported `/Dev` file system on **host01**.

a. Re-issue the `mount` command from the preceding step on **host01**.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

- The `mount` command is successful.

b. On **host01**, use the `df` command to display the mounted file systems.

```
[host01]# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/xvda2      20G  2.4G    17G   13%  /
/dev/xvda3     2.0G   11M   1.8G    1%  /home
/dev/xvda1     976M  135M   774M   15%  /boot
...
host03:/Dev    4.8G   20M   4.6G    1% /remote_dev
```

- Note that the `host03:/Dev` file system is mounted on the local file system `/remote_dev`.

9. Restore initial configuration in preparation for the next practice.

a. Use the `umount` command to unmount `/remote_dev` on **host01**.

```
[host01]# umount /remote_dev
```

b. From **host03**, use the `grep` command to search for the string "`service`" in the `/etc/firewalld/zones/public.xml` file.

```
[host03]# grep service /etc/firewalld/zones/public.xml
```

```
  <service name="ssh"/>
  <service name="dhcpv6-client"/>
  <service name="dhcp"/>
  <service name="nfs"/>
```

c. Remove the "dhcp" and "nfs" service entries from the /etc/firewalld/zones/public.xml file.

- You can use the vi editor to edit the file and delete the entries, or you can use the firewall-cmd command as shown.

```
[host03]# firewall-cmd --permanent --zone=public --remove-
service=nfs
success
[host03]# firewall-cmd --permanent --zone=public --remove-
service=dhcp
success
```

d. From **host03**, use the grep command to search for the "service" string in the /etc/firewalld/zones/public.xml file.

```
[host03]# grep service /etc/firewalld/zones/public.xml
  <service name="ssh"/>
  <service name="dhcpv6-client"/>
```

- Note that the entries for nfs and dhcp no longer exist.

e. Use the systemctl command to restart the firewalld service on **host03**.

```
[host03]# systemctl restart firewalld
```

f. From the Firewall Configuration GUI on **host03**, view the entries in the **Service** window.

- Note that the only trusted services are dhcpv6-client and ssh.

g. Quit the Firewall Configuration GUI on **host03** by selecting **File->Quit** from the menu bar.

Practice 16-5: Configuring `iptables`

Overview

In this practice, you use the `iptables` command to allow a client system to mount an NFS file system.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- You are the `root` user on **host01** and **host03**.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. From **host03**, disable and stop the `firewalld` service and start the `iptables` service.
   a. Check the status of the `firewalld` service with the `systemctl` command.

```
[host03]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service;
enabled; vendor preset: enabled)
   Active: active (running) since <date_time; time> ago
     Docs: man:firewalld(1)
 Main PID: 12391 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─12391 /usr/bin/python -Es /usr/sbin/firewalld --
nofork --nopid
...
```

   - You can see that `firewalld` is enabled and running.

   b. Use the `systemctl` command to disable and then stop the `firewalld` service.

```
[host03]# systemctl disable firewalld
Removed symlink /etc/systemd/system/dbus-
org.fedoraproject.FirewallD1.service.
Removed symlink /etc/systemd/system/multi-
user.target.wants/firewalld.service.
[host03]# systemctl stop firewalld
```

   c. Re-check the status of the `firewalld` service with the `systemctl` command.

```
[host03]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service;
disabled; vendor preset: enabled)
   Active: inactive (dead)
     Docs: man:firewalld(1)
```

```
...
```

- You can see that `firewalld` is now disabled and inactive.

d. Use `yum` to install the `iptables-services` package. The `iptables` and `ip6tables` services are in this package.

```
[host03]# yum install iptables-services
...
Transaction Summary
================================================================
Install  1 Package

Total download size: 51 k
Installed size: 25 k
Is this ok [y/d/N]: y
...
Complete!
```

e. Use the `systemctl` command to start the `iptables` service.

```
[host03]# systemctl start iptables
```

f. Use the `iptables -L` command to list all the rules in all the chains.

```
[host03]# iptables -L
Chain INPUT (policy ACCEPT)
target   prot  opt  source     destination
ACCEPT   all   --   anywhere   anywhere        state ...
...
Chain FORWARD (policy ACCEPT)
target   prot  opt  source     destination
REJECT   all   --   anywhere   anywhere        reject...

Chain OUTPUT (policy ACCEPT)
target   prot  opt  source     destination
```

g. Run the `iptables -L` command again, but this time pipe the output to `grep` and search for "`nfs`".

```
[host03]# iptables -L | grep nfs
```

- Note that there are currently no rules containing the "`nfs`" string.

2. Attempt to mount the NFS exported file system on **host03** from a remote host, **host01**.

a. From **host03**, use the `showmount -e` command to display exported file systems.

```
[host03]# showmount -e
Export list for host03.example.com:
/Dev *
```

- This confirms that the `/Dev` file system is exported to all client systems.

b. From **host01**, use the `mount` command to mount the exported `/Dev` NFS file system from **host03**.

- Mount with `rw` and `nosuid` options on the local mountpoint, `/remote_dev`.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
mount.nfs: No route to host
```

- With the `iptables` service enabled, and no firewall rule to trust `nfs`, the `mount` command fails.
- It takes time for the `mount` command to time out. Rather than wait, you can press Ctrl + C to abort the `mount` command.

3. From **host03**, view and modify the `iptable` rules.

a. Run `iptables -h` to display all options.

```
[host03]# iptables -h
iptables v1.4.21

Usage: iptables -[ACD] chain rule-specification [options]
       iptables -I chain [rulenum] rule-specification ...
...
  --line-numbers       print line numbers when listing
...
```

- From the help, note that the `--line-numbers` option displays line numbers.

b. Run the `iptables` command to list only those rules in the `INPUT` chain and include line numbers.

```
[host03]# iptables -L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source     destination
1    ACCEPT   all   --   anywhere   anywhere     state ...
...
4     ACCEPT   tcp   --   anywhere   anywhere     state ...ssh
...
```

- Note that line number 4 contains a rule to accept `ssh` traffic.
- You must create a similar entry to accept `nfs` traffic.

c. Use the `iptables` command to insert the "`nfs`" rule before line 4 with the following characteristics from the command line:

- Chain = `INPUT`
- Protocol = `tcp`
- State = `NEW`
- Destination port = `nfs`
- Target = `ACCEPT`

```
[host03]# iptables –I INPUT 4 –p tcp –m state --state NEW --
dport nfs –j ACCEPT
```

- This rule accepts incoming `tcp` traffic for `nfs`.

d. Repeat the preceding step to list the new `nfs` rule.

```
 [host03]# iptables –L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source     destination
1    ACCEPT   all   --   anywhere   anywhere     state ...
...
4    ACCEPT   tcp   --   anywhere   anywhere     state ...nfs
5    ACCEPT   tcp   --   anywhere   anywhere     state ...ssh
...
```

- Note that there now is a rule to accept `nfs` traffic.

e. Use the `cat` command to view the `/etc/sysconfig/iptables` file.

```
[host03]# cat /etc/sysconfig/iptables
...
-A INPUT –p tcp –m state --state NEW –m tcp --dport 22 –j ACCEPT
...
```

- Note that there is a rule for `ssh` (port `22`) but not for `nfs`.
- You still need to save the `iptables` rules to the `/etc/sysconfig/iptables` file.

f. Use the `service` command to save the `iptables` rules.

```
[host03]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables...
```

g. Use the `cat` command to view the `/etc/sysconfig/iptables` file.

```
[host03]# cat /etc/sysconfig/iptables
...
-A INPUT –p tcp –m state --state NEW -m tcp --dport 2049 –j
ACCEPT
...
```

- Note that the new rule has the actual port number, `2049`, for `nfs`.

h. Use the `grep` command to search for `2049` in the `/etc/services` file.

```
[host03]# grep 2049 /etc/services
nfs  2049/tcp ...
...
```

i. Use the `systemctl` command to restart the `iptables` service.

```
[host03]# systemctl restart iptables
```

4. From **host01**, attempt to mount the NFS file system.

a. Use the `mount` command to mount `host03:/Dev` on `/remote_dev`.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

- The `mount` command is successful this time.

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

```
[host01]# df -h
Filesystem      Size  Used  Avail  Use%  Mounted on
...
/dev/xvda2       20G  2.4G    17G   13%  /
/dev/xvda3      2.0G   11M   1.8G    1%  /home
/dev/xvda1      976M  135M   774M   15%  /boot
...
host03:/Dev     4.8G   20M   4.6G    1% /remote_dev
```

- Note that the `host03:/Dev` file system is mounted on the local file system `/remote_dev`.

c. From **host01**, use the `umount` command to unmount `/remote_dev`.

```
[host01]# umount /remote_dev
```

Practice 16-6: Configuring a TCP Wrapper

Overview

In this practice, you configure a TCP wrapper to deny one system from using OpenSSH utilities to connect to another system. You also create a custom log file to capture connection attempts that are denied.

Assumptions

- This practice is performed on the **host01** and **host03** VMs.
- You are the `root` user on **host01** and **host03**.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. From **host01**, confirm that you can use the `ssh` command to connect to **host03**.
   - If you get messages about the "`authenticity of host...`" reply with "`yes`."
   - Provide the password when prompted.
   - Use the `logout` command to log off after confirming that you can connect.

```
[host01]# ssh host03
The authenticity of host 'host03 (192.0.2.103)' can't be
established.
ECDSA key fingerprint is SHA ...
ECDSA key fingerprint is MD5 ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host03,192.0.2.103' (ECDSA) to the
list of known hosts.
root@host03's password:
Last login: ...

[root@host03 ~]# logout
Connection to host03 closed.
[host01]#
```

2. On **host03**, configure a TCP wrapper to deny **host01** from using OpenSSH utilities to connect.
   a. Use the `vi` editor to edit `/etc/hosts.deny` and add the following entry.

```
[host03]# vi /etc/hosts.deny
sshd : 192.0.2.101
```

   - This entry denies **host01** (`192.0.2.101`) from using the OpenSSH utilities to connect to **host03**.

   b. From **host01**, attempt to use the `ssh` command to connect to **host03**.

```
[host01]# ssh host03
ssh_exchange_identification: read: Connection reset by peer
```

- This time you are denied a connection.

3. On **host03**, modify the TCP wrapper to write a message to a log file.
   a. Use the `vi` editor to edit `/etc/hosts.deny` and modify the entry as follows.

```
[host03]# vi /etc/hosts.deny
sshd : 192.0.2.101 : spawn /bin/echo "%c tried to connect to %d
and was blocked." >> /var/log/tcpwrappers.log
```

   b. From **host01**, attempt to use the `ssh` command to connect to **host03**.

```
[host01]# ssh host03
ssh_exchange_identification: read: Connection reset by peer
```

   - You are still denied a connection, and a message is written to a log file.

   c. On **host03**, use the `cat` command to view the `/var/log/tcpwrappers.log` file.

```
[host03]# cat /var/log/tcpwrappers.log
192.0.2.101 tried to connect to sshd and was blocked.
```

4. Reverse changes to `/etc/hosts.deny`.
   a. On **host03**, use the `vi` editor to edit `/etc/hosts.deny` and delete the entry created earlier in this practice.
   - Delete the strikethrough line as follows:

```
[host03]# vi /etc/hosts.deny
~~sshd : 192.0.2.101 : spawn /bin/echo "%c tried to connect to %d~~
~~and was blocked." >> /var/log/tcpwrappers.log~~
```

   b. From **host01**, confirm that you can use the `ssh` command to connect to **host03**.
   - Provide the password when prompted.
   - Use the `logout` command to log off after confirming that you can connect.

```
[host01]# ssh host03
root@host03's password:
Last login: ...
[root@host03 ~]# hostname
host03.example.com
[root@host03 ~]# logout
Connection to host03 closed.
[host01]#
```

Practice 16-7: Restoring VM Configurations

1. Perform the following tasks on **host01**.
   a. Use the `systemctl` command to stop the `autofs` service.

```
[host01]# systemctl stop autofs
```

   b. Use the `yum` command to remove the `ftp` package.

   - Answer `y` to "Is this ok."

```
[host01]# yum remove ftp
...
Transaction Summary
================================================================
Remove        1 Package

Installed size: 96 k
Is this ok [y/N]: y
...
Complete!
```

   c. Remove the `/remote_dev` directory. Use the `cd` command to ensure you are in the `root` user's home directory first.

```
[host01]# cd
[host01]# rmdir /remote_dev
```

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

```
[host01]# exit
Connection to host01 closed.
```

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

2. Perform the following tasks on **host03**.
   a. Stop the `iptables` service.

```
[host03]# systemctl stop iptables
```

   b. Start and enable the `firewalld` service.

```
[host03]# systemctl start firewalld
[host03]# systemctl enable firewalld
Created symlink from /etc/systemd/system/dbus-
org.fedoraproject.FirewallD1.service to
/usr/lib/systemd/system/firewalld.service.
Created symlink from /etc/systemd/system/multi-
user.target.wants/firewalld.service to
/usr/lib/systemd/system/firewalld.service.
```

   c. Disable and stop the `vsftpd` service.

```
[host03]# systemctl disable vsftpd
Removed symlink /etc/systemd/system/multi-
user.target.wants/vsftpd.service.
[host03]# systemctl stop vsftpd
```

d. Disable and stop the `nfs` service.

```
[host03]# systemctl disable nfs
Removed symlink /etc/systemd/system/multi-user.target.wants/nfs-
server.service.
[host03 ~]# systemctl stop nfs
```

e. Unmount `/Dev` and remove the directory. Use the `cd` command to ensure you are in the `root` home directory first.

```
[host03]# cd
[host03]# umount /Dev
[host03]# rmdir /Dev
```

f. Remove `/var/ftp/pub/test_file`.

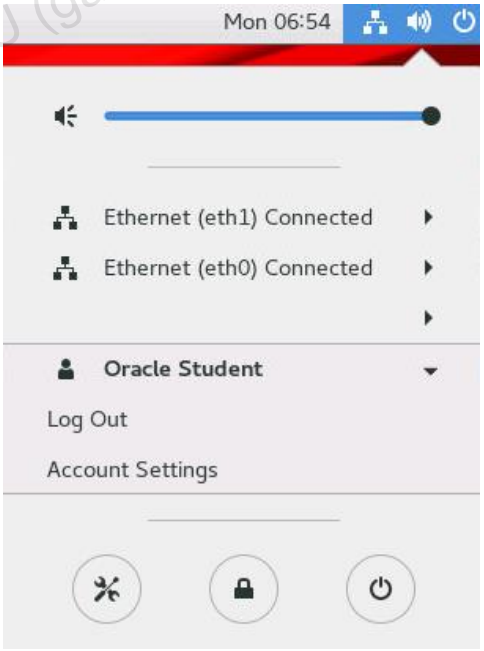
```
[host03]# rm /var/ftp/pub/test_file
rm: remove regular file '/var/ftp/pub/test_file'? y
```

g. Remove the `/home/oracle/jail` directory structure.

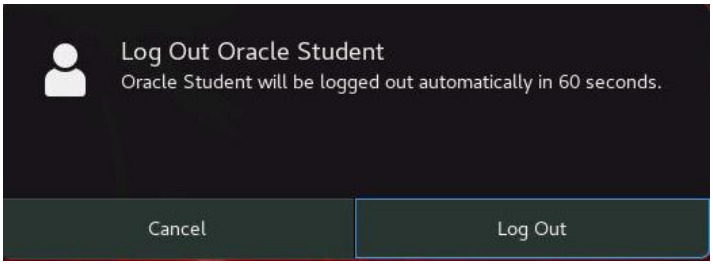
```
[host03]# cd /home/oracle
[host03]# /bin/rm -r jail
```

h. Log off from **host03**.

1) Click the Power icon in the upper right of the GNOME screen and select Oracle Student.



2) Click Log Out. The following pop-up message appears:



3) Click Log Out.
4) Click the **X** in the top-right corner of the GNOME login window to close the window.

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

Practices for Lesson 17: Oracle on Oracle

Practices for Lesson 17: Overview

Practices Overview

In these practices, you install and run the Oracle Database Pre-Install RPM for Oracle Linux 7. You also configure ASMLib.

Assumptions

You created a local Yum repository in the practices for the lesson titled “Package Management.”

Practice 17-1: Using `scp` to Upload `oracle` Packages

Overview

In this practice, you use the `scp` command to upload the `oracle-database-preinstall-18c` package and the `oracleasmlib` package from **dom0** to the **host03** VM.

Normally, you obtain these packages from ULN or, in the case of the `oracle-database-preinstall-18c` package, also from the Oracle Linux yum server. The `oracleasmlib` package is not available on the Oracle Linux yum server but, in addition to ULN, can be downloaded from the Oracle Technology Network here: https://www.oracle.com/technetwork/server-storage/linux/asmlib/ol7-2352094.html.

Assumptions

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

Tasks

1. Use the `scp` command to transfer the `oracle` packages from **dom0** to **host03**.
   a. From **dom0**, use the `cd` command to change to the `/OVS/seed_pool/sfws` directory.

```
[dom0]# cd /OVS/seed_pool/sfws
```

   b. Use the `ls` command to view the directory for the `oracle*` package.

```
[dom0]# ls oracle*
oracleasmlib-2.0.12-1.el7.x86_64.rpm
oracle-database-preinstall-18c-1.0-1.el7.x86_64.rpm
```

   c. Use the `scp` command to transfer these packages to the `root` user's home directory on **host03**.

```
[dom0]# scp oracle* host03:~
root@host03's password:
oracleasmlib-2.0.12-1.el7.x86_64.rpm ...
oracle-database-preinstall-18c-1.0-1.el7.x86_64.rpm ...
```

Practice 17-2: Installing and Running Oracle Database Pre-Install

Overview

In this practice, you install the `oracle-database-preinstall-18c` package and view the results.

Assumptions

You are logged in to **dom0** as the `root` user.

Tasks

1. Log in to **host03**.

   Use the `ssh` command to log in to **host03**. Provide the `root` password when prompted.

```
[dom0]# ssh host03
root@host03's password:
Last login: ...
```

All remaining commands in this practice are entered from **host03**.

2. Install the `oracle-database-preinstall-18c` package on **host03**.

   a. Use the `ls` command to confirm that the `oracle` packages are in the `root` user's home directory.

```
# ls oracle*
oracleasmlib-2.0.12-1.el7.x86_64.rpm
oracle-database-preinstall-18c-1.0-1.el7.x86_64.rpm
```

   b. Use the `yum` command to install the `oracle-database-preinstall-18c` package.

   - Answer **y** when prompted.

```
# yum install oracle-database-preinstall-18c-1.0-
1.el7.x86_64.rpm
...
Installing:
 oracle-database-preinstall-18c      x86_64 ...
Installing for dependencies:
...
Transaction Summary
================================================================
Install  1 Package (+5 Dependent packages)

Total size: 2.6 M
Total download size: 2.6 M
Installed size: 12 M
Is this ok [y/d/N]: y
```

```
Downloading packages:
...
Installed:
  oracle-database-preinstall-18c.x86_64 ...

Dependency Installed:
...

Complete!
```

3. View the results of the installation of the `oracle-database-preinstall-18c` package.
   a. Use the `rpm -ql` command to list the files provided by the `oracle-database-preinstall-18c` package.

```
# rpm -ql oracle-database-preinstall-18c
/etc/rc.d/init.d/oracle-database-preinstall-18c-firstboot
/etc/security/limits.d/oracle-database-preinstall-18c.conf
/etc/sysconfig/oracle-database-preinstall-18c
/etc/sysconfig/oracle-database-preinstall-18c/oracle-database-preinstall-18c-verify
/etc/sysconfig/oracle-database-preinstall-18c/oracle-database-preinstall-18c.param
/usr/bin/oracle-database-preinstall-18c-verify
/var/log/oracle-database-preinstall-18c
/var/log/oracle-database-preinstall-18c/results
```

   b. Use the `less` or `cat` command to view the contents of the `/etc/security/limits.d/oracle-database-preinstall-18c.conf` file.

```
# cat /etc/security/limits.d/oracle-database-preinstall-18c.conf
# oracle-database-preinstall-18c setting for nofile soft limit is 1024
oracle   soft   nofile    1024

# oracle-database-preinstall-18c setting for nofile hard limit is 65536
oracle   hard   nofile    65536

# oracle-database-preinstall-18c setting for nproc soft limit is 16384
# refer orabug15971421 for more info.
oracle   soft   nproc    16384

# oracle-database-preinstall-18c setting for nproc hard limit is 16384
oracle   hard   nproc    16384
```

```
# oracle-database-preinstall-18c setting for stack soft limit is
10240KB
oracle   soft   stack    10240

# oracle-database-preinstall-18c setting for stack hard limit is
32768KB
oracle   hard   stack    32768

# oracle-database-preinstall-18c setting for memlock hard limit
is maximum of 128GB on x86_64 or 3GB on x86 OR 90 % of RAM
oracle   hard   memlock    134217728

# oracle-database-preinstall-18c setting for memlock soft limit
is maximum of 128GB on x86_64 or 3GB on x86 OR 90% of RAM
oracle   soft   memlock    134217728
```

- Note that the shell resource limits for `oracle` are defined in this file.

c. Use the `ls -l` command to list the contents of the `/etc/security/limits.d` directory.

```
# ls -l /etc/security/limits.d
-rw-r--r--. 1 root root ... 20-nproc.conf
-rw-r--r--. 1 root root ... oracle-database-preinstall-18c.conf
```

- Note that the date for the `oracle-database-preinstall-18c.conf` file is current.

d. Use the `ls -l` command to display the file type of the `/usr/bin/oracle-database-preinstall-18c-verify` file.

```
# ls -l /usr/bin/oracle-database-preinstall-18c-verify
lrwxrwxrwx. ... /usr/bin/oracle-database-preinstall-18c-verify -
> /etc/sysconfig/oracle-database-preinstall-18c/oracle-database-
preinstall-18c-verify
```

- Note that this file is a symbolic link to the `/etc/sysconfig/oracle-database-preinstall-18c/oracle-database-preinstall-18c-verify` file.

e. Change to the `/etc/sysconfig/oracle-database-preinstall-18c` directory and view the contents of the directory.

```
# cd /etc/sysconfig/oracle-database-preinstall-18c
# ls
oracle-database-preinstall-18c.conf
oracle-database-preinstall-18c.param
oracle-database-preinstall-18c-verify
```

f. Use the `cat` command to view the contents of the `oracle-database-preinstall-18c.conf` file.

```
# cat oracle-database-preinstall-18c.conf
```

```
RUN_ORACLE_DATABASE_PREINSTALL_18C=NO
```

- Note that the value of this parameter is `NO`, meaning do not run the preinstall script again.
- The preinstall script is run automatically when you install the `oracle-database-preinstall-18c` package.

g. Use the `less` or `cat` command to view the contents of the `oracle-database-preinstall-18c.param` file.

```
# cat oracle-database-preinstall-18c.param
# Oracle-database-18c Parameter config file
#Kernel parameters
#category:arch:kernel:distribution:parameter:recommended-value

kernelcomment:
kernelcomment:oracle-database-preinstall-18c setting for
fs.file-max is 6815744
kernel:*:*:*:fs.file-max:6815744
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
kernel.sem is '250 32000 100 128'
kernel:*:*:*:kernel.sem:250 32000 100 128
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
kernel.shmmni is 4096
kernel:*:*:*:kernel.shmmni:4096
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
kernel.shmall is 1073741824 on x86_64
kernel:*:*:*:kernel.shmall:1073741824
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
kernel.shmmax is 4398046511104 on x86_64
kernel:*:*:*:kernel.shmmax:4398046511104
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
kernel.panic_on_oops is 1 per Orabug 19212317
kernel:*:*:*:kernel.panic_on_oops:1
kernelcomment:
```

```
kernelcomment:oracle-database-preinstall-18c setting for
net.core.rmem_default is 262144
kernel:*:*:*:net.core.rmem_default:262144
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
net.core.rmem_max is 4194304
kernel:*:*:*:net.core.rmem_max:4194304
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
net.core.wmem_default is 262144
kernel:*:*:*:net.core.wmem_default:262144
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
net.core.wmem_max is 1048576
kernel:*:*:*:net.core.wmem_max:1048576
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
net.ipv4.conf.all.rp_filter is 2
kernel:*:*:*:net.ipv4.conf.all.rp_filter:2
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
net.ipv4.conf.default.rp_filter is 2
kernel:*:*:*:net.ipv4.conf.default.rp_filter:2
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for fs.aio-
max-nr is 1048576
kernel:*:*:*:fs.aio-max-nr:1048576
kernelcomment:

kernelcomment:oracle-database-preinstall-18c setting for
net.ipv4.ip_local_port_range is 9000 65500
kernel:*:*:*:net.ipv4.ip_local_port_range:9000 65500
kernelcomment:



#Oracle OS User limits (domain=oracle/*)
#category:arch:kernel:distribution:type:item:recommended-value
```

```
usercomment:
usercomment:*:oracle-database-preinstall-18c setting for nofile
soft limit is 1024
user:*:*:*:soft:nofile:1024

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for nofile
hard limit is 65536
user:*:*:*:hard:nofile:65536

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for nproc
soft limit is 16384
usercomment:*:refer orabug15971421 for more info.
user:*:*:*:soft:nproc:16384

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for nproc
hard limit is 16384
user:*:*:*:hard:nproc:16384

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for stack
soft limit is 10240KB
user:*:*:*:soft:stack:10240

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for stack
hard limit is 32768KB
user:*:*:*:hard:stack:32768

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for memlock
hard limit is maximum of 128GB on x86_64 or 3GB on x86 OR 90 %
of RAM
user:*:*:*:hard:memlock:134217728

usercomment:
usercomment:*:oracle-database-preinstall-18c setting for memlock
soft limit is maximum of 128GB on x86_64 or 3GB on x86 OR 90% of
RAM
user:*:*:*:soft:memlock:134217728

#Kernel boot parameters
```

```
#category:arch:processor:kernel:distribution:parameter:recommend
ed-value
boot:x86_64:*:*:*:numa:off
boot:*:*:*:*:transparent_hugepage:never

#Group Names and IDs
#category:arch:processor:kernel:distribution:parameter:recommend
ed-value
group:*:*:*:*:oinstall:54321
group:*:*:*:*:dba:54322
group:*:*:*:*:oper:54323
group:*:*:*:*:backupdba:54324
group:*:*:*:*:dgdba:54325
group:*:*:*:*:kmdba:54326
group:*:*:*:*:racdba:54330

#User name and IDs
#category:arch:processor:kernel:distribution:parameter:recommend
ed-value
username:*:*:*:*:oracle:54321:oinstall,dba,oper,backupdba,dgdba,
kmdba,racdba
```

- Note that this file defines kernel parameters, Oracle OS user limits, kernel boot parameters, group names and IDs, and usernames and IDs.

h. Use the `less` command to view the contents of the `oracle-database-preinstall-18c-verify` file.

```
# less oracle-database-preinstall-18c-verify
#!/bin/bash
...
# Name: oracle-database-preinstall-18c-verify
# Description: A script to verify and set Oracle Preinstall
settings for 12c
# on Oracle Linux.
...
```

- Note that this file is the bash script that modifies settings.

4. View the modifications made by the `oracle-database-preinstall-18c-verify` script.
   - Note that the `oracle-database-preinstall-18c-verify` script is executed automatically when the `oracle-database-preinstall-18c` package is installed.
   a. Change to the `/var/log/oracle-database-server-12cR2-preinstall` directory and view the contents of the directory.

```
# cd /var/log/oracle-database-preinstall-18c
# ls -l
```

```
drwxr-xr-x. 3 root root ... backup
drwx------. 2 root root ... results
```

- Note that two directories exist in this directory.
- The `backup` directory contains copies of files before they were modified.
- The `results` directory contains the Oracle Database Pre-install log file.

b. Change to the `results` directory and view the contents of the directory.

```
# cd results
# ls
orakernel.log
```

c. Use the `less` command to view the `orakernel.log` (sample output shown).

```
# less orakernel.log
Adding group oinstall with gid 54321
Adding group dba with gid 54322
Adding group oper with gid 54323
Adding group backupdba with gid 54324
Adding group dgdba with gid 54325
Adding group kmdba with gid 54326
Adding group racdba with gid 54330
User oracle - Already exists. Not creating or modifying.
User creation passed

Saving a copy of the initial sysctl.conf
Verifying  kernel parameters as per Oracle recommendations...
Trying to remove instances of  - setting for fs.file-max is
Adding fs.file-max = 6815744
Trying to remove instances of  - setting for kernel.sem is
Adding kernel.sem = 250 32000 100 128
Trying to remove instances of  - setting for kernel.shmmni is
Adding kernel.shmmni = 4096
...
Setting kernel parameters as per oracle recommendations...
Altered file /etc/sysctl.conf
Saved a copy of the current file in /etc/sysctl.d/99-oracle-
database-preinstall-18c-sysctl.conf
Check /etc/sysctl.d for backups
Verification & setting of kernel parameters passed

Setting user limits using /etc/security/limits.d/oracle-
database-preinstall-18c.conf

Verifying oracle user OS limits as per Oracle recommendations...
```

```
Adding oracle soft nofile  1024
Adding oracle hard nofile  65536
Adding oracle soft nproc  16384
...
Setting oracle user OS limits as per Oracle recommendations...
Altered file /etc/security/limits.d/oracle-database-preinstall-18c.conf
Original file backed up at /var/log/oracle-database-preinstall-18c/backup/Oct-12-2018-11-41-21
Verification & setting of user limits passed

Saving a copy of /etc/default/grub file in /etc/default/grub-initial.orabackup
Saving a copy of /etc/default/grub in /var/log/oracle-database-preinstall-18c/backup/Oct-12-2018-11-41-21...
Verifying kernel boot parameters as per Oracle recommendations...
...
Setting kernel boot parameters as per Oracle recommendations...
...
Generating grub configuration file ...
...
done
...
Boot parameters will be effected on next reboot
Altered file /etc/default/grub
Copy of the changed file is in - /etc/default/grub-oracle-database-preinstall-18c.orabackup
Copy of the original file is in - /var/log/oracle-database-preinstall-18c/backup/Oct-12-2018-11-41-21
Verification & setting of boot parameters passed

Trying to add NOZEROCONF parameter...
Taking a backup of existing file to /etc/sysconfig/network.orabackup
Successfully added parameter NOZEROCONF to /etc/sysconfig/network
Setting /etc/sysconfig/network parameters passed

Disabling Transparent Hugepages.
Refer Oracle Note:1557478.1

Disabling defrag.
Refer Oracle Note:1557478.1
```

```
Taking a backup of old config files under /var/log/oracle-
database-preinstall-18c/backup/Oct-12-2018-11-41-21
```

- Note that the required user and groups are created as needed.
- Note that kernel parameters are set and `/etc/sysctl.conf` is backed up beforehand, to `/etc/sysctl.d/99-initial-sysctl.conf`. `/etc/sysctl.conf` is also copied to `/etc/sysctl.d/99-oracle-database-preinstall-18c-sysctl.conf`.
- Note that `oracle` user OS limits are set in `/etc/security/limits.d/oracle-database-preinstall-18c.conf`.
- Note that kernel boot parameters are set and `/etc/default/grub` is backed up beforehand, to `/etc/default/grub-initial.orabackup`.
- While not recorded in the `orakernel.log` file, the `/boot/grub2/grub.cfg` file is also backed up.
- Backup files are also placed in the `backup/<date>` directory.

d. Use the `find` command to list some of the files backed up before settings were changed, `*orabackup`.

```
# find / -name "*orabackup"
/etc/default/grub-oracle-database-preinstall-18c.orabackup
/etc/default/grub-initial.orabackup
/etc/sysconfig/network.orabackup
/boot/grub2/grub.cfg-oracle-database-preinstall-18c.orabackup
```

e. View the contents of the `backup/<date>` directory.

- A `<date>` subdirectory is created, which contains the original files.

```
# cd /var/log/oracle-database-preinstall-18c
# ls backup/<date>
grub  grub.cfg  grubenv.12-10-18-11-41-26  orakernel.log
sysctl.conf
```

- The `backup/<date>/grub` file is the same as the `/etc/default/grub-initial.orabackup` file.
- The `backup/<date>/grub.cfg` file is the same as the `/boot/grub2/grub.cfg-oracle-database-preinstall-18c.orabackup` file.
- The `backup/<date>/sysctl.conf` file is the same as the `/etc/sysctl.d/99-initial-sysctl.conf` file.

f. Use the `diff` command to view the changes made in the `grub.cfg` file.

```
# diff /boot/grub2/grub.cfg /boot/grub2/grub.cfg-oracle-
database-preinstall-18c.orabackup
100c100
<       linux16 /vmlinuz-4.1.12-112.16.4.el7uek.x86_64
root=UUID=638d74ba-d497-4441-90ff-66132424dd86 ro rhgb quiet
numa=off transparent_hugepage=never
---
```

```
>       linux16 /vmlinuz-4.1.12-112.16.4.el7uek.x86_64
root=UUID=638d74ba-d497-4441-90ff-66132424dd86 ro rhgb quiet
115c115
<       linux16 /vmlinuz-3.10.0-862.el7.x86_64
root=UUID=638d74ba-d497-4441-90ff-66132424dd86 ro rhgb quiet
numa=off transparent_hugepage=never
---
>       linux16 /vmlinuz-3.10.0-862.el7.x86_64
root=UUID=638d74ba-d497-4441-90ff-66132424dd86 ro rhgb quiet
129c129
<       linux16 /vmlinuz-0-rescue-
c74c9b01b7084c0b846ab611e2d10e16 root=UUID=638d74ba-d497-4441-
90ff-66132424dd86 ro rhgb quiet numa=off
transparent_hugepage=never
---
>       linux16 /vmlinuz-0-rescue-
c74c9b01b7084c0b846ab611e2d10e16 root=UUID=638d74ba-d497-4441-
90ff-66132424dd86 ro rhgb quiet
```

- Note that the `numa=off` and `transparent_hugepage=never` boot parameters are added to the `grub.cfg` file.

g. Use the `wc -l` command to display the number of lines in the `sysctl.conf` files.

```
# wc -l /etc/sysctl.conf /etc/sysctl.d/99-oracle-database-
preinstall-18c-sysctl.conf /etc/sysctl.d/99-initial-sysctl.conf
  53 /etc/sysctl.conf
  53 /etc/sysctl.d/99-oracle-database-preinstall-18c-sysctl.conf
  10 /etc/sysctl.d/99-initial-sysctl.conf
...
```

- Note that 43 new lines are added to `/etc/sysctl.conf`, and this file is copied to `/etc/sysctl.d/99-oracle-database-preinstall-18c-sysctl.conf` during the preinstall RPM installation. These two files are therefore identical.

h. Use the `diff` command to view the changes made in the `/etc/sysctl.d/99-oracle-database-server-12cR2-preinstall-sysctl.conf` file.

```
# diff /etc/sysctl.d/99-oracle-database-preinstall-18c-
sysctl.conf /etc/sysctl.d/99-initial-sysctl.conf
...
```

i. Use the `diff` command to view the changes made in the `/etc/default/grub` file.

```
# diff /etc/default/grub /etc/default/grub-initial.orabackup
6c6
< GRUB_CMDLINE_LINUX="rhgb quiet numa=off
transparent_hugepage=never"
---
> GRUB_CMDLINE_LINUX="rhgb quiet"
```

- Note that the `numa=off` and `transparent_hugepage=never` boot parameters have been added to the /`etc/default/grub` file.

j. Use the `cat` command to view the limits set in the `/etc/security/limits.d/oracle-database-preinstall-18c.conf` file.

```
# cat /etc/security/limits.d/oracle-database-preinstall-18c.conf
...
```

Practice 17-3: Preparing Disks for ASM Use

Overview

In this practice, you:

- Create one partition using the entire disk on `/dev/xvdb`
- Create one partition using the entire disk on `/dev/xvdd`

Assumptions

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

Tasks

1. Use the `fdisk` command to create a single primary partition on `/dev/xvdb`, using the entire disk as shown.

```
# fdisk /dev/xvdb
...
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): ENTER
Using default response p
Partition number (1-4, default 1): ENTER
First sector (2048-10485759, default 2048): ENTER
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default
10485759): ENTER
Using default value 10485759
Partition 1 of type Linux and of size 5 GiB is set

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

2. Use the `fdisk` command to create a single primary partition on `/dev/xvdd`, using the entire disk as shown.

```
# fdisk /dev/xvdd
...
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
```

```
   e   extended
Select (default p): ENTER
Using default response p
Partition number (1-4, default 1): ENTER
First sector (2048-10485759, default 2048): ENTER
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default
10485759): ENTER
Using default value 10485759
Partition 1 of type Linux and of size 5 GiB is set

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

Practice 17-4 Installing and Configuring ASMLib

Overview

In this practice, you:

- Install the `oracleasmlib` package
- Install the `oracleasm-support` package
- Configure ASMLib
- Load and initialize the ASMLib driver
- Mark disk partitions for ASM use
- View information about the ASM disk partitions

Assumptions

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

Tasks

1. Install the `oracleasm` packages.
   a. Use the `cd` command to change to the `root` user's home directory and then use the `yum` command to install the `oracleasmlib` package. This package was copied from **dom0** previously.
      - Answer **`y`** when prompted.

```
# cd
# yum install oracleasmlib-2.0.12-1.el7.x86_64.rpm
...
Transaction Summary
================================================================
Install  1 Package

Total size: 39 k
Installed size: 39 k
Is this ok [y/d/N] y
Downloading Packages:
...
Installed:
  oracleasmlib.x86_64 0:2.0.12-1.el7

Complete!
```

   b. Use the `yum` command to install the `oracleasm-support` package. Note that this package is obtained from the local repository created from the installation ISO.
      - Answer **`y`** when prompted.

```
# yum install oracleasm-support
...
```

```
Transaction Summary
================================================================
Install  1 Package

Total download size: 85 k
Installed size: 266 k
Is this ok [y/d/N] y
Downloading Packages:
...
Installed:
  oracleasm-support.x86_64 ...

Complete!
```

2. Use the `oracleasm` utility to configure ASMLib.
   a. Run the `oracleasm -h` command to display the usage and commands.

```
# oracleasm -h
Usage: oracleasm [--exec-path=<exec_path>] <command> [ <args> ]
       oracleasm -exec-path
       oracleasm -h
       oracleasm -V

The basic oracleasm commands are:
    configure       Configure the Oracle Linux ASMLib driver
    init            Load and initialize the ASMLib driver
    exit            Stop the ASMLib driver
    scandisks       Scan the system for Oracle ASMLib disks
    status          Display the status of the Oracle ASMLib ...
    listdisks       List known Oracle ASMLib disks
    listiids        List the iid files
    deleteiids      Delete the unused iid files
    querydisk       Determine if a disk belongs to Oracle AS...
    createdisk      Allocate a device for Oracle ASMLib use
    deletedisk      Return a device to the operating system
    renamedisk      Change the label of an Oracle ASMLib disk
    update-driver   Download the latest ASMLib driver
```

   b. Use the `oracleasm configure -i` command to configure the ASMLib driver.

```
# oracleasm configure -i
Configuring the Oracle ASM library driver.

This will configure the on-boot properties of the Oracle ASM
library driver. The following questions will determine whether
the driver is loaded on boot and what permissions it will have.
The current values will be shown in brackets ('[]'). Hitting
```

```
<ENTER> without typing an answer will keep that current value.
Ctrl-C will abort.

Default user to own the driver interface []: oracle
Default group to own the driver interface []: dba
Start Oracle ASM library driver on boot (y/n) [n]: y
Scan for Oracle ASM disks on boot (y/n) [y]: ENTER
Writing Oracle ASM library driver configuration: done
```

c. Use the `oracleasm init` command to load and initialize the ASMLib driver.

```
# oracleasm init
Creating /dev/oracleasm mount point: /dev/oracleasm
Loading module "oracleasm": oracleasm
Configuring "oracleasm" to use device physical block size
Mounting ASMlib driver filesystem: /dev/oracleasm
```

d. Use the `oracleasm configure` command without the `-i` option.

- This command shows the current configuration.

```
# oracleasm configure
ORACLEASM_ENABLED=true
ORACLEASM_UID=oracle
ORACLEASM_GID=dba
ORACLEASM_SCANBOOT=true
ORACLEASM_SCANORDER=""
ORACLEASM_SCANEXCLUDE=""
ORACLEASM_SCAN_DIRECTORIES=""
ORACLEASM_USE_LOGICAL_BLOCK_SIZE="false"
```

e. Use the `oracleasm status` command.

```
# oracleasm status
Checking if ASM is loaded: yes
Checking if /dev/oracleasm is mounted: yes
```

f. Use the `oracleasm createdisk` command to mark `/dev/xvdb1` for ASM use.

- Give the disk a label of `VOL1`.

```
# oracleasm createdisk VOL1 /dev/xvdb1
Writing disk header: done
Instantiating disk: done
```

g. Use the `oracleasm createdisk` command to mark `/dev/xvdd1` for ASM use.

- Give the disk a label of `VOL2`.

```
# oracleasm createdisk VOL2 /dev/xvdd1
Writing disk header: done
Instantiating disk: done
```

3. View ASM disks.

a. Use the `ls` command to display a long list of the `/dev/oracleasm/disks` directory.

```
# ls –l /dev/oracleasm/disks
...
brw-rw----. 1 oracle  dba  ... VOL1
brw-rw----. 1 oracle  dba  ... VOL2
```

b. Use the `oracleasm listdisks` command to list the disk names of marked ASMLib disks.

```
# oracleasm listdisks
VOL1
VOL2
```

c. Use the `oracleasm scandisks` command to enable cluster nodes to identify which shared disks have been marked as ASMLib disks on another node.

```
# oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
```

d. Use the `oracleasm querydisk` command to determine whether a disk name or disk device is being used by ASMLib.

- The first command uses the disk name as an argument.
- The second command uses the device name as an argument.

```
# oracleasm querydisk VOL1
Disk "VOL1" is a valid ASM disk
# oracleasm querydisk /dev/xvdd1
Device "/dev/xvdd1" is marked an ASM disk with the label "VOL2"
```

Practice 17-5 Reverting Changes Made to host03

Overview

In this practice, you:

- Unmark ASM disks, returning them to the system
- Shut down and unload the ASMLib driver
- Remove the `oracleasm-support`, `oracleasmlib`, and `oracle-database-preinstall-18c` packages
- Remove the `oracle` files from the `root` user's home directory
- Delete the `/dev/xvdb1` and `/dev/xvdd1` partitions
- Log off from **host03**

Assumptions

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

Tasks

1. Unmark ASM disks, returning them to the system, and shut down and unload the ASMLib driver.
   a. Use the `oracleasm deletedisk` command to unmark the ASM disk VOL1, returning it to the system.

```
# oracleasm deletedisk VOL1
Clearing disk header: done
Dropping disk: done
```

   b. Use the `oracleasm deletedisk` command to unmark the ASM disk VOL2, returning it to the system.

```
# oracleasm deletedisk VOL2
Clearing disk header: done
Dropping disk: done
```

   c. Use the `oracleasm exit` command to shut down and unload the ASMLib driver.

```
# oracleasm exit
Unmounting ASMlib driver filesystem: /dev/oracleasm
Unloading module "oracleasm": oracleasm
```

2. Remove `oracle` packages.
   a. Use the `yum` command to remove the `oracleasm-support` package.
      - Answer **`y`** when prompted.

```
# yum remove oracleasm-support
...
Transaction Summary
```

```
================================================================
Remove  1 Package

Installed size: 266 k
Is this ok [y/N]: y
...
Removed:
    oracleasm-support.x86_64 ...

Complete!
```

b. Use the `yum` command to remove the `oracleasmlib` package.

- Answer **`y`** when prompted.

```
# yum remove oracleasmlib
...
Transaction Summary
================================================================
Remove  1 Package

Installed size: 39 k
Is this ok [y/N]: y
...
Removed:
    oracleasmlib.x86_64 ...

Complete!
```

c. Use the `yum` command to remove the `oracle-database-preinstall-18c` package.

- Answer **`y`** when prompted.

```
# yum remove oracle-database-preinstall-18c
...
Transaction Summary
================================================================
Remove  1 Package

Installed size: 55 k
Is this ok [y/N]: y
...
```

```
Removed:
     oracle-database-preinstall-18c.x86_64 ...

Complete!
```

d. Use the `rm` command to remove the `oracleasmlib-...el7.x86_64.rpm` and `oracle-database-preinstall-18c-...el7.x86_64.rpm` packages from the `root` user's home directory. Use the `cd` command to ensure you are in the `root` user's home directory first. Answer **`y`** when prompted.

```
# cd
# rm oracle*
rm: remove regular file ‘oracleasmlib-...el7.x86_64.rpm’? y
rm: remove regular file ‘oracle-database-preinstall-18c-
...el7.x86_64.rpm’? y
```

3. Delete the `/dev/xvdb1` and `/dev/xvdd1` partitions.

a. Use the `fdisk` command to delete the `/dev/xvdb1` partition.

- Use the `d` command to delete partition 1.
- Use the `p` command to print the partition table and confirm there are no partitions.
- Use the `w` command to save the partition table and exit the `fdisk` utility.

```
# fdisk /dev/xvdb
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): p

Disk /dev/xvdb: 5368 MB, 5368709120 bytes ...
...
    Device Boot      Start         End      Blocks   Id  System

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

b. Use the `fdisk` command to delete the `/dev/xvdd1` partition.

- Use the `d` command to delete partition 1.
- Use the `p` command to print the partition table and confirm there are no partitions.
- Use the `w` command to save the partition table and exit the `fdisk` utility.

```
# fdisk /dev/xvdd
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): p

Disk /dev/xvdd: 5368 MB, 5368709120 bytes ...
...
    Device Boot      Start         End      Blocks   Id  System

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

4. Log off from **host03** in preparation for the next practice.

   Use the `logout` command to close the `ssh` connection to **host03**.

```
# logout
Connection to host03 closed.
```

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

Practices for Lesson 18: System Monitoring and Management

Practices for Lesson 18: Overview

Practices Overview

In these practices, you:

- Use the `sosreport` utility to collect system information
- Use standard Linux utilities to monitor system resource usage
- Use OSWatcher Black Box and OSWatcher Analyzer

Practice 18-1: Using `sosreport` to Collect System Information

Overview

In this practice, you:

- Use the `sosreport` utility to collect system information
- Extract the compressed TAR file and view the collected information
- View the status of the `sosreport` plug-ins

Assumptions

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

Tasks

1. Log in to **host03**.

   Use the `ssh` command to log in to **host03**. Provide the `root` password when prompted.

```
[dom0]# ssh host03
root@host03's password:
Last login: ...
```

2. Use the `rpm` command to verify that the `sos` package is installed.

```
# rpm -q sos
sos-...noarch
```

   - In this example, the package is installed.

3. Run the `sosreport` command.

   a. Press **Enter** when prompted to continue.

   b. Press **Enter** when prompted to enter your first initial and last name.

   c. Enter number **1** as the case number for which you are generating the report.

```
# sosreport
sosreport (version 3.5)

This command will collect diagnostic and configuration
information from this Oracle Linux system and installed
applications.

An archive containing the collected information will be
generated in /var/tmp/sos... and may be provided to a Oracle
America support representative.

Any information provided to Oracle America will be treated in
accordance with the published support policies at:

  https://linux.oracle.com/
```

```
The generated archive may contain data considered sensitive and
its content should be reviewed by the originating organization
before being passed to any third party.

No changes will be made to system configuration.

Press ENTER to continue, or CTRL-C to quit. ENTER

Please enter your first initial and last name [host03...]: ENTER
Please enter the case number that you are generating... 1
...
Running plugins. Please wait ...
...
Creating compressed archive...

Your sosreport has been generated and saved in:
  /var/tmp/sosreport-host03.example.com.1-...tar.xz

The checksum is: ...

Please send this file to your support representative.
```

4. View the `sosreport` file.
   a. Use the `cd` command to change to the `/var/tmp` directory.

```
# cd /var/tmp
```

   b. Use the `ls` command to display a long listing of the `sos*` files.

```
# ls -l sos*
-rw------- sosreport-host03.example.com.1-...tar.xz
-rw-r--r-- sosreport-host03.example.com.1-...tar.xz.md5
```

   - Note the two `sosreport` files, one with the `.xz` extension and one with the `.md5` extension.
   - The `.xz` file is the compressed data file.
   - Note the size of the `.xz` file.

   c. Use the `xz -d` command to uncompress the `.xz` file.

```
# xz -d sosreport-host03.example.com.1-...tar.xz
```

   d. Use the `ls` command to display a long listing of the `sos*` files.

```
# ls -l sos*
-rw------- sosreport-host03.example.com.1-...tar
-rw-r--r-- sosreport-host03.example.com.1-...tar.xz.md5
```

   - Note that the `sosreport` file with the `.tar.xz` extension now has a `.tar` extension.
   - Note the size of the `.tar` file—it is considerably larger than the compressed (`.xz`) file.

e. Use the `tar` command to extract the `.tar` file.

```
# tar xvf sosreport-host03.example.com.1-...tar
...
```

- Note that the `tar` file is extracted in a `sosreport-host03...` directory.
- Note that the uncompress and extract steps can be accomplished with a single `tar` command: `tar Jxvf sosreport-host03.example.com.1-...tar`. Adding the "`J`" tar option causes `tar` to use `xz` to uncompress the file in addition to extracting the files from the archive.

f. Use the `cd` command to change to the `sosreport-host03...` directory.

```
# cd sosreport-host03...
```

g. Use the `ls` command to display a long listing of the `sosreport-host03...` directory.

```
# ls -l
...
dr-xr-xr-x    ...       boot
lrwxrwxrwx    ...       chkconfig -> sos_commands/services/...
lrwxrwxrwx    ...       date -> sos_commands/general/date
...
drwxr-xr-x    ...       etc
lrwxrwxrwx    ...       free -> sos_commands/memory/free
lrwxrwxrwx    ...       hostname -> sos_commands/general/...
...
dr-xr-xr-x    ...       lib
...
```

- Note that a number of directories that contain data collected from the system exist.
- Note that a number of symbolic links that contain the output of several status-related commands exist.

5. Use the `sosreport -l` command to list the plug-ins.

```
# sosreport -l
sosreport (version 3.5)

The following plugins are currently enabled:

 abrt                 Automatic Bug Reporting Tool
 anaconda             Anaconda installer
 anacron              Anacron job scheduling service
...
The following plugins are currently disabled:

 acpid                inactive       ACPI daemon information
 activemq             inactive       ActiveMQ message broker
```

```
 apache                inactive       Apache http daemon
...
The following plugin options are available:

 abrt.detailed         off   collect detailed info for every ...
 boot.all-images       off   collect lsinitrd for all images
 dmraid.metadata       off   capture dmraid device metadata
...
```

Practice 18-2: Using Standard Linux Performance Monitoring Tools

Overview

In this practice, you use standard Linux system resource monitoring utilities to observe the following:

- CPU statistics
- Memory statistics
- Disk I/O statistics
- Network statistics

Assumptions

- You are the `root` user on **host03**.
- The output shown in the tasks is only a sample. Your output will be different.

Tasks

1. From **dom0**, open a second terminal window.
   a. Use the `su` command to become the `root` user. Provide the `root` password when prompted.

```
[dom0]$ su -
Password:
[dom0]#
```

   b. From this window, use the `ssh` command to log in to **host03**. Provide the `root` password when prompted.

```
[dom0]# ssh host03
root@host03's password:
Last login: ...
```

   - You are now logged on to **host03** from two different terminal windows.

2. Observe the CPU statistics.
   a. In one window, use the `top` command to display CPU usage and load averages.
      - The `top` command also monitors process statistics and memory usage.

```
# top
```

   b. In the second window, run the following command to generate a system load:

```
# dd if=/dev/zero of=/dev/null bs=1024
```

- A sample output of `top` is shown as follows:

```
top - 08:11:35 up  1:53,  5 users,  load average: 1.20, 1.16, 1.14
Tasks: 169 total,   3 running, 166 sleeping,   0 stopped,   0 zombie
%Cpu(s): 14.6 us, 84.8 sy,  0.0 ni,  0.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.7 st
KiB Mem :  2045360 total,   945960 free,   621912 used,   477488 buff/cache
KiB Swap:  4281340 total,  4281340 free,        0 used.  1354600 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 9298 root      20   0  107956    744    672 R 99.0  0.0   4:35.93 dd
 1603 root      20   0  289268  46844  19996 S  0.3  2.3   0:05.18 X
 2065 oracle    20   0 1894840 235256  77816 S  0.3 11.5   0:24.69 gnome-shell
    1 root      20   0  128172   8316   5552 S  0.0  0.4   0:01.73 systemd
    2 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0       0      0      0 S  0.0  0.0   0:00.00 ksoftirqd/0
    5 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0       0      0      0 S  0.0  0.0   0:00.04 kworker/u30+
    7 root      20   0       0      0      0 S  0.0  0.0   0:00.11 rcu_sched
    8 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      20   0       0      0      0 R  0.0  0.0   0:00.19 rcuos/0
   10 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcuob/0
   11 root      rt   0       0      0      0 S  0.0  0.0   0:00.00 migration/0
   12 root      rt   0       0      0      0 S  0.0  0.0   0:00.00 watchdog/0
   13 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 khelper
   14 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kdevtmpfs
   15 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 netns
```

- In this example, CPU usage is high as indicated by the `0.0 id` (idle) statistic.
- The biggest consumer of the CPU is the `dd` process, which has a PID of `9298`.
- Load average is the average number of processes in the run queue or the number of processes waiting to run on the CPU.
  - The load average over the last 1 minute is `1.20`, over the last 5 minutes is `1.16`, and over the last 15 minutes is `1.14`.
  - A high load average is an indication that your system does not have sufficient CPU capacity.
- Press **`q`** to quit the `top` command.

c. Use the `mpstat -P ALL 1` command to display CPU statistics.

- The `1` argument provides statistics at one-second intervals.
- The `-P ALL` option provides statistics for each individual processor and globally for all processors.

```
[root@host03 ~]# mpstat -P ALL 1
Linux    <version>   .el7uek.x86_64 (host03.example.com)     <date>        _x86_64_        (1 CPU)

04:17:03 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
04:17:04 AM  all   16.83    0.00   82.18    0.00    0.00    0.00    0.99    0.00    0.00    0.00
04:17:04 AM    0   16.83    0.00   82.18    0.00    0.00    0.00    0.99    0.00    0.00    0.00

04:17:04 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
04:17:05 AM  all   15.69    0.00   83.33    0.00    0.00    0.00    0.98    0.00    0.00    0.00
04:17:05 AM    0   15.69    0.00   83.33    0.00    0.00    0.00    0.98    0.00    0.00    0.00

04:17:05 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
04:17:06 AM  all   16.16    0.00   83.84    0.00    0.00    0.00    0.00    0.00    0.00    0.00
04:17:06 AM    0   16.16    0.00   83.84    0.00    0.00    0.00    0.00    0.00    0.00    0.00

04:17:06 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
04:17:07 AM  all   15.00    0.00   84.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00
04:17:07 AM    0   15.00    0.00   84.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00

04:17:07 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
04:17:08 AM  all   14.00    0.00   86.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
04:17:08 AM    0   14.00    0.00   86.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- In this example, CPU usage is high as indicated by the `0.00` (`%idle`) statistic.
- Note `%usr`, which is the percentage of CPU used while executing at the user level.
- Note `%sys`, which is the percentage of CPU used while executing at the system (kernel) level. This does not include time spent servicing hardware and software interrupts.
- Press Ctrl + C after viewing a few intervals.

d. Use the `sar -u` command to display system-wide CPU usage.

```
[root@host03 ~]# sar -u
Linux    <version>   .el7uek.x86_64 (host03.example.com)     <date>        _x86_64_        (1 CPU)

12:00:01 AM     CPU     %user     %nice   %system   %iowait    %steal     %idle
12:10:01 AM     all      0.03      0.00      0.03      0.48      0.60     98.86
12:20:01 AM     all      0.03      0.00      0.02      0.48      0.60     98.88
12:30:01 AM     all      0.02      0.00      0.02      0.48      0.60     98.88
12:40:01 AM     all      0.02      0.00      0.02      0.48      0.60     98.88
12:50:01 AM     all      0.03      0.00      0.02      0.47      0.60     98.89
01:00:01 AM     all      0.02      0.00      0.02      0.47      0.60     98.89
01:10:01 AM     all      0.04      0.00      0.03      0.48      0.60     98.86
01:20:01 AM     all      0.03      0.00      0.02      0.48      0.60     98.88
01:30:01 AM     all      0.02      0.00      0.02      0.47      0.60     98.89
01:40:01 AM     all      0.03      0.00      0.02      0.48      0.59     98.89
01:50:01 AM     all      0.03      0.00      0.02      0.48      0.59     98.89
02:00:01 AM     all      3.44      0.00     21.39      0.37      0.59     74.21
02:10:01 AM     all     14.78      0.02     84.63      0.00      0.58      0.00
02:20:01 AM     all     13.58      0.00     85.86      0.00      0.56      0.00
02:30:01 AM     all     13.68      0.00     85.76      0.00      0.56      0.00
02:40:01 AM     all     13.67      0.00     85.77      0.00      0.56      0.00
02:50:01 AM     all     13.71      0.00     85.73      0.00      0.57      0.00
03:00:02 AM     all     13.60      0.00     85.84      0.00      0.56      0.00
03:10:01 AM     all     13.64      0.00     85.80      0.00      0.56      0.00
03:20:01 AM     all     13.64      0.01     85.78      0.00      0.57      0.00
03:30:01 AM     all     13.69      0.00     85.74      0.00      0.56      0.00
03:40:01 AM     all     13.66      0.00     85.77      0.00      0.56      0.00
03:50:01 AM     all     13.70      0.00     85.74      0.00      0.56      0.00
04:00:01 AM     all     13.67      0.00     85.77      0.00      0.56      0.00
04:10:01 AM     all     12.03      0.00     75.59      0.06      0.56     11.76
04:20:01 AM     all      6.26      0.00     28.37      0.35      0.63     64.39
Average:        all      7.20      0.00     44.40      0.23      0.58     47.59
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- In this example, CPU usage is low when `%idle` statistics are high.
- CPU usage is high when `%idle` statistics are 0.00.
- An `Average` line is shown in this example.

e. Use the `sar -q` command to display run queue length and load averages.

```
[root@host03 ~]# sar -q
Linux    <version>  .el7uek.x86_64 (host03.example.com)     <date>        _x86_64_        (1 CPU)

12:00:01 AM   runq-sz  plist-sz   ldavg-1   ldavg-5  ldavg-15   blocked
12:10:01 AM         4       216      0.00      0.01      0.05         0
12:20:01 AM         4       216      0.01      0.02      0.05         0
12:30:01 AM         4       216      0.00      0.01      0.05         0
12:40:01 AM         4       216      0.00      0.01      0.05         0
12:50:01 AM         4       216      0.03      0.02      0.05         0
01:00:01 AM         4       216      0.00      0.01      0.05         0
01:10:01 AM         4       216      0.00      0.01      0.05         0
01:20:01 AM         4       216      0.00      0.01      0.05         0
01:30:01 AM         4       216      0.00      0.01      0.05         0
01:40:01 AM         4       216      0.01      0.02      0.05         0
01:50:01 AM         4       216      0.00      0.01      0.05         0
02:00:01 AM         3       220      0.92      0.40      0.19         0
02:10:01 AM         5       367      1.06      1.05      0.68         0
02:20:01 AM         5       365      1.03      1.04      0.85         0
02:30:01 AM         5       365      1.04      1.10      0.97         0
02:40:01 AM         5       365      1.00      1.03      0.98         0
02:50:01 AM         5       365      1.03      1.07      1.03         0
03:00:02 AM         4       366      1.03      1.04      1.05         0
03:10:01 AM         5       366      1.04      1.04      1.05         0
03:20:01 AM         5       366      1.09      1.06      1.05         0
03:30:01 AM         4       367      1.04      1.03      1.05         0
03:40:01 AM         5       365      1.16      1.07      1.06         0
03:50:01 AM         5       365      1.07      1.08      1.05         0
04:00:01 AM         5       365      1.07      1.07      1.05         0
04:10:01 AM         4       362      0.35      0.82      0.97         0
04:20:01 AM         5       371      0.98      0.64      0.73         0
Average:            4       297      0.58      0.56      0.55         0
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- The CPU is not saturated when there is a low run queue length (`runq-sz`) and low load averages (last 1 minute: `ldavg-1`, last 5 minutes: `ldavg-5`, and last 15 minutes: `ldavg-15`).
- A run queue size greater than the number of CPUs on your system is usually indicative of a CPU bottleneck.
- An `Average` line is shown in this example.

f. Use the `vmstat 1` command to display CPU statistics.

- This command is primarily used for monitoring virtual memory statistics.

```
[root@host03 ~]# vmstat 1
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 4  0      0 920672  39500 438368    0    0    58     3 1079  160 14 81  4  0  1
 1  0      0 920300  39500 438368    0    0     0     0 1127  204 18 82  0  0  0
 1  0      0 920284  39500 438368    0    0     0     0 1126  159 16 83  0  0  1
 2  0      0 920300  39500 438368    0    0     0     0 1124  145 17 82  0  0  1
 1  0      0 920300  39500 438368    0    0     0     0 1127  182 15 85  0  0  0
 1  0      0 920300  39500 438368    0    0     0     0 1125  122 15 84  0  0  1
 1  0      0 920300  39500 438368    0    0     0     0 1128  134 14 86  0  0  0
 1  0      0 920300  39500 438368    0    0     0     0 1123  161 15 84  0  0  1
 1  0      0 920300  39500 438368    0    0     0     0 1128  148 14 86  0  0  0
 2  0      0 920300  39500 438368    0    0     0     0 1123  148 16 83  0  0  1
 1  0      0 920284  39508 438368    0    0     0    36 1139  187 16 83  0  0  1
 1  0      0 920284  39508 438368    0    0     0     0 1123  119 16 84  0  0  0
 2  0      0 920268  39508 438368    0    0     0     0 1136  185 18 81  0  0  1
 1  0      0 920284  39508 438368    0    0     0     0 1121  162 16 84  0  0  0
 2  1      0 920284  39508 438368    0    0     0     0 1126  127 14 85  0  0  1
```

- The `1` argument provides statistics at one-second intervals.
- CPU usage is higher when `id` (`%idle`) statistics are lower.
- High run queue length (`r`) statistics are an indication of CPU saturation.
- Press Ctrl + C after viewing a few intervals.

g. Use the `iostat` command to display CPU usage.

- This command is primarily used for monitoring system I/O device loads.

```
[root@host03 ~]# iostat
Linux   <version>   .el7uek.x86_64 (host03.example.com)     <date>         _x86_64_
(1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.38    0.00    2.03    0.49    0.60   96.51

Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
xvda              0.35         1.86         2.25     934323    1131720
xvdb              0.00         0.04         0.00      19168       2052
xvdd              0.00         0.04         0.00      19184       2052
scd0              0.00         0.04         0.00      22550          0
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- In this example, the average `%idle` statistic for the CPU is 96.51%.

3. Observe the memory statistics.

   a. Use the `top` command to display memory usage.

```
# top
```

   A sample output of `top` is shown as follows:

```
top - 08:23:59 up  2:05,  5 users,  load average: 1.13, 1.21, 1.18
Tasks: 168 total,   3 running, 165 sleeping,   0 stopped,   0 zombie
%Cpu(s): 14.2 us, 85.1 sy,  0.0 ni,  0.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.7 st
KiB Mem :  2045360 total,   926604 free,   640796 used,   477960 buff/cache
KiB Swap:  4281340 total,  4281340 free,        0 used.  1335696 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 9298 root      20   0  107956    744    672 R 98.7  0.0  16:43.16 dd
 2065 oracle    20   0 1909800 250264  77816 S  0.7 12.2   0:36.34 gnome-shell
10276 root      20   0  157720   4244   3536 R  0.3  0.2   0:00.02 top
    1 root      20   0  128172   8316   5552 S  0.0  0.4   0:01.78 systemd
    2 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0       0      0      0 S  0.0  0.0   0:00.00 ksoftirqd/0
    5 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0       0      0      0 S  0.0  0.0   0:00.04 kworker/u30:0
    7 root      20   0       0      0      0 S  0.0  0.0   0:00.11 rcu_sched
    8 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      20   0       0      0      0 R  0.0  0.0   0:00.21 rcuos/0
   10 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcuob/0
   11 root      rt   0       0      0      0 S  0.0  0.0   0:00.00 migration/0
   12 root      rt   0       0      0      0 S  0.0  0.0   0:00.01 watchdog/0
   13 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 khelper
   14 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kdevtmpfs
   15 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 netns
   16 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 perf
   17 root      20   0       0      0      0 S  0.0  0.0   0:00.00 xenwatch
```

   - The upper section of `top` displays memory statistics:
     - `KiB Mem`: This line reflects how much physical memory your system has: how much is free, how much is used, and how much is associated with buff/cache.
     - `KiB Swap`: This line reflects how much swap memory your system has, how much is used, and how much is free.
     - When a computer runs out of physical memory and starts using swap space, its performance deteriorates dramatically.
     - If you run out of swap, you will most likely crash your programs or the OS.
   - The lower section of `top` displays a list of processes sorted by CPU usage, with the top consumer of CPU listed first.

b. Sort by memory usage, press F or f to change the sort field, and then use the up/down arrow keys to highlight either of the following:

- `%MEM` = Memory usage (RES)
- `RES` = Resident size (kb)

- Press the **s** key to set the selected sort field.
- Press **q** or **Esc** to exit the field management window and return to the `top` display.

```
top - 08:31:41 up  2:13,  5 users,  load average: 1.20, 1.14, 1.14
Tasks: 168 total,   3 running, 165 sleeping,   0 stopped,   0 zombie
%Cpu(s): 13.9 us, 85.4 sy,  0.0 ni,  0.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.7 st
KiB Mem :  2045360 total,   926480 free,   640848 used,   478032 buff/cache
KiB Swap:  4281340 total,  4281340 free,        0 used.  1335616 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 2065 oracle    20   0 1909800 250268  77816 S  0.3 12.2   0:38.74 gnome-shell
 2278 oracle    20   0 1114848  72276  31516 S  0.0  3.5   0:00.11 evolution-calen
 2388 oracle    20   0 1212792  68932  28940 S  0.0  3.4   0:00.09 evolution-calen
 2431 oracle    20   0 1130060  68716  28772 S  0.0  3.4   0:00.09 evolution-calen
 2354 oracle    20   0  922580  52292  28300 S  0.0  2.6   0:00.49 gnome-software
 1603 root      20   0  292412  50044  20000 S  0.3  2.4   0:08.33 X
 2222 oracle    20   0 1370708  37792  29160 S  0.0  1.8   0:00.51 gnome-settings-
  673 root      20   0  338252  37640  11320 S  0.0  1.8   0:00.70 firewalld
 2482 oracle    20   0 1176524  37492  31356 S  0.0  1.8   0:00.04 evolution-addre
 2436 oracle    20   0 1069768  37456  31036 S  0.0  1.8   0:00.05 evolution-addre
 2180 oracle    20   0 1006472  37304  30728 S  0.0  1.8   0:00.06 evolution-sourc
 2240 oracle    20   0 1012016  36128  25944 S  0.0  1.8   0:00.58 nautilus-deskto
 2563 oracle    20   0  721740  34988  24196 S  0.0  1.7   0:03.79 gnome-terminal-
 2095 oracle    20   0  906704  33640  27556 S  0.0  1.6   0:00.05 gnome-shell-cal
 2117 oracle    20   0  906212  32516  26860 S  0.0  1.6   0:00.03 goa-daemon
 2343 oracle    20   0  533668  24884  19464 S  0.0  1.2   0:00.06 abrt-applet
```

- Press **q** to quit the `top` command.

c. Use the `vmstat 1` command to display memory statistics.

```
[root@host03 ~]# vmstat 1
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 4  0      0 927116  39600 438428    0    0    52     2 1084  155 14 81  4  0  1
 1  0      0 927116  39600 438428    0    0     0     0 1129  141 15 84  0  0  1
 1  0      0 927116  39600 438428    0    0     0     0 1136  241 16 83  0  0  1
 1  0      0 927084  39600 438428    0    0     0     0 1126  132 15 85  0  0  0
 1  0      0 927116  39600 438428    0    0     0     0 1123  112 14 85  0  0  1
 1  0      0 926836  39600 438428    0    0     0     0 1138  229 15 85  0  0  0
 1  0      0 926868  39600 438428    0    0     0     0 1123  120 15 84  0  0  1
 1  0      0 926868  39600 438428    0    0     0     0 1126  126 16 83  0  0  1
 1  0      0 926836  39600 438428    0    0     0     0 1123  141 19 81  0  0  0
 1  0      0 926868  39600 438428    0    0     0     0 1126  126 16 83  0  0  1
 2  0      0 926868  39600 438428    0    0     0     0 1135  112 15 85  0  0  0
 1  0      0 926868  39600 438428    0    0     0     0 1126  145 15 84  0  0  1
 1  0      0 926868  39600 438428    0    0     0     0 1122  102 15 85  0  0  0
 1  0      0 926836  39600 438428    0    0     0     0 1125  147 15 84  0  0  1
 1  0      0 926868  39600 438428    0    0     0     0 1123  109 13 87  0  0  0
```

- The `1` argument provides statistics at one-second intervals.
- The important memory statistics are:
  - `swpd` – The amount of virtual memory used
  - `free` – The amount of idle memory
  - `si` – The amount of memory swapped in from disk (per second)
  - `so` – The amount of memory swapped out to disk (per second)
- In this example, the system has sufficient free memory and is not swapping.
- Press Ctrl + C after viewing a few intervals.

d. Use the `free -m` command to display memory statistics.

```
[root@host03 ~]# free -m
              total        used        free      shared  buff/cache   available
Mem:           1997         625         905           9         466        1305
Swap:          4180           0        4180
```

- This example uses the `-m` option to display amounts in megabytes.
- This command displays the total amount of free and used physical and swap memory in your system.

e. Use the `sar -r` command to display memory usage statistics. Partial output is shown as follows:

```
[root@host03 ~]# sar -r
Linux   <version>   .el7uek.x86_64 (host03.example.com)     <date>        _x86_64_  (1 CPU)

12:00:01 AM kbmemfree kbmemused  %memused kbbuffers  kbcached  kbcommit   %commit  kbactive   kbinact   kbdirty
12:10:01 AM    514280   1531080     74.86    132476    977360   1356740     21.44    751316    549844         0
12:20:01 AM    514192   1531168     74.86    132488    977368   1356740     21.44    751364    549844         0
12:30:01 AM    514192   1531168     74.86    132504    977372   1356740     21.44    751404    549844         0
12:40:01 AM    514156   1531204     74.86    132524    977384   1356740     21.44    751452    549844         0
12:50:01 AM    514068   1531292     74.87    132548    977380   1356740     21.44    751496    549844         0
01:00:01 AM    514076   1531284     74.87    132564    977392   1356740     21.44    751548    549844         0
01:10:01 AM    513952   1531408     74.87    132592    977396   1356740     21.44    751608    549844         0
01:20:01 AM    513948   1531412     74.87    132604    977404   1356740     21.44    751652    549844         0
01:30:01 AM    513920   1531440     74.87    132620    977408   1356740     21.44    751692    549844         0
01:40:01 AM    513840   1531520     74.88    132636    977416   1356740     21.44    751732    549844         0
01:50:01 AM    513840   1531520     74.88    132656    977412   1356740     21.44    751804    549844         0
02:00:01 AM    507724   1537636     75.18    132700    977472   1302292     20.58    757200    549680         0
02:10:01 AM    140148   1905212     93.15    133152   1012592   3333036     52.68   1103792    547544        76
02:20:01 AM    141716   1903644     93.07    133196   1012616   3331412     52.66   1102568    547512         8
02:30:01 AM    141508   1903852     93.08    133224   1012604   3331412     52.66   1102652    547512         0
02:40:01 AM    141456   1903904     93.08    133260   1012648   3331412     52.66   1102716    547516        12
02:50:01 AM    141332   1904028     93.09    133304   1012620   3331412     52.66   1102756    547512         0
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- Observe the `%memused` value, which is the percentage of used memory.
- An `Average` line is not shown in this example.

f. Use the `sar -B` command to display memory paging statistics. Partial output is shown as follows:

```
[root@host03 ~]# sar -B
Linux   <version>   .el7uek.x86_64 (host03.example.com)     <date>        _x86_64_        (1 CPU)

12:00:01 AM  pgpgin/s pgpgout/s   fault/s  majflt/s  pgfree/s pgscank/s pgscand/s pgsteal/s    %vmeff
12:10:01 AM      0.00      0.47     36.86      0.00     28.08      0.00      0.00      0.00      0.00
12:20:01 AM      0.00      0.27     27.26      0.00     21.00      0.00      0.00      0.00      0.00
12:30:01 AM      0.00      0.22     27.96      0.00     21.62      0.00      0.00      0.00      0.00
12:40:01 AM      0.00      0.30     27.27      0.00     21.00      0.00      0.00      0.00      0.00
12:50:01 AM      0.00      0.22     27.31      0.00     21.02      0.00      0.00      0.00      0.00
01:00:01 AM      0.00      0.19     27.41      0.00     21.12      0.00      0.00      0.00      0.00
01:10:01 AM      0.00      0.42     36.98      0.00     28.15      0.00      0.00      0.00      0.00
01:20:01 AM      0.00      0.27     27.28      0.00     21.00      0.00      0.00      0.00      0.00
01:30:01 AM      0.00      0.14     27.32      0.00     21.07      0.00      0.00      0.00      0.00
01:40:01 AM      0.00      0.27     27.20      0.00     20.98      0.00      0.00      0.00      0.00
01:50:01 AM      0.00      0.25     27.18      0.00     20.91      0.00      0.00      0.00      0.00
02:00:01 AM      0.00      1.81     55.68      0.00     40.44      0.00      0.00      0.00      0.00
02:10:01 AM     57.29     15.59    477.65      0.48    417.90      0.00      0.00      0.00      0.00
02:20:01 AM      0.03      0.58     57.07      0.00     49.04      0.00      0.00      0.00      0.00
02:30:01 AM      0.07      0.31     42.10      0.00     34.87      0.00      0.00      0.00      0.00
02:40:01 AM      0.00      0.42     41.71      0.00     34.85      0.00      0.00      0.00      0.00
02:50:01 AM      0.00      0.27     28.36      0.00     22.18      0.00      0.00      0.00      0.00
03:00:02 AM      0.05      0.73     29.68      0.00     23.02      0.00      0.00      0.00      0.00
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- Observe `pgscank/s`, which is the number of pages scanned by the `kswapd` daemon per second, and `pgscand/s`, which is the number of pages scanned directly per second.
- An `Average` line is not shown in this example.

g. Use the `sar -W` command to display memory swapping statistics. Partial output is shown as follows:

```
[root@host03 ~]# sar -W
Linux   <version>   .el7uek.x86_64 (host03.example.com)     <date>         _x86_64_
        (1 CPU)

12:00:01 AM  pswpin/s pswpout/s
12:10:01 AM      0.00      0.00
12:20:01 AM      0.00      0.00
12:30:01 AM      0.00      0.00
12:40:01 AM      0.00      0.00
12:50:01 AM      0.00      0.00
01:00:01 AM      0.00      0.00
01:10:01 AM      0.00      0.00
01:20:01 AM      0.00      0.00
01:30:01 AM      0.00      0.00
01:40:01 AM      0.00      0.00
01:50:01 AM      0.00      0.00
02:00:01 AM      0.00      0.00
02:10:01 AM      0.00      0.00
02:20:01 AM      0.00      0.00
02:30:01 AM      0.00      0.00
02:40:01 AM      0.00      0.00
02:50:01 AM      0.00      0.00
03:00:02 AM      0.00      0.00
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- The `pswpin/s` value is the number of swap pages the system swapped in per second, and `pswpout/s` is the number of swap pages the system swapped out per second.
- An `Average` line is not shown in this example.

h. Use the `cat` command to view the contents of `/proc/meminfo`. Partial output is shown as follows:

```
[root@host03 ~]# cat /proc/meminfo
MemTotal:        2045360 kB
MemFree:          913616 kB
MemAvailable:    1322828 kB
Buffers:           39668 kB
Cached:           364084 kB
SwapCached:            0 kB
Active:           737088 kB
Inactive:         254776 kB
Active(anon):     588824 kB
Inactive(anon):     9124 kB
Active(file):     148264 kB
Inactive(file):   245652 kB
Unevictable:           0 kB
Mlocked:               0 kB
SwapTotal:       4281340 kB
SwapFree:        4281340 kB
Dirty:                 0 kB
Writeback:             0 kB
AnonPages:        588148 kB
Mapped:           170176 kB
Shmem:              9844 kB
```

4. Observe the disk I/O statistics.

   a. Use the `iostat -xz` command to display I/O statistics for devices.

```
[root@host03 ~]# iostat -xz
Linux   <version>  .el7uek.x86_64 (host03.example.com)      <date>      _x86_64_        (1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          14.36    0.00   81.41    0.35    0.72    3.16

Device:         rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await r_await w_await  svctm  %util
xvda              0.00     0.03    2.32    0.48    44.09     2.14    33.05     0.05   17.39   19.35    7.91   1.56   0.44
xvdb              0.00     0.00    0.04    0.00     0.79     0.00    44.83     0.00   20.35   20.35    0.00   7.58   0.03
xvdd              0.00     0.00    0.03    0.00     0.68     0.00    43.70     0.00   17.30   17.30    0.00  12.44   0.04
scd0              0.00     0.00    0.00    0.00     0.11     0.00    83.36     0.00   42.80   42.80    0.00  37.40   0.01
```

   - Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
   - Observe the `%util` value, which is the percentage of CPU time during which I/O requests were issued to the device.
   - Device saturation occurs when this value is close to `100%`.
   - Observe the `avgqu-sz` value, which is the average queue length of the requests that were issued to the device.
   - If average queue length is greater than `1`, it is an indication of device I/O saturation.

b. Use the `sar -d` command to display I/O statistics for devices. Partial output is shown as follows:

```
[root@host03 ~]# sar -d
Linux    <version>   .el7uek.x86_64 (host03.example.com)      <date>       _x86_64_        (1 CPU)

12:00:01 AM       DEV       tps  rd_sec/s  wr_sec/s  avgrq-sz  avgqu-sz     await     svctm     %util
12:10:01 AM  dev202-0      0.11      0.00      0.95      8.61      0.00      3.61      0.73      0.01
12:10:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:10:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:10:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:20:01 AM  dev202-0      0.06      0.00      0.53      8.89      0.00      2.28      0.78      0.00
12:20:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:20:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:20:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:30:01 AM  dev202-0      0.05      0.00      0.44      8.80      0.00      2.77      1.10      0.01
12:30:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:30:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:30:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:40:01 AM  dev202-0      0.07      0.00      0.60      8.78      0.00      2.80      0.78      0.01
12:40:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:40:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:40:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:50:01 AM  dev202-0      0.06      0.00      0.44      8.00      0.00      1.12      0.24      0.00
12:50:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:50:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:50:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00

12:50:01 AM       DEV       tps  rd_sec/s  wr_sec/s  avgrq-sz  avgqu-sz     await     svctm     %util
01:00:01 AM  dev202-0      0.04      0.00      0.39      8.92      0.00      1.15      0.23      0.00
01:00:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
01:00:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
```

- Note that the Oracle Linux version and date field have been replaced with "<version>" and "<date>", but normally would be shown.
- This command also provides `%util` and `avgqu-sz` statistics.
- An `Average` line is not shown in this example.

5. Observe network statistics.

a. Use the `ip -s link` command to observe network statistics. Partial output is shown.

```
[root@host03 ~]# ip -s link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    RX: bytes  packets  errors  dropped overrun mcast
    264790     2703     0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    264790     2703     0       0       0       0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default qlen 1000
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    28447534   36057    0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    5958580    18432    0       0       0       0
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default qlen 1000
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    253721     7766     0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    77860      579      0       0       0       0
```

- The number of bytes received and bytes transmitted on each interface is provided by the `RX: bytes` and `TX: bytes` values.
- This command also provides the number of packets transmitted and received, errors, dropped packets, overruns, and collisions.
- Frames are dropped, and the overrun counter is incremented when the capacity of the interface is exceeded.

b. Use the `netstat -s` command to observe network statistics.

- If the `ip` command shows an excessive amount of errors, more information can be found by examining the `netstat -s` output. Partial output is shown as follows:

```
[root@host03 ~]# netstat -s
Ip:
    48514 total packets received
    7 with invalid addresses
    0 forwarded
    0 incoming packets discarded
    32353 incoming packets delivered
    22852 requests sent out
    332 outgoing packets dropped
Icmp:
    1171 ICMP messages received
    0 input ICMP message failed.
    ICMP input histogram:
        destination unreachable: 1139
        echo requests: 7
        echo replies: 25
    1203 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 1171
        echo request: 25
        echo replies: 7
IcmpMsg:
        InType0: 25
        InType3: 1139
        InType8: 7
        OutType0: 7
        OutType3: 1171
        OutType8: 25
Tcp:
    28 active connections openings
    64 passive connection openings
    1 failed connection attempts
    2 connection resets received
    3 connections established
    30000 segments received
    22105 segments send out
    9 segments retransmited
    0 bad segments received.
    3 resets sent
```

- This command displays summary statistics for each protocol.
- Observe the number of segments retransmitted as an indicator of network interface saturation.
- Many performance problems associated with the network involve retransmission of TCP packets.

c. Use the `netstat -i` command to observe a table listing of network interfaces.

```
[root@host03 ~]# netstat -i
Kernel Interface table
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0      1500      153      0      0 0           80      0      0      0 BMRU
eth1      1500       98      0      0 0           39      0      0      0 BMRU
lo       65536       28      0      0 0           28      0      0      0 LRU
virbr0    1500        0      0      0 0            0      0      0      0 BMU
```

- Observe the `RX-ERR` and `TX-ERR` values for any receive and transmit errors.

6. Use the System Monitor GUI to display system resource usage.

   This application requires that you access the GNOME desktop.

   a. Use the `exit` command to close the `ssh` connection to **host03**.

   - Do not close the connection in the window that is running the "`dd`" command from an earlier step.
   - Close the connection in the window in which you ran `top`, `vmstat`, `iostat`, and `netstat` commands.

```
[host03]# exit
logout
Connection to host03 closed.
```

   b. From **dom0**, connect to **host03** by using the `xm vncviewer host03&` command.

```
[dom0]# xm vncviewer host03&
```

   The GNOME login window appears.

   c. Select Oracle Student from the GNOME login window; enter the password.

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

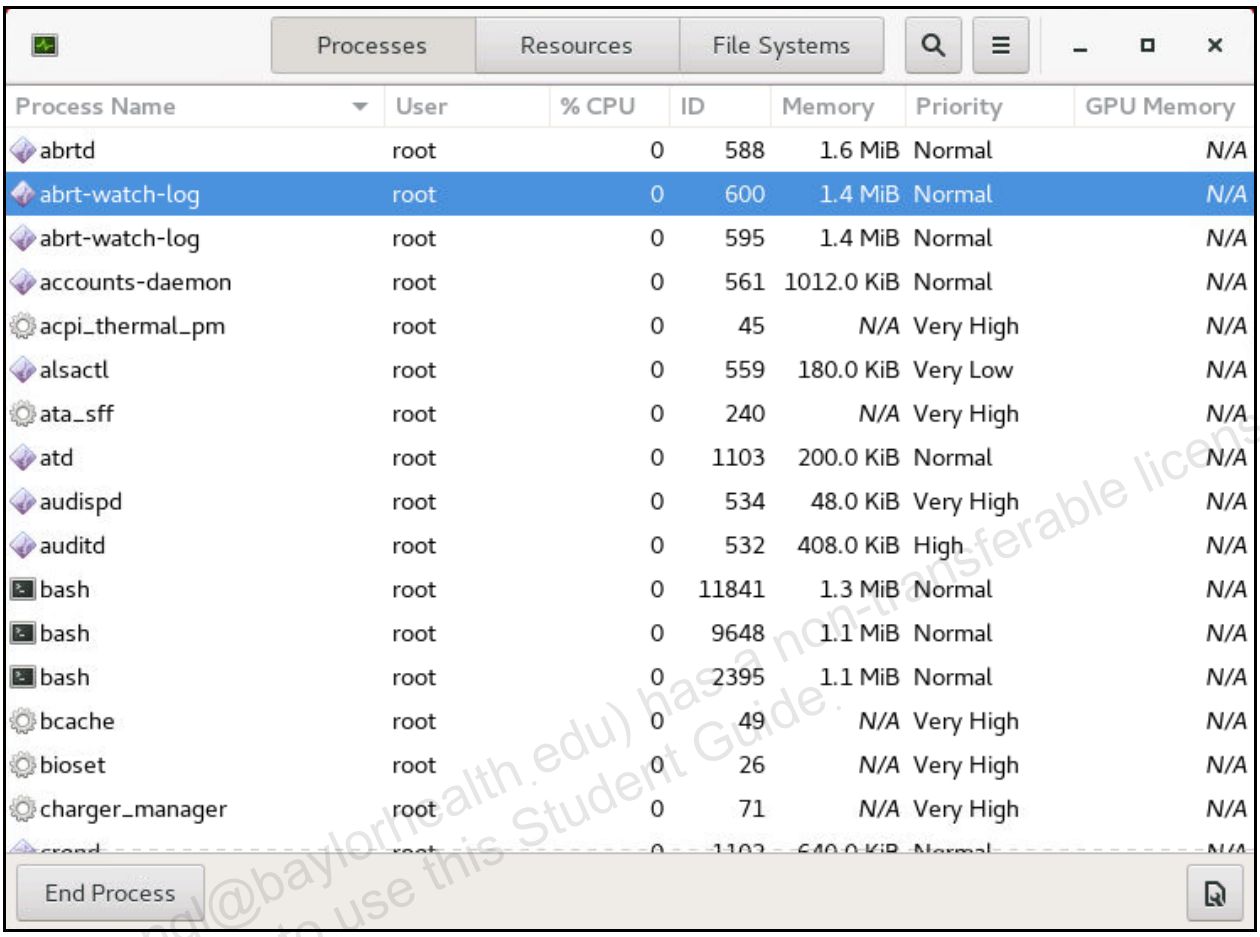
   e. From **host03**, use the `su -` command to become the `root` user. Enter the password when prompted.

```
$ su -
Password:
Last login: ...
#
```

   f. From the terminal window, enter the `gnome-system-monitor` command to display the System Monitor GUI.

```
# gnome-system-monitor
```

- The System Monitor is displayed with the Processes tab selected, as shown in the following screenshot. Partial output is shown.



- Click any column header to sort on that column.
- Note that there is an "**End Process**" button you can use to kill a selected process.

g. Click the other tabs and observe the information displayed on each tab:

1) Resources tab

- Note that there are graphs showing CPU, Memory and Swap, and Network history in real time.

2) File Systems tab

- Note the file system information displayed, including devices, mount points, file system types, and space usage.

h. Click the **X** character in the top right of the window to exit the System Monitor GUI.

- Do not exit the GNOME desktop.

Practice 18-3: Installing and Using OSWatcher

Overview

In this practice, you install and run the OSWatcher Black Box (OSWbb) product and view the collected data. Note that MOS Doc ID 580513.1 describes "How to Start OSWatcher Black Box Every System Boot Using RPM oswbb-service." The OSWatcher TAR file has been staged on **dom0** for this course.

Assumptions

- You are the `root` user on the **host03** VM.
- The `oswbb812.tar` file exists in the `/OVS/seed_pool/oswbb` directory on **dom0**.

Tasks

1. From **host03**, use the `scp` command to copy the `oswbb812.tar` file from **dom0**. Enter **`yes`** to continue connecting and provide the **dom0** `root` password when prompted.

```
# scp dom0:/OVS/seed_pool/oswbb/oswbb812.tar ~
The authenticity of host 'dom0 (192.0.2.1)' can't be
established.
RSA key fingerprint is SHA256:...
RSA key fingerprint is MD5:...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'dom0,192.0.2.1' (RSA) to the list of
known hosts.
root@dom0's password:
oswbb812.tar                          100% 4760KB   27.3MB/s   00:00
```

2. From **host03**, install, start, and stop OSWbb.
   a. Use the `cd` command to change to the `root` user's home directory.

```
# cd
```

   b. Use the `tar` command to extract the `oswbb812.tar` file.

```
# tar xvf oswbb812.tar
oswbb/
...
```

   c. Use the `cd` command to change to the `oswbb` directory, and then use the `ls` command with the `--file-type` option to view the contents of the directory.

```
# cd oswbb
# ls --file-type
analysis/           locks/          psmemsub.sh
call_du.sh          ltop.sh         sarsub.sh
call_sar.sh         mpsub.sh        src/
call_uptime.sh      nfssub.sh       startOSWbb.sh
```

```
data/                 OSWatcherFM.sh  stopOSWbb.sh
docs/                 OSWatcher.sh    tar_up_full_archive.sh
Example_extras.txt    OSWatcher.sh~   tar_up_partial_archive.sh
Exampleprivate.net    oswbba.jar      tmp/
genprvnet.sh          oswib.sh        topaix.sh
gif/                  oswnet.sh       vmsub.sh
ifconfigsub.sh        oswrds.sh       xtop.sh
iosub.sh              oswsub.sh
```

- Note the `startOSWbb.sh` file, which is the script used to start OSWbb.

d. Use the `startOSWbb.sh` command to start OSWbb.

```
# ./startOSWbb.sh
Info...You did not enter a value for snapshotInterval.
Info...Using default value = 30
Info...You did not enter a value for archiveInterval.
Info...Using default value = 48
Setting the archive log directory to /root/oswbb/archive

Testing for discovery of OS Utilities...
VMSTAT found on your system.
IOSTAT found on your system.
MPSTAT found on your system.
IFCONFIG found on your system.
NETSTAT found on your system.
TOP found on your system.
TRACEROUTE found on your system.
Cat /etc/oratab: No such file or directory

Discovery of CPU CORE COUNT
...
CPU CORE COUNT = 1
VCPUS/THREADS = 1

Discovery completed.

Starting OSWatcher v8.1.2 on ...
With SnapshotInterval = 30
With ArchiveInterval = 48
...
Starting Data Collection...

oswbb heartbeat:...
oswbb heartbeat:...
```

```
oswbb heartbeat:...
...
```

- OSWbb started successfully, the discovery process completed, and data collection begins.
- The default intervals (`snapshotInterval = 30` and `archiveInterval = 48`) are used.

e. After a few data collection events (`oswbb heartbeat`) have completed, use the `stopOSWbb.sh` command to stop OSWbb. Press **Enter** first to get the prompt back.

```
# ./stopOSWbb.sh
```

3. View the data collection directories.

   Use the `cd` command to change to the `archive` directory, and then use the `ls` command with the `--file-type` option to view the contents of the directory.

```
# cd archive
# ls --file-type
oswcpuinfo/   oswmeminfo/  oswprvtnet/   oswtop/
oswifconfig/  oswmpstat/   oswps/        oswvmstat/
oswiostat/    oswnetstat/  oswslabinfo/
```

- The `archive` directory is created when OSWbb is started for the first time.
- The directory contains subdirectories for each data collector.

4. View the `oswiostat` directory.

   a. Use the `cd` command to change to the `oswiostat` directory, and then use `ls` to view the contents of the directory.

```
# cd oswiostat
# ls
host03.example.com_iostat...
```

   b. Use the `less` command to view the file.

```
# less host03...
Linux OSWbb v...
zzz ***...
...
```

- Note that this file contains the output of the `iostat -x` command.
- The `iostat` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

5. View the `oswmpstat` directory.

   a. Use the `cd` command to change to the `oswmpstat` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswmpstat
# pwd
/root/oswbb/archive/oswmpstat
```

```
# ls
host03.example.com_mpstat...
```

b. Use the `less` command to view the file.

```
# less host03...
Linux OSWbb v...
zzz ***...
...
```

- Note that this file contains the output of the `mpstat` command.
- The `mpstat` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

6. View the `oswprvtnet` directory.

Use the `cd` command to change to the `oswprvtnet` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswprvtnet
# pwd
/root/oswbb/archive/oswprvtnet
# ls
```

- Note that this directory is empty.
- This directory contains the status of RAC private networks.
- You need to manually create the `private.net` file to run `traceroute` commands.

7. View the `oswslabinfo` directory.

a. Use the `cd` command to change to the `oswslabinfo` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswslabinfo
# pwd
/root/oswbb/archive/oswslabinfo
# ls
host03.example.com_slabinfo...
```

b. Use the `less` command to view the file.

```
# less host03...
zzz ***...
slabinfo - version: ...
...
```

- Note that this file contains the contents of the `/proc/slabinfo` file.
- The `/proc/slabinfo` file is read at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

8. View the `oswvmstat` directory.

a. Use the `cd` command to change to the `oswvmstat` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswvmstat
# pwd
/root/oswbb/archive/oswvmstat
# ls
host03.example.com_vmstat...
```

b. Use the `less` command to view the file.

```
# less host03...
Linux OSWbb v...
SNAP_INTERVAL 30
CPU_CORES 1
OSWBB_ARCHIVE_DEST /root/oswbb/archive
zzz ***...
...
```

- Note that this file contains the output of the `vmstat` command.
- The `vmstat` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

9. View the `oswmeminfo` directory.

a. Use the `cd` command to change to the `oswmeminfo` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswmeminfo
# pwd
/root/oswbb/archive/oswmeminfo
# ls
host03.example.com_meminfo...
```

b. Use the `less` command to view the file.

```
# less host03...
zzz ***...
MemTotal: ...
MemFree: ...
...
```

- Note that this file contains the contents of the `/proc/meminfo` file.
- The `/proc/meminfo` file is read at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

10. View the `oswnetstat` directory.

a. Use the `cd` command to change to the `oswnetstat` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswnetstat
# pwd
/root/oswbb/archive/oswnetstat
# ls
host03.example.com_netstat...
```

b. Use the `less` command to view the file.

```
# less host03...
Linux OSWbb v...
zzz ***...
...
```

- Note that this file contains the output of the `netstat` command.
- The `netstat` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

11. View the `oswps` directory.

a. Use the `cd` command to change to the `oswps` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswps
# pwd
/root/oswbb/archive/oswps
# ls
host03.example.com_ps...
```

b. Use the `less` command to view the file.

```
# less host03...
Linux OSWbb v...
zzz ***...
...
```

- Note that this file contains the output of the `ps` command.
- The `ps` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

12. View the `oswtop` directory.

a. Use the `cd` command to change to the `oswtop` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswtop
# pwd
/root/oswbb/archive/oswtop
# ls
host03.example.com_top...
```

b. Use the `less` command to view the file.

```
# less host03...
```

```
Linux OSWbb v...
zzz ***...
...
```

- Note that this file contains the output of the `top` command.
- The `top` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

13. View the `oswifconfig` directory.
    a. Use the `cd` command to change to the `oswifconfig` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswifconfig
# pwd
/root/oswbb/archive/oswifconfig
# ls
host03.example.com_ifconfig...
```

    b. Use the `less` command to view the file.

```
# less host03...
Linux OSWbb v...
zzz ***...
...
```

- Note that this file contains the output of the `ifconfig` command.
- The `top` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

14. View the `oswcpuinfo` directory.
    a. Use the `cd` command to change to the `oswcpuinfo` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswcpuinfo
# pwd
/root/oswbb/archive/oswcpuinfo
# ls
host03.example.com_cpuinfo...
```

    b. Use the `less` command to view the file.

```
# less host03...
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 58
model name      : Intel(R) Core(TM) ...
...
```

- Note that this file contains the contents of `/proc/cpuinfo`.

Practice 18-4: Using OSWatcher Analyzer

Overview

In this practice, you perform the following:

- Start OSWatcher Analyzer (OSWbba) on **host03**.
- View CPU and Memory Graphs from OSWbba.
- Use OSWbba to analyze the data.
- View the analysis report.

Assumptions

- You are the `root` user on **host03**.
- You completed the previous practice “Installing and Using OSWatcher.”

Tasks

1. Start OSWatcher Analyzer (OSWbba) on **host03**.
   a. Use the `java -version` command to display the Java version number.

```
# java -version
openjdk version "1.8.0_161"
...
```

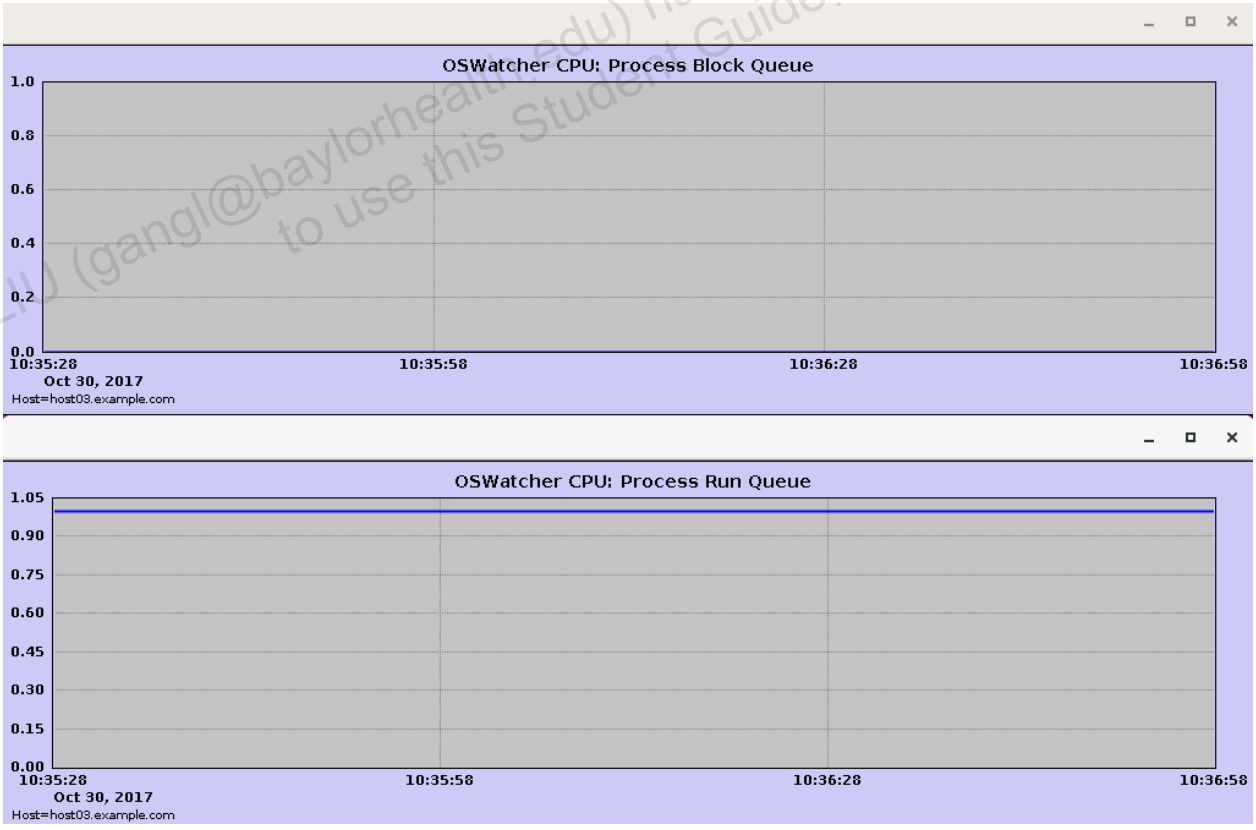
   - In this example, version `1.8.0_161` is installed.
   - The minimum version is `1.4.2`.

   b. Use the `java -jar` command to display the OSWbba menu. First, ensure that you are in the directory where OSWbba is installed (`~/oswbb`).

```
# cd ~/oswbb
# java -jar oswbba.jar -i ~/oswbb/archive
...
Enter 1 to Display CPU Process Queue Graphs
Enter 2 to Display CPU Utilization Graphs
Enter 3 to Display CPU Other Graphs
Enter 4 to Display Memory Graphs
Enter 5 to Display Disk IO Graphs

Enter GC to Generate All CPU Gif Files
Enter GM to Generate All Memory Gif Files
Enter GD to Generate All Disk Gif Files
Enter GN to Generate All Network Gif Files

Enter L to Specify Alternate Location of Gif Directory
Enter Z to Zoom Graph Time Scale (Does not change analysis
dataset)
```

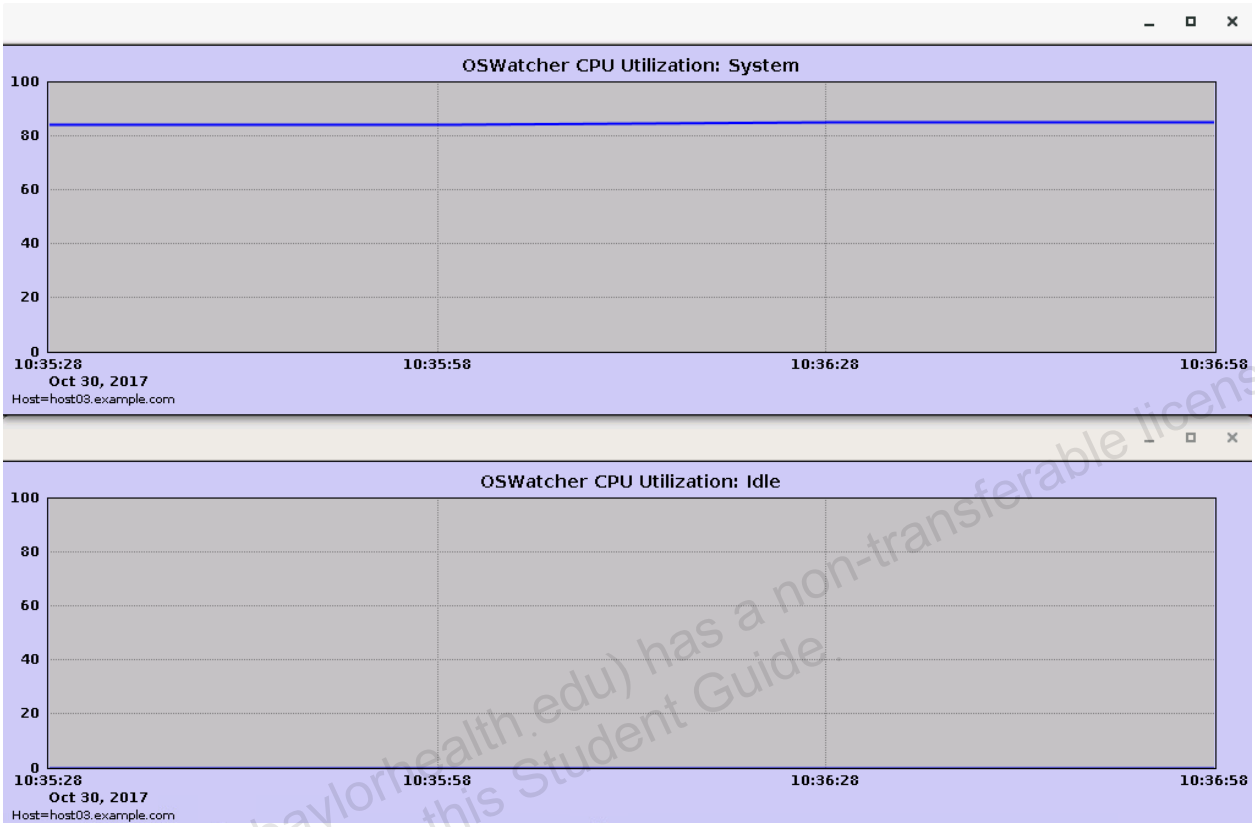
```
Enter B to Returns to Baseline Graph Time Scale (Does not change
analysis dataset)
Enter R to Remove Currently Displayed Graphs

Enter X to Export Parsed Data to Flat File
Enter S to Analyze Subset of Data(Changes analysis dataset
including graph time scale)
Enter A to Analyze Data
Enter D to Generate DashBoard

Enter Q to Quit Program

Please Select an Option:
```

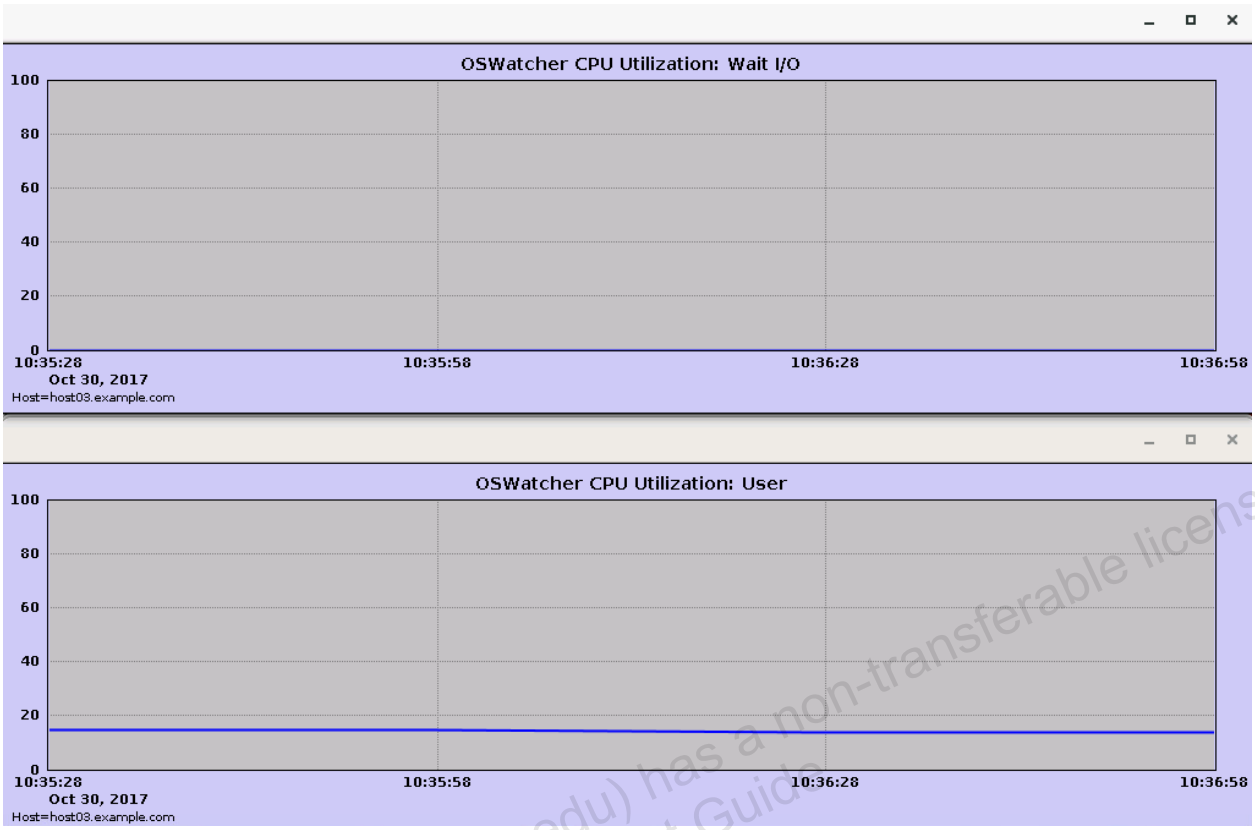
2. View CPU and Memory Graphs. Sample output is shown as follows. Your graphs will be different.
   a. From the OSWbba menu, select 1 to display the two CPU Process Queue graphs: Process Block Queue and Process Run Queue. View each graph by clicking each one from the window list at the bottom of the GNOME screen or drag and align the graphs to see them, as desired.



   - After viewing the graphs, click the terminal window from the window list if necessary and select R from the menu to remove the graphs.

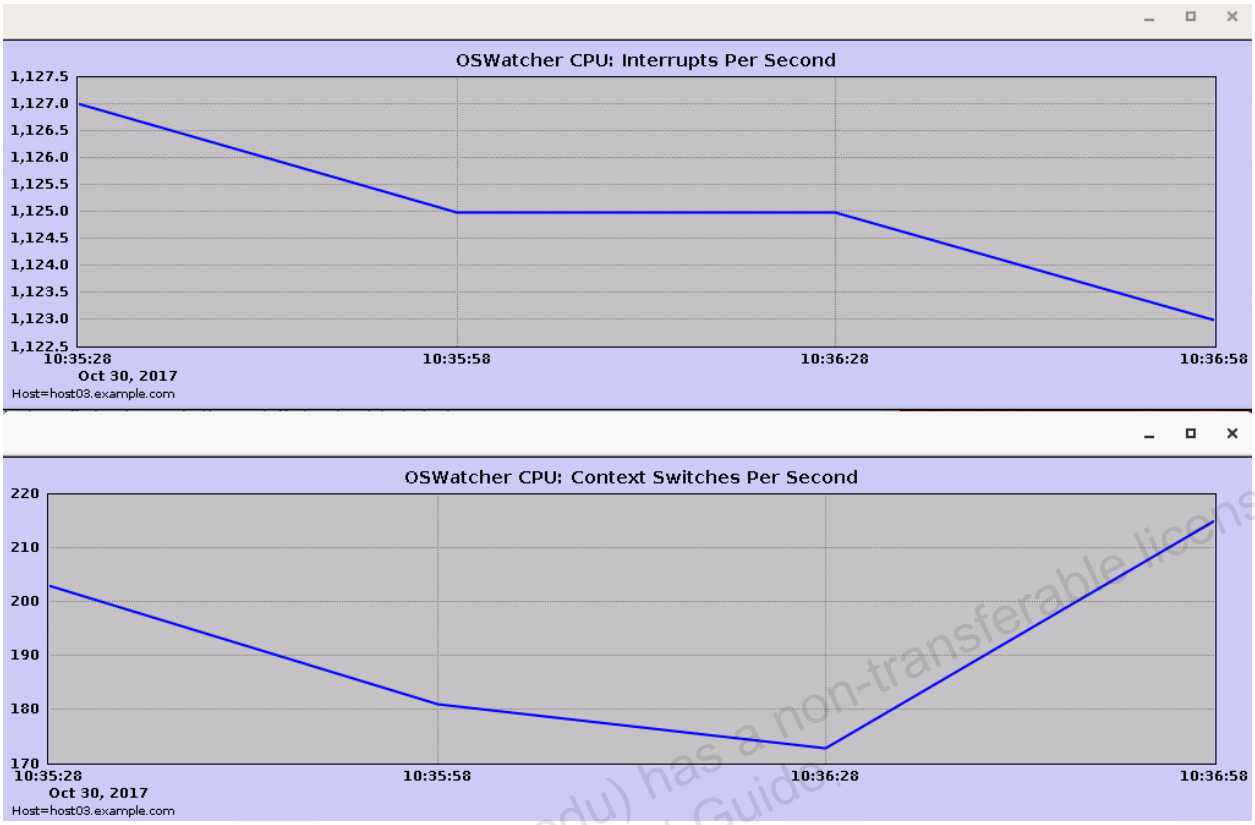
b. From the OSWbba menu, select 2 to display the four CPU Utilization graphs: System, Idle, Wait I/O, and User. View each graph by clicking each one from the window list at the bottom of the GNOME screen or drag and align the graphs to see them, as desired.





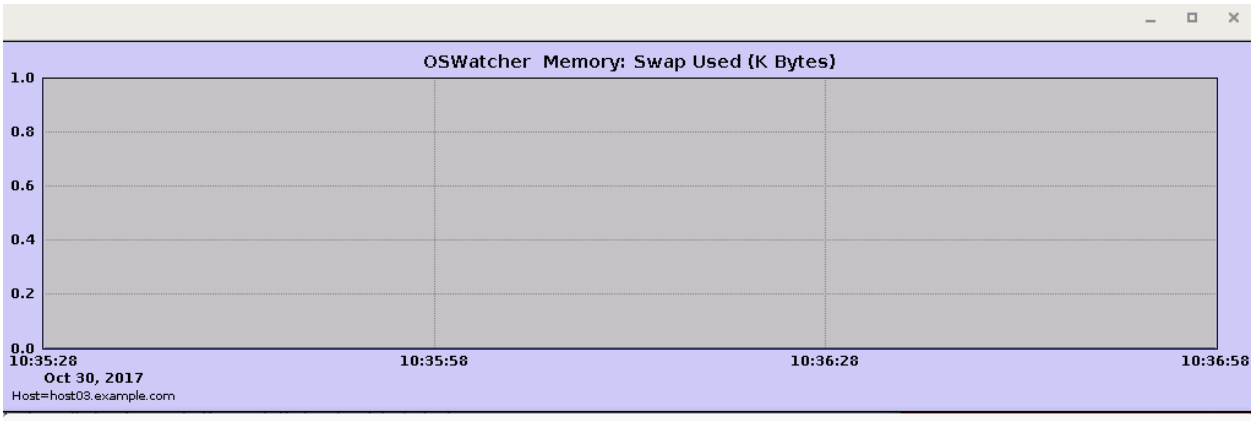
- After viewing the graphs, click the terminal window from the window list if necessary and select R from the menu to remove the graphs.

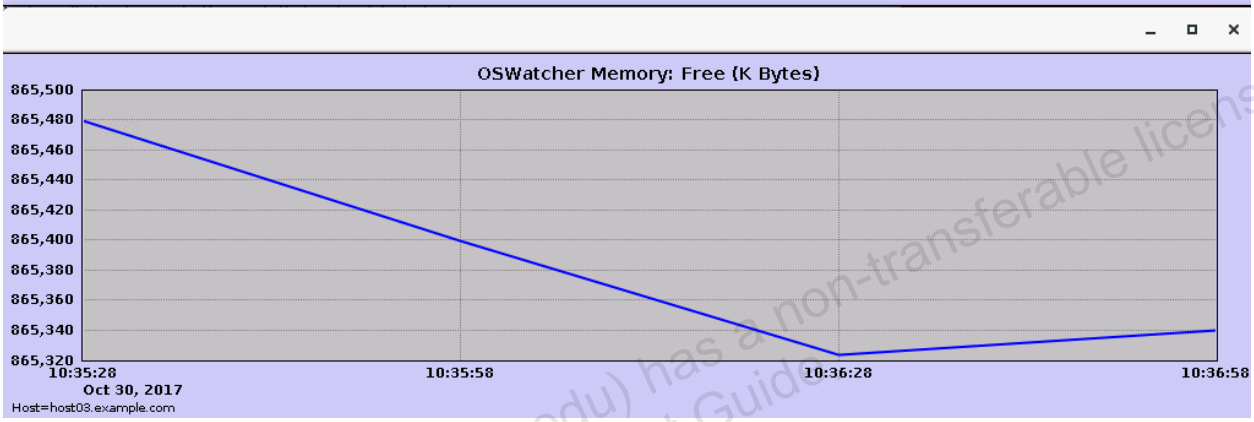
c. From the OSWbba menu, select 3 to display the two CPU Other graphs: Interrupts Per Second and Context Switches Per Second. View each graph by clicking each one from the window list at the bottom of the GNOME screen, or drag and align the graphs to see them, as desired.

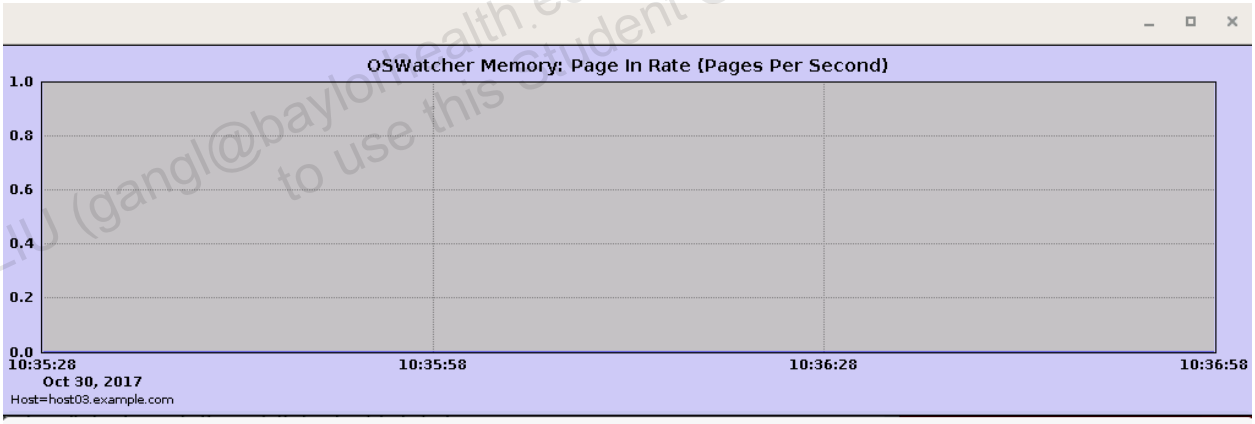


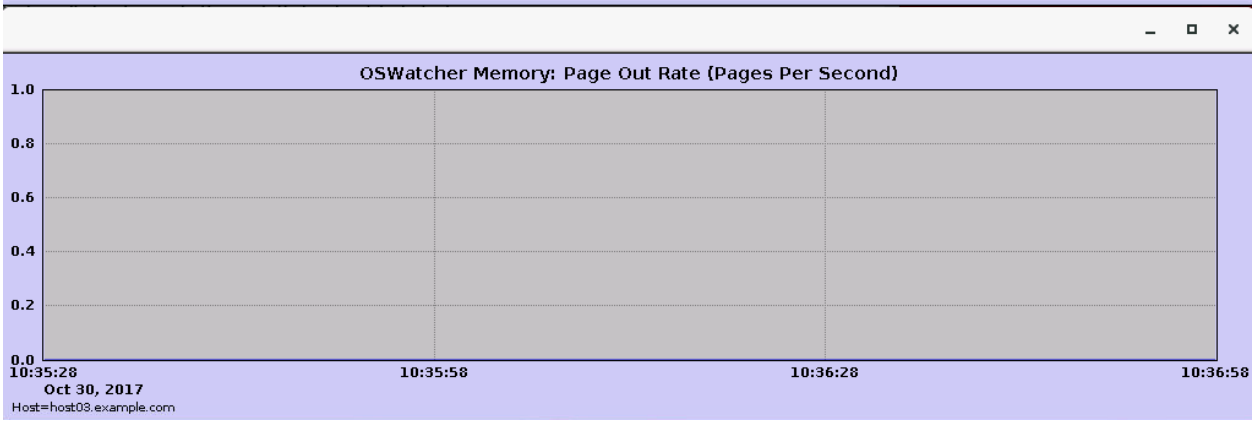
- After viewing the graphs, click the terminal window from the window list if necessary and select R from the menu to remove the graphs.

d. From the OSWbba menu, select 4 to display the four Memory Graphs: Swap Used, Free, Page In Rate, and Page Out Rate. View each graph by clicking each one from the window list at the bottom of the GNOME screen, or drag and align the graphs to see them, as desired.









- After viewing the graphs, click the terminal window from the window list if necessary and select R from the menu to remove the graphs.

3. Run the OSWbba analyzer and view the report.
   a. From the OSWbba menu, select A to analyze the collected data and produce a report. Provide a file name or press Enter to accept the default name.

```
...
Enter A to Analyze Data
...
Please Select an Option:a
Enter a unique analysis directory name or enter <CR> to accept
default name: ENTER
A new analysis file analysis/host03.../analysis.txt has been
created.
```

   - This message "A new analysis file analysis/host03.../analysis.txt has been created" appears, and the menu is then redisplayed.
   - You could alternatively run the analyzer from the command line by using the following command: (Do not run this command; this is information only.)

```
# java –jar oswbba.jar –i ~/oswbb/archive –A
```

   b. Exit the OSWbba menu by selecting the Q option to quit.
   c. From the **host03** command line, change to the directory where the `analysis.txt` file is located and use the `ls` command to display the analysis file in the `~/oswbb/analysis/host03...` directory.

```
# cd ~/oswbb/analysis/host03*
# ls
analysis.txt
```

   - The directory names where the `analysis.txt` file is found will vary.

   d. Use the `less` command to view the `analysis.txt` file.

   - Partial contents are displayed, showing initial information, followed by an outline of the contents of the file.

```
# less analysis.txt
This report is best viewed in a fixed font editor like
textpad...

OSWatcher Analyzer

Input Archive:       /root/oswbb/archive
Archive Source Dest: /root/oswbb/archive
Archive Begin Time:  ...
Archive End Time:    ...
Analysis Begin Time: ...
Analysis End Time:   ...
Hostname:            HOST03.EXAMPLE.COM
OS Version:          Linux
OSW Version:         v...
```

```
Snapshot Freq:        30
CPU COUNT:            1

################################################################
# Contents Of This Report:
#
# Section 1: System Status
# Section 2: System Slowdowns
#   Section 2.1: System Slowdown RCA Process Level Ordered By
Impact
# Section 3: System General Findings
# Section 4: CPU Detailed Findings
#   Section 4.1: CPU Run Queue:
#   Section 4.2: CPU Utilization: Percent Busy
#   Section 4.3: CPU Utilization: Percent Sys
# Section 5: Memory Detailed Findings
#   Section 5.1: Memory: Process Swap Queue
#   Section 5.2: Memory: Scan Rate
#   Section 5.3  Memory: Page In:
#   Section 5.4  Memory: Page Out:
#   Section 5.5  Memory: Page Tables (Linux only):
#   Section 5.6: Top 5 Memory Consuming Processes Beginning
#   Section 5.7: Top 5 Memory Consuming Processes Ending
# Section 6: Disk Detailed Findings
#   Section 6.1: Disk Percent Utilization Findings
#   Section 6.2: Disk Service Times Findings
#   Section 6.3: Disk Wait Queue Times Findings
#   Section 6.4: Disk Throughput Findings
#   Section 6.5: Disk Reads Per Second
#   Section 6.6: Disk Writes Per Second
#   Section 6.7: Disk Percent CPU waiting on I/O
# Section 7: Network Detailed Findings
#   Section 7.1  Network Data Link Findings
#   Section 7.2: Network IP Findings
#   Section 7.3: Network UDP Findings
#   Section 7.4: Network TCP Findings
# Section 8: Process Detailed Findings
#   Section 8.1: PS Process Summary Ordered By Time
#   Section 8.2: PS for Processes With Status = D or T Ordered
By Time
#   Section 8.3: PS for (Processes with CPU > 0) When System
Idle CPU < 30% Ordered By Time
```

```
#   Section 8.4: Top VSZ Processes Increasing Memory Per
Snapshot
#   Section 8.5: Top RSS Processes Increasing Memory Per
Snapshot
#   Section 8.6: New Processes Created Since Last Snapshot
#   Section 8.7: Exiting Processes Since Last Snapshot
#
...
```

e. After viewing the analysis file, press `q` to exit the `less` command.

4. Remove the `oswbb` files on **host03**.

a. Use the `/bin/rm -r` command to remove `oswbb` and its contents. Use the `cd` command to change to the home directory of the `root` user first.

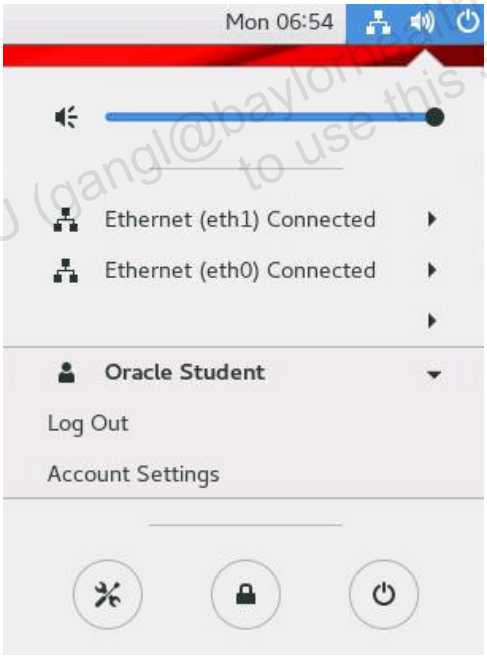
```
# cd
# /bin/rm -r oswbb
```

b. Use the `rm` command to remove `oswbb812.tar`.

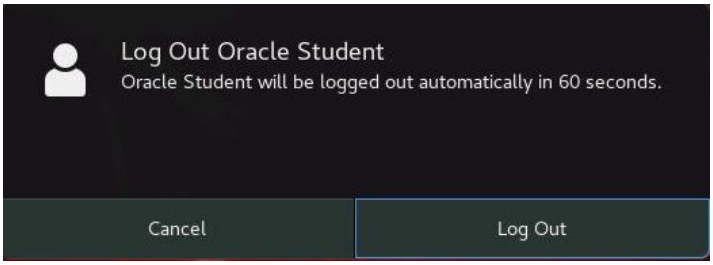
```
# rm oswbb812.tar
rm: remove regular file `oswbb812.tar'? y
```

5. Log off from **host03** in preparation for the next practice.

a. Select the Power icon in the upper right of the GNOME screen > select Oracle Student.



1) Click Log Out. The following pop-up appears:



2) Click Log Out.
3) Click the X in the top-right corner of the GNOME login window to close the window.

6. Terminate the `dd` command started previously.

   In the second terminal window on **host03,** in which you initiated the following `dd` command, press Ctrl + C to terminate the command.

```
# dd if=/dev/zero of=/dev/null bs=1024
CTRL-C
```

7. Log off from **host03** in preparation for the next practice.

   Use the `logout` command to close the `ssh` connection to **host03**.

```
# logout
Connection to host03 closed.
```

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

Practices for Lesson 19: System Logging

Practices for Lesson 19: Overview

Practices Overview

In these practices, you configure system logging, use `rsyslog` templates to format log messages, install and run `logwatch`, view the `journald` journal, configure persistent `journald` storage, start the process accounting service, `psacct`, and run process accounting utilities.

Assumptions

You created a local Yum repository in the practices for the lesson titled “Package Management.”

Practice 19-1: Configuring System Logging

Overview

In this practice, you view the system logging configuration file, modify the file, and observe the impact of the modifications. You also configure log file rotation.

Assumptions

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

Tasks

1. Log in to **host03**.

   Use the `ssh` command to log in to **host03** as the `root` user.

   Provide the `root` password when prompted.

```
[dom0]# ssh host03
root@host03's password:
Last login: ...
```

All remaining commands in this practice are entered from **host03**.

2. Explore the main configuration file for system logging, `/etc/rsyslog.conf`.

   a. Use the `less` command to view the system logging configuration file.

```
# less /etc/rsyslog.conf
...
#### MODULES ####
...
$ModLoad imuxsock   # provides support for local system...
$ModLoad imjournal   # provides access to the systemd journal
...
#### GLOBAL DIRECTIVES ####

# Where to place auxiliary files
$WorkDirectory /var/lib/rsyslog

# Use default timestamp format
$ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat
...
# Include all config files in /etc/rsyslog.d/
$IncludeConfig /etc/rsyslog.d/*.conf

# Turn off message reception via local log socket;
# local messages are retrieved through imjournal now.
$OmitLocalLogging on
```

```
# File to store the position in the journal
$IMJournalStateFile imjournal.state

#### RULES ####

# Log all kernel messages to the console.
# Logging much else clutters up the screen
#kern.*                                 /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none   /var/log/messages

# The authpriv file has restricted access.
authpriv.*                              /var/log/secure

# Log all the mail messages in one place.
mail.*                                  -/var/log/maillog

# Log cron stuff
cron.*                                  /var/log/cron
...
```

b. Use the up-arrow and down-arrow keys to view the various sections of the file.
   - The **MODULES** section uses the `$ModLoad` command to load the modules.
   - The **GLOBAL DIRECTIVES** section specifies configuration options.
   - The **RULES** section defines a selector (`facility.priority`) and an action.

c. Press the **q** key to exit the `less` command.

3. Change the action for `cron` logging.

   a. Use the `ls` command to list the `cron*` files in the `/var/log` directory.
      - You might not have any rotated log files, which are files with a date stamp.

```
# ls /var/log/cron*
/var/log/cron
/var/log/cron-2018...
...
```

   b. Use the `vi` editor to modify the system logging configuration file. Change the action for `cron` logging to log in to a different log file: `/var/log/cron_new`.

```
# vi /etc/rsyslog.conf
...
# Log cron stuff
cron.*            /var/log/cron                 (old entry)
cron.*            /var/log/cron_new             (new entry)
```

c. Use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

d. Use the `crontab -e` command to create a `cron` job that runs the `ls` command every minute.

```
# crontab -e
* * * * * ls
```

e. Use the `ls` command to list the `cron*` files in the `/var/log` directory.

```
# ls /var/log/cron*
/var/log/cron
/var/log/cron-2018...
...
/var/log/cron_new
```

- Note the new log file, `cron_new`.

f. Use the `tail` command to view the last two lines in the `cron` log file.

```
# tail -2 /var/log/cron
Oct 15 14:01:01 host03 run-parts(/etc/cron.hourly)[11907]: ...
Oct 15 14:10:01 host03 CROND[27301]: (root) CMD (/usr/lib64/...)
```

- This is sample output only.

g. Use the `head` command to view the first entries in the `cron_new` log file.

```
# head /var/log/cron_new
Oct 15 14:18:53 host03 crontab[32562]: (root) BEGIN EDIT (root)
Oct 15 14:19:02 host03 crontab[32562]: (root) REPLACE (root)
...
```

- Note from the date_time stamps that the new log entries are being written to `cron_new`.

h. Use the `vi` editor to modify the system logging configuration file. Change the action for `cron` logging back in to the original log file.

```
# vi /etc/rsyslog.conf
...
# Log cron stuff
cron.*              /var/log/cron_new              (old entry)
cron.*              /var/log/cron                  (new entry)
```

i. Use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

j. Use the `tail` command to ensure that `cron` is now logging in to `/var/log/cron`.

- This is sample output only.

```
# tail -1 /var/log/cron
Oct 15 14:28:01 host03 CROND[842]: (root) CMD (ls)
# tail -1 /var/log/cron_new
```

```
Oct 15 14:27:01 host03 CROND[758]: (root) CMD (ls)
```

- Note that the entry in `cron` has a later time stamp than the entry in `cron_new`.
- You might need to wait a minute for the `cron` job to run.

k. Use the `rm` command to delete the `cron_new` log file.

```
# rm /var/log/cron_new
rm: remove regular file '/var/log/cron_new'? y
```

l. Use `crontab -r` to remove the `crontab`.

```
# crontab -r
```

4. Configure `rsyslog` to log debug messages.

a. Use the `vi` editor to modify the system logging configuration file. Add an entry at the bottom of the file to log all debug messages to `/var/log/debug`.

```
# vi /etc/rsyslog.conf
...
*.debug                                /var/log/debug
```

b. Use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

c. Use the `logger` command to generate an **informational** log message.

- Logger is an interface to the `syslog(3)` system log module.
- Logger makes entries in the system log.

```
# logger -p info "This is an info-priority message"
```

d. Use the `tail` command to view the log files.

```
# tail /var/log/messages
...
Oct 15 14:31:05 host03 root: This is an info-priority message
# tail /var/log/debug
...
Oct 15 14:31:05 host03 root: This is an info-priority message
```

- Note that the message was written to both log files.

e. Use the `logger` command to generate a **debug-level** log message.

```
# logger -p debug "This is a debug-priority message"
```

f. Use the `tail` command to view the log files.

```
# tail /var/log/messages
...
# tail /var/log/debug
...
Oct 15 14:33:42 host03 root: This is a debug-priority message
```

- Note that the debug-level message was written only to `/var/log/debug`.

g. Use the `vi` editor to modify `/etc/rsyslog.conf` and remove the entry at the bottom of the file to log all debug messages.

```
# vi /etc/rsyslog.conf
...
*.debug                                /var/log/debug  (delete this entry)
```

h. Use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

5. Configure log file rotation.

a. Use the `ls` command to view the contents of the `/var/log` directory.

- This is sample output only.

```
# ls /var/log/messages*
/var/log/messages
/var/log/messages-2018...
...

# ls /var/log/maillog*
/var/log/maillog
/var/log/maillog-2018...
...

# ls /var/log/cron*
/var/log/cron
/var/log/cron-2018...
...
```

- Note that some files in `/var/log` have numbers at the end of the file name.
- These numbers represent a rotated log with the time stamp added to the log file name.
- You might not have log files with a time stamp appended to the file name. It depends on how long your system has been running.

b. Use the `vi` editor to modify the `/etc/logrotate.conf` configuration file. Change the frequency of the default log file rotation from **weekly** to **daily**.

```
# vi /etc/logrotate.conf
...
# rotate log files weekly
weekly                                          (old entry)
daily                                           (new entry)
```

- Your log files now rotate daily after making this change.

Practice 19-2: Using `rsyslog` Templates

Overview

In this practice, you use `rsyslog` templates to format `rsyslog` output.

Assumptions

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

Tasks

1. Define and use a template.
   a. Use the `vi` editor to modify `/etc/rsyslog.conf` and define a template.
      - Add the template definition line shown at the bottom of the file.
      - This entry creates a template named `class`.
      - Do not exit the `vi` editor.

```
# vi /etc/rsyslog.conf
...
$template class,"Message: %msg%\n"
```

   b. Continue editing `/etc/rsyslog.conf` and create a log file that uses the template.
      - Add the new line (shown in **bold**) after the entry that defined the template.
      - This entry writes all messages to the `/var/log/class.log` file and formats the entries by using the `class` template.
      - Exit the `vi` editor and save the file after adding the new line.

```
...
$template class,"Message: %msg%\n"
*.*   /var/log/class.log;class
```

   c. After saving the changes to `/etc/rsyslog.conf`, use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

   d. Use the `cat` command to view the `/var/log/class.log` file.

```
# cat /var/log/class.log
Message: ...
...
```

      - Note that all entries are preceded by the text "`Message:`" followed by the actual message, as defined in the `class` template.

2. Modify the `class` template.
   a. Use the `vi` editor to edit `/etc/rsyslog.conf` and modify the `class` template.
      - Change the template definition as shown.

```
# vi /etc/rsyslog.conf
...
```

```
$template class,"Message: %msg%\n"       (old entry)

$template class,"Time: %timestamp%, Facility: %syslogfacility-
text%, Priority: %syslogpriority-text%, Hostname: %hostname%,
Message: %msg%\n"                        (new entry)
```

b. After saving the changes to `/etc/rsyslog.conf`, use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

c. Use the `cat` command to view the `/var/log/class.log` file.

```
# cat /var/log/class.log
Message: ...
...
Time: ...
...
```

- Note that the newest entries now include the `Time`, `Facility`, `Priority`, `Hostname`, and `Message` properties, as defined in the `class` template.

3. Restore default configuration.

a. Use the `vi` editor to modify `/etc/rsyslog.conf` and:

- Delete the template definition entry
- Delete the rule that uses the template entry

```
# vi /etc/rsyslog.conf
...
$template class, "Time: %timestamp%, Facility: %syslogfacility-
text%, Priority: %syslogpriority-text%, Hostname: %hostname%,
Message: %msg%\n"
*.*    /var/log/class.log;class
```

b. After saving the changes to `/etc/rsyslog.conf`, use the `systemctl` command to restart the `rsyslog` service.

```
# systemctl restart rsyslog
```

c. Use the `rm` command to remove the `/var/log/class.log` file.

```
# rm /var/log/class.log
rm: remove regular file '/var/log/class.log'? y
```

Practice 19-3: Using `logwatch`

Overview

In this practice, you install the `logwatch` package, view the main configuration file (the `cron` file), and run the `logwatch` utility from the command line.

Assumptions

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

Tasks

1. Install the `logwatch` package.
   a. Use the `rpm` command to determine whether the `logwatch` package is already installed.

```
# rpm -q logwatch
package logwatch is not installed
```

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

   b. Use the `yum` command to install the `logwatch` package.
   - Answer **`y`** when prompted.

```
# yum install logwatch
...
Transaction Summary
================================================
Install    1 Package (+3 Dependent packages)
Total download size: 1.6 M
Installed size: 14 M
Is this ok [y/d/N]: y
...
Installed:
  logwatch.noarch ...

Dependency Installed:
...

Complete!
```

2. View `logwatch` files.
   a. Use the `find` command to list all `logwatch` files.
   - Only partial output is displayed.

```
# find / -name "*logwatch*"
...
/usr/sbin/logwatch
```

```
...
/usr/share/doc/logwatch-7.4.0
...
/usr/share/logwatch/default.conf/logwatch.conf
...
/etc/cron.daily/0logwatch
...
/etc/logwatch/conf/logwatch.conf
...
```

b. Use the `less` command to view the `logwatch` configuration file.

```
# less /usr/share/logwatch/default.conf/logwatch.conf
...
```

- Note various configurable items such as the following:
  - `LogDir`
  - `TmpDir`
  - `MailTo`
  - `MailFrom`
  - `Range`
  - `Detail`
  - `Service`

c. Use the `less` command to view the `logwatch cron` file.

```
# less /etc/cron.daily/0logwatch
#!/bin/sh

#Set logwatch location
LOGWATCH_SCRIPT="/usr/sbin/logwatch"
#Add options to this line. Most options should be defined in
/etc/logwatch/conf/logwatch.conf,
#but some are only for the nightly cronrun such as --output mail
and should be set here.
#Other options to consider might be "--format html" or "--encode
base64", man logwatch for more details.
OPTIONS="--output mail"

#Call logwatch
$LOGWATCH_SCRIPT $OPTIONS

exit 0
```

3. Run `logwatch` from the command line.

a. Run the `logwatch --help` command to view the `logwatch` help.

```
# logwatch --help
```

```
Usage: /usr/sbin/logwatch [--detail <level] [--logfile <name>]
   [--output <output_type>]
...
```

b. Run `logwatch` with the `range` option and a date_range parameter of `today`. This will process log data from the current day.

- Sample output is displayed.

```
# logwatch --range today

################## logwatch 7.4.0 (03/01/11) ##################
       Processing Initiated: Mon Oct...
       Date Range Processed: today
...
       Logfiles for Host: host03.example.com
################################################################
...
----------------------SSHD Begin ------------------------

SSHD Started: 2 Time(s)

 Users logging in through sshd:
    root:
       192.0.2.1 (example.com): 7 times
...
####################### Logwatch End #########################
```

4. Remove the `logwatch` package.

Use the `yum` command to remove the `logwatch` package.

- Answer **`y`** when prompted.

```
# yum remove logwatch
...
Transaction Summary
================================================================
Remove  1 Package

Installed size: 1.9 M
Is this ok [y/N]: y
...
Removed:
  logwatch.noarch ...

Complete!
```

Practice 19-4: Using `journald`

Overview

In this practice, you use the `journalctl` command to query the `systemd` journal, view `journald` metadata, and enable persistent `journald` storage. Your log output might vary.

Assumptions

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

Tasks

1. Use the `journalctl` command to query the `systemd` journal.
   a. Run `journalctl` with no options or arguments.

```
# journalctl
-- Logs begin at ..., end at ...
<date_time> host03.example.com systemd-journal[98]: ...
<date_time> host03.example.com kernel: Initializing cgroup ...
...
<date_time> host03.example.com kernel: Linux version 4.1.12-...
<date_time> host03.example.com kernel: Command line: BOOT_IM...
<date_time> host03.example.com kernel: e820: BIOS-provided ...
<date_time> host03.example.com kernel: BIOS-e820 [mem 0x...
...
<date_time> host03.example.com kernel: vbd vbd-5632: 19 xen...
...
```

   - Note that all log data is displayed, including rotated logs.
   - Note that the beginning of the boot process is indicated with a special entry.
   - Note that entries with error priority and higher are in red.
   - Note that entries with notice and warning priority are in bold font (which might not be visible in your terminal window; opening a terminal window in the GNOME GUI will show the bold font).

   b. Run the `journalctl -h` command to display usage and query options.

```
# journalctl -h
journalctl [OPTIONS...] [MATCHES...]
Query the journal.
Flags:
     --system        Show only the system journal
     --user          Show only the user journal for the curr...
     -M --machine=CONTAINER   Operate on local container
     -S --since=DATE Show entries not older than ...
     -U --until=DATE Show entries not newer than ...
...
```

```
  -k --dmesg          Show kernel message log from the curren...
  -u --unit=UNIT      Show logs from the specified unit
...
  -p –priority=RANGE Show entries with the specified priority
...
```

c. Run the `journalctl -r` command to display the newest log entries first.

```
# journalctl -r
-- Logs begin at ..., end at ...
<date_time> host03.example.com yum[5971]: Erased: logwatch...
<date_time> host03.example.com CROND[5719]: (root) CMD (/usr...
<date_time> host03.example.com systemd[1]: Starting session ...
<date_time> host03.example.com systemd[1]: Started session ...
...
<date_time> host03.example.com dbus[586]: [system] Successful...
<date_time> host03.example.com dbus[586]: [system] Activat...
...
```

d. Run the `journalctl -n 3` command to display the three newest log entries.

- Using the `-n <number>` option displays a specific `<number>` of the most recent log entries.

```
# journalctl –n 3
-- Logs begin at ..., end at ...
<date_time> host03.example.com systemd[1]: Started session ...
<date_time> host03.example.com systemd[1]: Starting session ...
<date_time> host03.example.com CROND[6319]: (root) CMD (/usr...
```

e. Run the `journalctl -p crit` command to display log entries with a priority of `crit`, `alert`, or `emerg`.

- You can use the `-p` option to display log entries of any priority. Valid priorities are `debug`, `info`, `notice`, `warning`, `err`, `crit`, `alert`, and `emerg`.
- Without specifying a range with the "`-p`" option, log entries at the given level and those with greater significance are shown (a range has the form `FROM..TO`, for example, `crit..alert`).

```
# journalctl –p crit
-- Logs begin at ..., end at ...
<date_time> host03.example.com smartd[561]: DEVICESAN failed ...
<date_time> host03.example.com smartd[561]: In the system's ...
```

f. Run the `journalctl -u crond` command to display log entries associated with the `crond` unit.

- You can use the `-u` option to display log entries for any `systemd` unit.

```
# journalctl –u crond
-- Logs begin at ..., end at ...
```

```
<date_time> host03.example.com systemd[1]: Started Command ...
<date_time> host03.example.com systemd[1]: Starting Command ...
<date_time> host03.example.com crond[1338]: (CRON) INFO (RAN...
<date_time> host03.example.com crond[1338]: (CRON) INFO (runn...
```

2. View `journald` metadata.

   - `journald` adds structured metadata to the messages, which assists in troubleshooting.

   a. Type the `journalctl` command, add a space, and then press the Tab key twice to display the metadata fields. After displaying the metadata fields, type CTRL + C to terminate the command.

```
# journalctl <TAB> <TAB>
_AUDIT_LOGINUID=  _HOSTNAME=             SYSLOG_IDENTIFIER=
_AUDIT_SESSION=   _KERNEL_DEVICE=        SYSLOG_PID=
_BOOT_ID=         _KERNEL_SUBSYSTEM=     _SYSTEMD_CGROUP=
_CMDLINE=         MACHINE_ID=            _SYSTEMD_OWNER_UID=
CODE_FILE=        MESSAGE=               _SYSTEMD_SESSION=
CODE_FUNC=        MESSAGE_ID=            _SYSTEMD_UNIT=
CODE_LINE=        __MONOTONIC_TIMESTAMP= _TRANSPORT=
_COMM=            _PID=                  _UDEV_DEVLINK=
COREDUMP_EXE=     PRIORITY=              _UDEV_DEVNODE=
__CURSOR=         __REALTIME_TIMESTAMP=  _UDEV_SYSNAME=
ERRNO=            _SELINUX_CONTEXT=      _UID=
_EXE=             _SOURCE_REALTIME_TIMESTAMP=
_GID=             SYSLOG_FACILITY=
# journalctl <CTRL-C>
#
```

   b. Run the `journalctl -o verbose` command to display log entries in `verbose` format.

      - `Verbose` format shows the metadata fields and values for all journal entries.
      - You can use the `-o` option to display log entries in any supported format. Supported format options are `short`, `short-iso`, `short-precise`, `short-monotonic`, `verbose`, `export`, `json`, `json-pretty`, `json-see`, and `cat`.

```
# journalctl -o verbose
-- Logs begin at ..., end at ...
<date_time>
    PRIORITY=6
    _TRANSPORT=driver
    MESSAGE=Runtime journal is using 8.0M (max 99.8M, trying...
    MESSAGE_ID=...
    _PID=98
    _UID=0
    _GID=0
```

```
    _COMM=systemd-journal
    _EXE=/usr/lib/systemd/systemd-journald
...
```

c. Run the `journalctl -F _UID` command to display unique values for the `_UID` metadata field.

- Sample output is shown.

```
# journalctl -F _UID
997
42
99
81
172
999
70
0
```

d. Run the `journalctl _UID=<value>` command to show only log entries that match the condition.

- This example uses `172` as `<value>`. Substitute a valid `UID` value output from the previous command.

```
# journalctl _UID=172
-- Logs begin at ..., end at ...
<date_time> host03.example.com rtkit-daemon[609]: Successful...
<date_time> host03.example.com rtkit-daemon[609]: Successful...
<date_time> host03.example.com rtkit-daemon[609]: Running.
<date_time> host03.example.com rtkit-daemon[609]: Watchdog ...
<date_time> host03.example.com rtkit-daemon[609]: Canary ...
...
```

3. Enable persistent `journald` storage.

a. Use the `mount` command, pipe the output to `grep`, and search for the string "`run`".

```
# mount | grep run
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,mode=755)
...
```

- Note that the file system type mounted on `/run` is `tmpfs`.

b. Use the `ls -l` command to view the `/run/log/journal/` directory.

```
# ls -l /run/log/journal
drwxr-s---+ ... <date_time> ...
```

- By default, journal data is stored in this directory.
- Note the date_time stamp on this directory.

c. Use the `mkdir -p` command to make the `/var/log/journal` directory.

```
# mkdir -p /var/log/journal
```

d. Use the `systemctl` command to restart the `systemd-journald` service.

```
# systemctl restart systemd-journald
```

e. Use the `ls -l` command to view the `/var/log/journal` directory.

```
# ls –l /var/log/journal
drwxr-xr-x. ... <date_time> ...
```

- Journal data is now stored in this directory.
- Note the `date_time` stamp on this directory is more recent than the `date_time` stamp on `/run/log/journal`.

Practice 19-5: Using Process Accounting

Overview

In this practice, you check for the presence of the process accounting file, /var/account/pacct. You also check for the presence of the /var/log/wtmp file and start the `psacct` service to enable process accounting. You check the size of the pacct file and run some process accounting utilities. The `psacct` service is then stopped.

Assumptions

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

Tasks

1. Check for the presence of files used by process accounting.
   a. Use the `ls` command to verify the presence of /var/account/pacct.

```
# ls -l /var/account/pacct
-rw-------. 1 root root     0 <date_time> pacct
```

   - Note that the pacct file is present with a zero size. The pacct file is the default process accounting file.

   b. Use the `ls` command to verify the presence of /var/log/wtmp.

```
# ls -l /var/log/wtmp
-rw-rw-r--. 1 root utmp 70272 <date_time> /var/log/wtmp
```

   - Note that the wtmp file is present and has a non-zero size. The wtmp file records all logins and logouts.

2. Start process accounting.
   a. Use the `systemctl` command to start the `psacct` service.

```
# systemctl start psacct
```

   b. Check the status of the `psacct` service.

```
# systemctl status psacct
● psacct.service - Kernel process accounting
   Loaded: loaded (/usr/lib/systemd/system/psacct.service;
disabled; vendor preset: disabled)
   Active: active (exited) since <date_time; time> ago
  Process: 14546 ExecStart=/usr/sbin/accton /var/account/pacct
(code=exited, status=0/SUCCESS)
  Process: 14544 ExecStartPre=/usr/libexec/psacct/accton-create
(code=exited, status=0/SUCCESS)
 Main PID: 14546 (code=exited, status=0/SUCCESS)

<date_time> host03.example.com systemd[1]: Starting Kernel
process accounting...
```

```
<date_time> host03.example.com accton[14546]: Turning on process
accounting, file set to ...'.
<date_time> host03.example.com systemd[1]: Started Kernel
process accounting.
Hint: Some lines were ellipsized, use -l to show in full.
```

- Note that the `psacct` service is active and that the `accton` command was called with the process accounting file, /var/account/pacct, passed as an argument.

c. Check the size of the /var/account pacct file.

```
# ls -l /var/account/pacct
-rw-------. 1 root root 2112 <date_time> /var/account/pacct
```

- Note that the pacct file now has a non-zero size.

3. Run process accounting utilities. Your output will vary.

a. Use the `ac` command with no arguments to view the total connect time for all users in hours. Recall that `ac` uses `/var/log/wtmp` as input.

```
# ac
        total     3672.30
```

b. Use the `ac` command with the `-p` option to view totals for individual users.

```
# ac -p
        student1                                      0.45
        oracle                                     3432.87
        root                                        238.98
        total     3672.30
```

c. Use the `ac` command with the `-d` option to view totals by day.

```
# ac -d
...
Aug 30  total        8.65
Sep  4  total       10.64
Sep  5  total       28.61
Today   total       12.41
```

d. Use the `dd` command to use some CPU time, while not retaining the output.

```
# dd if=/dev/zero of=/dev/null bs=500M count=10
10+0 records in
10+0 records out
5242880000 bytes (5.2 GB) copied, 0.65634 s, 8.0 GB/s
```

e. Use `lastcomm` to view information about the `dd` command you just ran. `lastcomm` uses the default process accounting file (/var/account/pacct) as input by default. Your output will vary.

```
# lastcomm dd
dd                 root     pts/1      0.65 secs Fri Sep  7 04:50
```

- Fields shown here include the command name (`dd`), the user who executed the command (`root`), the terminal used (`pts/1`), CPU seconds used, and the time the command (process) exited. Flags can follow the command name, but none are shown in this example. In this case, the `dd` command took 0.65 seconds of CPU time.

4. Turn off process accounting.

   a. Use the `systemctl` command to turn off process accounting.

```
# systemctl stop psacct
```

   b. Check the status of the `psacct` service.

```
# systemctl status psacct
● psacct.service - Kernel process accounting
   Loaded: loaded (/usr/lib/systemd/system/psacct.service;
disabled; vendor preset: disabled)
   Active: inactive (dead)
...
<date_time> host03.example.com systemd[1]: Stopping Kernel
process accounting...
<date_time> host03.example.com accton[18993]: Turning off
process accounting.
<date_time> host03.example.com systemd[1]: Stopped Kernel
process accounting.
Hint: Some lines were ellipsized, use -l to show in full.
```

   - Note that the `psacct` service is now inactive.

5. Log off from **host03** in preparation for the next practice.

   Use the `logout` command to close the `ssh` connection to **host03**.

```
# logout
Connection to host03 closed.
```

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

Practices for Lesson 20: Troubleshooting

Practices for Lesson 20: Overview

Practices Overview

In these practices, you perform a variety of troubleshooting exercises and fix some common problems. Specific problems you encounter include the following:

- System boots into single-user mode by default.
- Status utilities are not producing expected output.
- A `cron` job fails to run.
- A user cannot log in.
- File system does not mount.
- You cannot `ping` remote hosts.
- You cannot log in to remote hosts using `ssh`.
- Log file is not getting updated.

Each practice:

- Tells you what problem the activity is designed to simulate
- Tells you how to set up the problem
- Gives you some hints or things to check
- Has an associated “solution” that provides steps to diagnose and resolve the problem

Each solution:

- Includes only the steps to resolve the problem
- Does not include steps to set up the problem

Appendix:

- An appendix exists, which provides the source code for the executables that cause the problems that you need to troubleshoot and fix.

Practice 20-1: Transferring Utilities from dom0

Overview

In this practice, you transfer executable files hosted on **dom0** to **host01**. You are instructed to run one of these executables that causes the problem that you need to troubleshoot and fix.

Assumptions

You are logged in to **dom0** as the `root` user.

Tasks

1. Use the `scp` command to copy files from **dom0** to **host01**.

   Provide the `root` password when prompted.

```
[dom0]# scp /OVS/seed_pool/ts_scripts/* host01:~
root@host01's password:
eight.x                                   100%   11KB  11.0KB/s   00:00
five.x                                    100%   11KB  11.1KB/s   00:00
four.x                                    100%   11KB  11.0KB/s   00:00
nine.x                                    100%   11KB  11.1KB/s   00:00
seven.x                                   100%   11KB  11.3KB/s   00:00
six.x                                     100%   11KB  11.1KB/s   00:00
three.x                                   100%   11KB  11.0KB/s   00:00
two.x                                     100%   11KB  11.1KB/s   00:00
```

2. Log in to **host01**.

   a. From **dom0**, connect to the **host01** guest by using the `xm vncviewer host01&` command.

```
[dom0]# xm vncviewer host01&
```

   b. Log in as `root`. Provide the password when prompted.

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

Practice 20-2: System Boots into Single-User Mode

Overview

In this practice, your system boots into single-user mode by default, which is not what you want.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

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

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

   - The `multi-user.target` unit corresponds to run level 3 on a SysV init system.

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

```
# runlevel
5 3
```

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

2. Execute the `two.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
# ./two.x
```

3. Reboot the system.
   a. Use the `systemctl reboot` command to reboot your system.
   - It might take a few minutes for the reboot to complete.

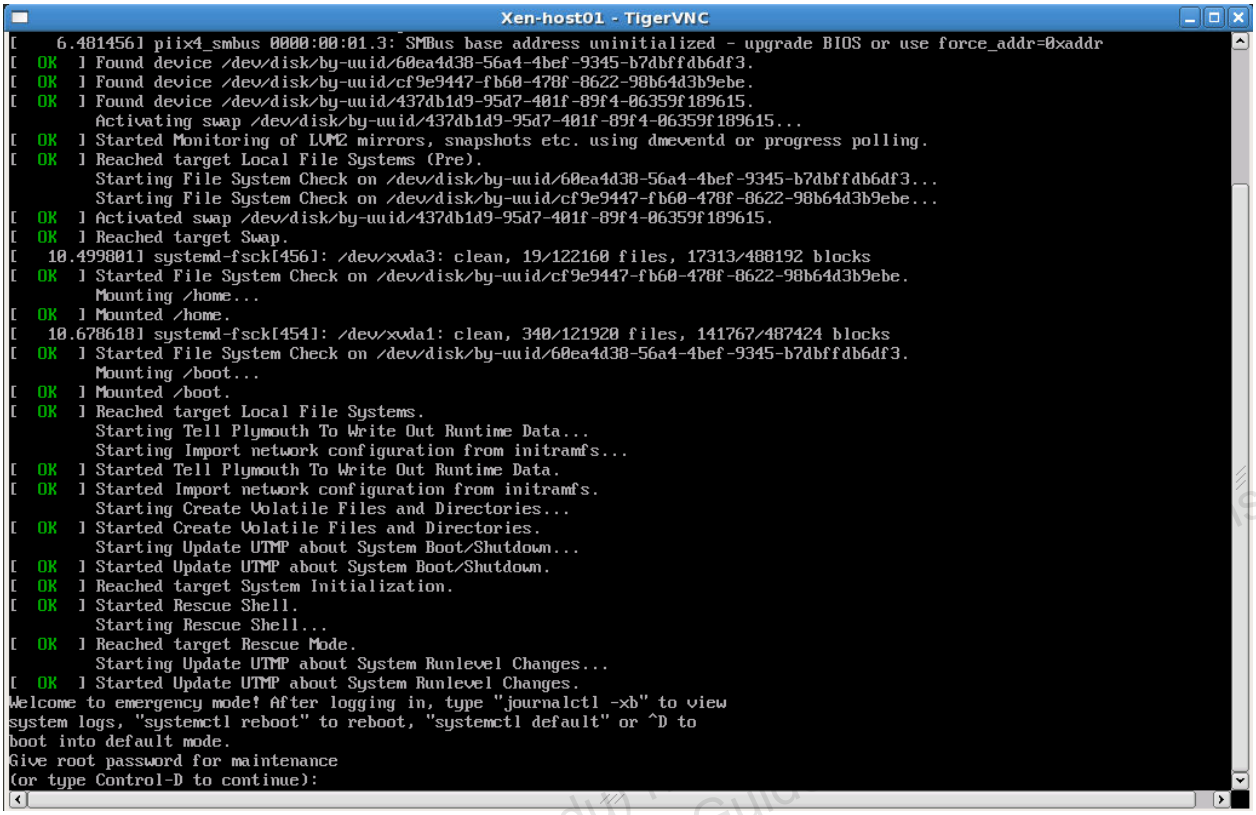
```
# systemctl reboot
...
```

   - After you reboot your system, your `ssh` session closes.

   b. From **dom0**, connect to **host01** by using the `xm vncviewer host01&` command.

```
# xm vncviewer host01&
```

   - The following window appears:



- Note that the system has “Reached target Rescue Mode.” In Oracle Linux 7, rescue mode is the same as single-user mode.

c. Enter the `root` password for maintenance.

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

d. 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`).

4. Diagnose and fix the problem so that the system does not boot into single-user mode.
   - Review the lesson titled “Oracle Linux 7 Boot Process.”
     - Which file allows you to view kernel boot parameters? Is `single` a kernel boot parameter?
     - Which file allows you to specify kernel boot parameters? Is `single` specified as a kernel boot parameter?
     - How do you change kernel boot parameters at boot time?
     - How do you permanently change kernel boot parameters?
5. When the problem is fixed, reboot the system and log in to **host01** using `ssh` to confirm it does not boot into single-user mode.

Solution 20-2: System Boots into Single-User Mode

Steps

1. Determine why the system boots into single-user mode.
   a. Use the `cat` command to view the kernel boot parameters in the `/proc/cmdline` file.

```
# cat /proc/cmdline
BOOT_IMAGE=/vmlinuz-4.1.12-112.16.4.el7uek.x86_64 root=UUID=...
ro rhgb quiet LANG=en_US.UTF-8 single
```

   - Note the word `single` is a kernel boot parameter that is causing your system to boot into single-user mode.

   b. Use the `grep` command and search for the word `single` in the `/boot/grub2/grub.cfg` file.

```
# grep single /boot/grub2/grub.cfg
linux16 /vmlinuz-4.1.12-112.16.4.el7uek.x86_64... single
```

   - Note the word `single` appears on the kernel line for the `4.1.12-112.16.4` kernel.
   - This would cause the `4.1.12-112.16.4` kernel to boot into single-user mode by default.

   c. Use the `uname -r` command to determine which kernel is running.

```
# uname -r
4.1.12-112.16.4.el7uek.x86_64
```

   - The `4.1.12-112.16.4` kernel is running. The word `single` needs to be removed from the associated kernel line in the `/boot/grub2/grub.cfg` file.

2. Fix the system so it does not boot into single-user mode.
   a. Use the `vi` editor to modify `/boot/grub2/grub.cfg` and remove the word `single` at the end of the `linux16 /vmlinuz-4.1.12-112.16.4` kernel line.

   - Note that a backup copy of the `grub.cfg` file was created by running the `two.x` script.
   - The backup copy is `/boot/grub2/grub.cfg.save`. As an alternative to editing `grub.conf`, you can copy `grub.cfg.save` to `grub.cfg`.

```
# vi /boot/grub2/grub.cfg
...
linux16 /vmlinuz-4.1.12-112.16.4.el7uek.x86_64... ~~single~~
...
```

   b. Press Ctrl + D to continue the boot process.

   - The login prompt appears after the boot process completes. (You may need to press the Enter key.)

```
# CTRL-D
...
host01 login:
```

c. Log in as `root`. Provide the password when prompted.

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

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

```
# runlevel
1 3
```

- Note the run level is `3`. The previous run level was `1`.

3. Confirm the system does not boot into single-user mode.

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

- It might take a few minutes for the reboot to complete.

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

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

b. Connect to **host01** by using `ssh`.

- Wait a few seconds for the reboot to complete.
- Provide the password when prompted.

```
[dom0]# ssh host01
root@host01's password:
```

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

```
# runlevel
N 3
```

- Note that the run level is now `3` and not single-user.

Practice 20-3: Status Commands Fail

Overview

In this practice, you note that some of the “status” utilities are not producing the expected output. You diagnose and fix this problem.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Execute the `three.x` program from the `root` user’s home directory.

```
# cd
# pwd
/root
# ./three.x
```

2. Run some “status” tools and note the errors.
   a. Run the `mpstat` command.

```
# mpstat
Linux 4.1.12-112.16.4...
Cannot open /proc/stat: No such file or directory
```

   b. Run the `iostat` command.

```
# iostat
Linux 4.1.12-112.16.4...
Cannot open /proc/stat: No such file or directory
```

   c. Run the `netstat` command.

```
# netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address Foreign Address State
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags    Type    State   I-Node Path
```

   d. Run the `route` command.

```
# route
/proc/net/route: No such file or directory
INET (IPv4) not configured in this system.
```

e. Run the `ifconfig` command.

```
# ifconfig
Warning: cannot open /proc/net/dev (No such file or directory).
Limited output.
...
```

- Note that none of these tools produce the expected output.

3. Diagnose and fix the problem with the “status” tools.
   - Review the lesson titled “System Configuration.”
     - Commands that are failing are commands that provide information about the current state of the kernel.
     - Which file system contains a hierarchy of special files that represent the current state of the kernel?
     - How do you check if this file system is mounted?
     - If this file system is not mounted, how do you mount it?
4. Re-run some of the earlier commands that failed, to ensure that the problem was fixed.
   a. Run the `mpstat` command.
   b. Run the `iostat` command.
   c. Run the `netstat` command.
   d. Run the `route` command.
   e. Run the `ifconfig` command.

Solution 20-3: Status Commands Fail

Steps

1. Some of the output indicates a problem with the `proc` file system. Diagnose the problem.
   a. Use the `ls` command to display the contents of `/proc`.

```
# ls /proc
```

   - No output from this command suggests the `proc` file system is not mounted.

   b. Run the `mount` command to display mounted file systems.

```
# mount
mount: failed to read mtab: No such file or directory
```

   c. Run the `df -h` command to display mounted file systems.

```
# df -h
df: cannot read table of mounted file systems: No such file or
directory
```

2. Fix the problem.
   - It appears that the `/proc` file system is not mounted.

   Use the `mount proc /proc -t proc` command to mount the `proc` file system.

   - The `t` option explicitly provides the file system type of `proc` to the `mount` command.

```
# mount proc /proc -t proc
```

3. Re-run some of the earlier "status" commands to ensure that the problem is fixed.
   a. Run the `mpstat` command.
   b. Run the `iostat` command.
   c. Run the `netstat` command.
   d. Run the `route` command.
   e. Run the `ifconfig` command.

Practice 20-4: `cron` Job Fails to Run

Overview

In this practice, you diagnose and fix a problem that is preventing a `cron` job from running.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Create a `crontab` for the `root` user.
   a. Use the `crontab -e` command to create a `crontab` job that runs the `vmstat` command every minute.

```
# crontab -e
* * * * * vmstat
```

   - The `crontab -e` command uses the `vi` editor. Save your changes and exit `vi`.

   b. After one minute, use the `tail` command to view the last few lines in the `/var/log/cron` log file.

```
# tail /var/log/cron
...
<date_time> host01 CROND[...] (root) CMD (vmstat)
```

   - Note that there is an entry in this file that states the date and time that the `cron` daemon ran the `vmstat` command.
   - You also have mail, because the output from `cron` jobs is sent to the user's mailbox.

   c. Use the `mail` command to view the results of your job.

```
# mail
...
N  # (Cron Daemon)  <date_time> ... "Cron <root@host01> vmstat"
&
```

   - Note you have mail that contains the output of the `vmstat` command.

   d. View the details of mailbox entries by entering the associated number and then press **Enter**.

   - To re-display the header, press **h** and then press **Enter**.
   - To quit the mail program, press **q** and then press **Enter**.

   e. Press **q** and then **Enter** to quit the mail program.

```
& q
```

2. Execute the `four.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
```

```
# ./four.x
```

3. Edit the `crontab` for the `root` user.
   a. Use the `crontab -e` command to edit the `crontab` job and replace `vmstat` with `iostat`.

```
# crontab -e
* * * * * vmstat                    (old entry)
* * * * * iostat                    (new entry)
```

   b. After one minute, use the `tail` command to view the last few lines in the `/var/log/cron` log file.

```
# tail /var/log/cron
...
```

   - Note that there is not an entry in this file stating that the `iostat` command ran.

   c. Use the `mail` command to view your incoming mailbox.

```
# mail
...
&
```

   - Note that there is no mail entry containing the output of the `iostat` command.

   d. Press **q** and then **Enter** to quit the mail program.

```
& q
```

4. Diagnose and fix the problem of the `cron` job failing to run.
   - Review the lesson titled "Automating Tasks."
     - Does the `root` user have a `crontab` entry?
     - Is the `root` user's `crontab` entry valid?
     - Are the permissions correct on the `root` user's `crontab` file?
     - Is the `cron` daemon running?
5. Verify that the `cron` job is running every minute.
   a. Run the `mail` command to view the output of the `iostat` command.
   b. Use the `tail` command to view `/var/log/cron` and ensure the `cron` job is running every minute.
6. Remove the `root` user's `crontab` before continuing to the next practice.

```
# crontab -r
```

Solution 20-4: `cron` Job Fails to Run

Steps

1. Diagnose the problem of the `cron` job failing to run.
   a. Use the `crontab -l` command to list the `crontab` entry.

```
# crontab -l
* * * * * iostat
```

   - Note that the `crontab` entry exists and the format is valid.

   b. Use the `ls -l` command to view the root user's `crontab` entry in the `/var/spool/cron` directory.

```
# ls -l /var/spool/cron
-rw-------. root root ... root
```

   - The `root` file exists with the proper permissions (read-write for the owner).

   c. Use the `cat` command to view the `/var/spool/cron/root` file.

```
# cat /var/spool/cron/root
* * * * * iostat
```

   - This confirms the `crontab` entry is valid.

   d. Use the `systemctl` command to view the status of the `crond` daemon.

```
# systemctl status crond
crond.service - Command Scheduler
   Loaded: loaded (/usr/lib/systemd/system/crond.service: ...
   Active: inactive (dead) since ...
...
```

   - Note that the `crond` service is not running. This is the cause of the problem.

2. Fix the problem of the `cron` job failing to run.
   a. Use the `systemctl` command to start the `crond` daemon.

```
# systemctl start crond
```

3. Verify that the `cron` job is running every minute.
   a. After one minute, use the `tail` command to view the last few lines in the `/var/log/cron` log file.

```
# tail /var/log/cron
...
<date_time> host01 CROND[...] (root) CMD (iostat)
```

   - Note that there is now an entry in this file that states the date and time that the `cron` daemon ran the `iostat` command.

b. Use the `mail` command to view the results of your job.

```
# mail
...
N # (Cron Daemon)  <date_time> ... "Cron <root@host01> iostat"
```

- Note that you have mail that contains the output of the `iostat` command.

c. Press **q** and then **Enter** to quit the mail program.

```
& q
```

4. Remove the `root` user's `crontab` before continuing to the next practice.

```
# crontab -r
```

Practice 20-5: User Cannot Log In

Overview

In this practice, you diagnose and fix a problem that is preventing a user from being able to log in.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Add a new user.
   a. Use the `useradd john` command to add a user `john`.

```
# useradd john
```

   b. Use the `passwd john` command to assign a password (`password`) to the user `john`.
      - Disregard the "BAD PASSWORD" warning.

```
# passwd john
Changing password for user john.
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.
```

2. Verify you can log in as user `john`.
   a. Use the `exit` command to log out as user `root`.

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

      - The `ssh` connection closes.
   b. From **dom0**, use the `ssh` command to log in to **host01** as user `john`.
      - Password is `password`. Verify you can successfully log in as user `john`.

```
[dom0]# ssh john@host01
john@host01's password: password
$ pwd
/home/john
$ whoami
john
```

      - You can successfully log in as user `john`.

c. Use the `exit` command to log out as user `john`.

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

- The `ssh` connection closes.

3. From **dom0**, log in to **host01** as `root`.

a. Use the `ssh` command to log in to **host01**.

- Provide the password when prompted.

```
[dom0]# ssh host01
root@host01's password:
```

b. Run the `five.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
# ./five.x
```

c. Use the `exit` command to log out.

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

- The `ssh` connection closes.

4. From **dom0**, log in to **host01** as user `john`.

a. Use the `ssh` command to log in to **host01**.

- Password is `password`.

```
[dom0]# ssh john@host01
root@host01's password: password
Permission denied, please try again.
root@host01's password: CTRL-C
```

- Note you cannot log in as user `john`.
- When prompted a second time for a password, press Ctrl + C.

5. Diagnose and fix the problem of user `john` not being able to log in.

- Review the lesson titled "User and Group Administration."
  - Which configuration file contains usernames? Is the user `john` present in this file?
  - Which configuration file contains user passwords? Is the user `john` present in this file?
  - If entries are missing in these files, how do you re-create the entries?
  - If passwords are corrupted or forgotten, how do you re-create passwords?

6. Verify that you can successfully log in to **host01** as user `john` using `ssh` and then log out from **host01**.

7. Log in to **host01** as `root` using `ssh` in preparation for the next practice.
8. As `root` on **host01**, delete user `john`.

```
# userdel john
```

Solution 20-5: User Cannot Log In

Steps

1. Diagnose and fix the problem of user `john` not being able to log in.
   a. From **dom0**, log in to **host01** as the `root` user. Provide the password when prompted.

```
[dom0]# ssh host01
root@host01's password:
```

   b. Use the `grep` command to search for `john` in the `/etc/passwd` file.

```
# grep john /etc/passwd
john:x:1001:1001::/home/john:/bin/bash
```

   - Note the `john` entry is present.

   c. Use the `grep` command to search for `john` in the `/etc/shadow` file.

```
# grep john /etc/shadow
#john:$6$...:0:99999:7:::
```

   - Note the `john` entry is present but is commented out (`#` sign).

   d. Use the `vi` editor to modify `/etc/shadow`. Remove the `#` character from the beginning of the `john` line.

   - Because `/etc/shadow` is a read-only file, use `:wq!` to write, save, and exit `vi` after removing the `#` character.

```
# vi /etc/shadow
...
#john:$6$...:0:99999:7:::                (old entry)
john:$6$...:0:99999:7:::                 (new entry)
```

   - Alternatively, you could run the following command to re-create a password entry in the `/etc/shadow` file for user `john`.

```
# passwd john
Changing password for user john.
...
```

   e. Use the `exit` command to log out as user `root`.

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

   - The `ssh` connection closes.

2. Verify you can log in as user `john`.
   a. From **dom0**, use the `ssh` command to log in to **host01** as user `john`.

   - Password is `password`. Verify you can successfully log in as user `john`.

```
[dom0]# ssh john@host01
john@host01's password: password
$ pwd
```

```
/home/john
$ whoami
john
```

- Note that you can now log in as user `john`.

b. Use the `exit` command to log out as user `john`.

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

c. Use the `ssh` command to log in to **host01** as `root` in preparation for the next practice.

- Provide the password when prompted.

```
[dom0]# ssh host01
root@host01's password:
```

d. As `root` on **host01**, delete user `john`.

```
# userdel john
```

Practice 20-6: File System Troubleshooting

Overview

In this practice, you diagnose and fix an unmountable file system.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Create a partition on `/dev/xvdb`.
   a. Use the `fdisk -l` command to display the partition table on `/dev/xvdb`.

```
# fdisk -l /dev/xvdb

Disk /dev/xvdb: 5368 MB, 5368709120 bytes, 10485760 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

   - Note that there are no partitions on `/dev/xvdb`.

   b. Use the `fdisk` command to partition `/dev/xvdb`.

```
# fdisk /dev/xvdb
...
Command (m for help):
```

   c. Add a new **primary** partition and assign it the number **1**.

```
Command (m for help): n
Partition type:
    p    primary partition (0 primary, 0 extended, 4 free)
    e    extended
Select (default p): ENTER
Using default response p
Partition number (1-4, default 1): ENTER
```

   d. Continue adding the new partition, using the entire disk as follows:

```
First sector (2048-10485759, default 2048): ENTER
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default
10485759): ENTER
Using default value 10485759
Partition 1 of type Linux and of size 5 GiB is set
```

   e. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
```

```
Calling ioctl() to re-read partition table.
Syncing disks.
```

2. Make a file system on `/dev/xvdb1`.
   a. Use the `mkfs` command to make an **ext4** file system on `/dev/xvdb1`.

```
# mkfs -t ext4 /dev/xvdb1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
...
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

3. Mount the file system.
   a. Use the `mkdir` command to create `/Test`.

```
# mkdir /Test
```

   b. Use the `mount` command to mount `/dev/xvdb1` on `/Test`.

```
# mount /dev/xvdb1 /Test
```

4. Copy files to `/Test`.
   a. Use the `cp` command to copy `/boot/init*` to `/Test`. Use the `ls` command to display the contents of the `/Test` directory.

```
# cp /boot/init* /Test
# ls /Test
initramfs-0-rescue-...img
initramfs-3.10.0-...el7.x86_64.img
initramfs-4.1.12-...el7uek.x86_64.img
...
```

5. Unmount the file system on `/dev/xvdb1`.
   a. Use the `umount` command to unmount `/Test`.

```
# umount /Test
```

6. Execute the `six.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
# ./six.x
```

7. Attempt to mount the file system on `/dev/xvdb1` on `/Test`.
   a. Use the same `mount` command used previously.

```
# mount /dev/xvdb1 /Test
mount: /dev/xvdb1 is write-protected, mounting read-only
```

```
mount: unknown filesystem type '(null)'
```

- Note that the `mount` command fails.

8. Diagnose and fix the problem with the file system.
   - Review the lesson titled “Partitions, File Systems, and Swap.”
     - How do you specify the file system type with the `mount` command?
     - What commands are available to check and repair file systems?
9. Ensure you can mount the file system on `/Test`.
   - Ensure the files (`initramfs*`) exist on `/Test`.

Note: The following is a clean-up task. Perform this task only after the File System problem is fixed.

10. After ensuring the problem is fixed, restore **host01** to its beginning state.
    a. Use the `umount` command to unmount `/Test`.
       - This is necessary before the associated partition can be deleted. Ensure you are in the `root` user's home directory first.

```
# cd
# umount /Test
```

    b. Use the `fdisk` command to delete the `/dev/xvdb1` partition.
       - Use the `d` command to delete partition 1.
       - Use the `p` command to print the partition table and confirm there are no partitions.
       - Use the `w` command to save the partition table and exit the `fdisk` utility.

```
# fdisk /dev/xvdb
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): p

Disk /dev/xvdb: 5368 MB, 5368709120 bytes ...
...
    Device Boot      Start         End      Blocks   Id  System

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

    c. View the devices in the `/proc/partitions` file to ensure that the `xvdb1` entry no longer exists.

```
# cat /proc/partitions
major minor  #blocks  name

 202        0   27262976 xvda
 202        1    1048576 xvda1
 202        2   20971520 xvda2
 202        3    2097152 xvda3
 202        4          1 xvda4
 202        5    3143680 xvda5
 202       16    5242880 xvdb
 202       48    5242880 xvdd
  11        0    4336640 sr0
```

d. Use the `rmdir` command to remove the `/Test` mount point.

```
# rmdir /Test
```

Solution 20-6: File System Troubleshooting

Steps

1. Diagnose the problem.
   a. Run the `mount` command but include the `-t ext4` option to specify the file system type.

```
# mount -t ext4 /dev/xvdb1 /Test
mount: wrong fs type, bad option, bad superblock on /dev/xvdb1,
       missing codepage or helper program, or other error

       In some cases useful info is found in syslog - try
       dmesg | tail or so
```

   b. Per the preceding suggestion, run the `dmesg | tail` command.

```
# dmesg | tail
...
[ ...] EXT4-fs (xvdb1): VFS: Can't find ext4 filesystem
```

   - Note the VFS error: "Can't find ext4 filesystem" on `xvdb1`.

   c. Use the `tail` command to view the last lines in the `/var/log/messages` file.

```
# tail /var/log/messages
...
<date_time> host01 kernel: EXT4-fs (xvdb1): VFS: Can't find ext4
filesystem
...
```

   - Note the error message written to the `/var/log/messages` file is similar to the error message given by the `dmesg` command.

   d. Use the `tune2fs -l /dev/xvdb1` command to list the contents of the file system superblock.

```
# tune2fs -l /dev/xvdb1
tune2fs 1.42.9 (28-Dec-2013)
tune2fs: Bad magic number in super-block while trying to open
/dev/xvdb1
Couldn't find valid filesystem superblock
```

   - Note the "superblock" error.

   e. Use the `dumpe2fs /dev/xvdb1` command to dump file system information.

```
# dumpe2fs /dev/xvdb1
dumpe2fs 1.42.9 (28-Dec-2013)
dumpe2fs: Bad magic number in super-block while trying to open
/dev/xvdb1
Couldn't find valid filesystem superblock
```

   - Note this command reports the same "superblock" error on `/dev/xvdb1`.

2. Fix the file system.
   a. Use the `fsck /dev/xvdb1` command to check (and repair) the file system.
      - Press **Enter** to accept the default **`y`** (yes) answer.

```
# fsck /dev/xvdb1
fsck from util-linux 2.23.2
e2fsck 1.42.9 (28-Dec-2013)
ext2fs_open2: Bad magic number in super-block
fsck.ext2: Superblock invalid, trying backup blocks...
/dev/xvdb1 was not cleanly unmounted, check forced.
Resize inode not valid. Recreate<y>? ENTER
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
Free blocks count wrong for group #0 ...
Fix<y>? ENTER
Free blocks count wrong for group #1 ...
Fix<y>? ENTER
Free blocks count wrong ...
Fix<y>? ENTER
Free inodes count wrong for group #0 ...
Fix<y>? ENTER
Free inodes count wrong ...
Fix<y>? ENTER

/dev/xvdb1: *****    FILE SYSTEM WAS MODIFIED *****
/dev/xvdb1: ...
```

   b. Run the `fsck /dev/xvdb1` command a second time to check the file system.

```
# fsck /dev/xvdb1
fsck from util-linux 2.23.2
e2fsck 1.42.9 (28-Dec-2013)
/dev/xvdb1: clean ...
```

      - Note that the file system is fixed (clean).
3. Mount the file system.
   a. Use the `mount` command to mount `/dev/xvdb1` on `/Test`.

```
# mount /dev/xvdb1 /Test
```

      - The `mount` command is successful. The `fsck` command fixed the corrupt superblock.

b. Use the `ls` command to list the contents of the `/Test` directory.

```
# ls /Test
initramfs-0-rescue-...img
initramfs-3.10.0...el7.x86_64.img
initramfs-4.1.12-...el7uek.x86_64.img
...
```

- Note that all the original files are present in the directory.

4. After ensuring the problem is fixed, restore **host01** to its beginning state.

a. Use the `umount` command to unmount `/Test`.

- This is necessary before the associated partition can be deleted. Ensure you are in the `root` user's home directory first.

```
# cd
# umount /Test
```

b. Use the `fdisk` command to delete the `/dev/xvdb1` partition.

- Use the `d` command to delete partition 1.
- Use the `p` command to print the partition table and confirm there are no partitions.
- Use the `w` command to save the partition table and exit the `fdisk` utility.

```
# fdisk /dev/xvdb
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted

Command (m for help): p

Disk /dev/xvdb: 5368 MB, 5368709120 bytes ...
...
   Device Boot      Start         End      Blocks   Id  System

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

c. View the devices in the `/proc/partitions` file to ensure that the `xvdb1` entry no longer exists.

```
# cat /proc/partitions
major minor  #blocks  name


 202        0   27262976 xvda
```

```
 202        1    1048576 xvda1
 202        2   20971520 xvda2
 202        3    2097152 xvda3
 202        4          1 xvda4
 202        5    3143680 xvda5
 202       16    5242880 xvdb
 202       48    5242880 xvdd
  11        0    4336640 sr0
```

d. Use the `rmdir` command to remove the `/Test` mount point.

```
# rmdir /Test
```

Practice 20-7: Network Connectivity Problem

Overview

In this practice, you diagnose and fix a network connectivity problem.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. From **dom0**, connect to the **host01** VM using VNC.
   a. Run the `dmesg` command with the "`C`" option to clear the kernel ring buffer so that no messages are printed to the screen when connecting with `vncviewer`.

```
# dmesg -C
```

   b. Use the `exit` command to close the `ssh` connection to **host01**.

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

   c. From **dom0**, connect to **host01** by using the `xm vncviewer host01&` command.

```
# xm vncviewer host01&
```

   d. Log in as the `root` user. Press **Enter** to align the cursor with the login prompt as needed.
      - Enter the password when prompted.

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

2. Verify the network interface is configured properly.
   a. Use the `ping` command to contact **dom0** and **host03**. Press Ctrl + C to abort the commands after connectivity is confirmed.

```
# ping dom0
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
CTRL-C
# ping host03
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
CTRL-C
```

3. Execute the `seven.x` program from the `root` user's home directory.

```
# cd
# pwd
```

```
/root
# ./seven.x
```

4. Repeat the task to test the network interface configuration.
   a. Use the `ping` command to contact **dom0**.
      - Press Ctrl + C to abort the command.

```
# ping dom0
PING example.com (192.0.2.1) 56(84) bytes of data.
CTRL-C
```

   b. Use the `ping` command to contact **host03**.
      - Press Ctrl + C to abort the command.

```
# ping host03
PING example.com (192.0.2.103) 56(84) bytes of data.
CTRL-C
```

      - The remote systems are no longer reachable.
5. Diagnose and fix the network connectivity problem.
   - Review the lesson titled “Network Configuration”.
     - How do you display the configuration of your network?
     - Are the network interfaces up?
     - What are the IP addresses of **dom0** and **host03**?
     - Are **dom0** and **host03** on the same network as **host01**?
     - Do you have a route to **dom0** and **host03**?
     - How do you view the route table?
     - In which files do you configure network interfaces?
     - Are these network interface configuration files configured properly?
     - Is the network service running?
6. Verify network connectivity to the remote hosts is working.
   a. Run the `ping dom0` command.
   b. Run the `ping host03` command.

Solution 20-7: Network Connectivity Problem

Steps

1. Diagnose the network connectivity problem.
   a. Use the `ip addr` command to display the configuration of your network interfaces.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state ...
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 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 192.0.3.101/24 brd 192.0.3.255 scope global ...
...
    inet6 ...
...
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 ...
...
    inet6 ...
...
```

   - The output shows that the `eth0` interface is `UP` and has an IP address of `192.0.3.101`.

   b. Use the `route` command to display the route table.

```
# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags ...  Iface
default         example.com     0.0.0.0         UG    ...  eth0
example.com     0.0.0.0         255.255.255.255 UH    ...  eth0
192.0.3.0       0.0.0.0         255.255.255.0   U     ...  eth0
192.168.1.0     0.0.0.0         255.255.255.0   U     ...  eth1
192.168.122.0   0.0.0.0         255.255.255.0   U     ...  virbr0
```

   - The route table indicates the route to the `192.0.3.0` network is through `eth0`.

   c. Use the `cat` command to view the contents of the `/etc/hosts` file.

```
# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain ...
192.0.2.1      example.com                dom0
192.0.2.101    host01.example.com         host01
192.0.2.102    host02.example.com         host02
```

```
192.0.2.103    host03.example.com          host03
```

- The contents of the `/etc/hosts` file indicate the IP address should be configured with `192.0.2` network addresses and not `192.0.3`.

2. Fix the network connectivity problem.

   a. Use the `vi` editor to modify `/etc/sysconfig/network-scripts/ifcfg-eth0` and change the `IPADDR` directive as shown.

```
# vi /etc/sysconfig/network-scripts/ifcfg-eth0
...
IPADDR=192.0.3.101                          (old value)
IPADDR=192.0.2.101                          (new value)
...
```

   b. Use the `systemctl` command to restart the network service.

   - Always restart the service whenever a configuration file is changed.
   - Because you are connected from **dom0** to **host01** using VNC, you will not lose your connection when restarting the network.

```
# systemctl restart network
```

3. Verify network connectivity to the remote hosts is working.

   a. Use the `ping` command to contact **dom0** and **host03**. Use Ctrl + C to abort the commands after connectivity is confirmed.

```
# ping dom0
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
CTRL-C
# ping host03
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
CTRL-C
```

   - Network connectivity to the remote hosts works.

Practice 20-8: Remote Access Problem

Overview

In this practice, you diagnose and fix a remote access problem.

Assumptions

- You are the `root` user on the **host01** VM.
- This practice is performed on both **host01** and **host03**.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. Log in to **host03**.
   a. Open a second terminal window on **dom0** by double-clicking the Terminal icon on the **dom0** desktop.

      Stay connected to **host01** from another terminal window.

   b. Become the `root` user on **dom0** from the second terminal window. Provide the password when prompted.

```
[dom0]$ su -
Password:
[dom0]# whoami
root
```

   c. From the second terminal window, connect to the **host03** guest by using the `xm vncviewer host03&` command.

```
[dom0]# xm vncviewer host03&
```

      - The GNOME login window appears.

   d. Select Oracle Student from the GNOME login window; enter the password.
   e. Right-click on the GNOME desktop and select **Open Terminal** from the pop-up menu.
   f. In the terminal window, become the `root` user by entering the `su -` command and provide the `root` password.

```
[host03]$ su -
Password:
[host03]# whoami
root
```

2. Ensure remote connectivity is working from **host03** to **host01**.
   a. As the `root` user on **host03**, confirm that you can use the `ssh` command to connect to **host01**.
      - Answer **`yes`** to continue connecting, when prompted.
      - Provide the `root` password when prompted.

- Use the `hostname` command to confirm that you did connect.

```
[host03]# ssh host01
The authenticity of host 'host01 (192.0.2.101)' can't be
established.
ECDSA key fingerprint is ...
ECDSA key fingerprint is ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host01,192.0.2.101' (ECDSA) to the
list of known hosts.
root@host01's password:
Last login: ...
[host01]# hostname
host01.example.com
```

b. Use the `logout` command to log off **host01**.

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

3. Ensure remote connectivity is working from **host01** to **host03**.

a. As the `root` user on **host01**, confirm that you can use the `ssh` command to connect to **host03**.

- Provide the `root` password when prompted.
- Use the `hostname` command to confirm that you did connect.

```
[host01]# ssh host03
root@host03's password:
Last login: ...
[host03]# hostname
host03.example.com
```

b. Use the `logout` command to log off **host03**.

```
[host03]# logout
Connection to host03 closed.
```

4. From **host01**, execute the `eight.x` program from the `root` user's home directory.

```
[host01]# cd
[host01]# pwd
/root
[host01]# ./eight.x
```

5. Ensure remote connectivity is working from **host01** to **host03**.

a. As the `root` user on **host01**, confirm that you can use the `ssh` command to connect to **host03**.

```
[host01]# ssh host03
root@host03's password:
Last login: ...
```

```
[host03]# hostname
host03.example.com
```

b. Use the `logout` command to log off **host03**.

```
[host03]# logout
Connection to host03 closed.
```

6. Ensure remote connectivity is working from **host03** to **host01**.

a. As the `root` user on **host03**, confirm that you can use the `ssh` command to connect to **host01**.

```
[host03]# ssh host01
ssh: connect to host host01 port 22: No route to host
```

- Note the `ssh` command fails.

7. Diagnose and fix the remote connectivity problem.

- Think about what you can and what you cannot do:
  - You can `ssh` from **host01** to **host03**
  - You cannot ssh from **host03** to **host01**
- Can you `ping` **host01** from **host03**? If so, then there is nothing wrong with the network interface configuration.
- Review the lesson titled “OpenSSH.”
  - Ensure the `sshd` service is configured properly and that the service is running.
- Note the error message, “ssh: connect to host host01 port 22: No route to host.”
  - View the `/var/log/messages` log file for entries related to `ssh` or port `22`.
  - View the `/var/log/secure` log file for entries related to `ssh` or port `22`.
- Review the lesson titled “Security Administration.”
  - Is the firewall prohibiting a connection on port `22`?
  - Is there a TCP wrapper configured that is causing the problem?

8. After fixing the problem, ensure you can `ssh` from **host03** to **host01**.

Solution 20-8: Remote Access Problem

Steps

1. Attempt to diagnose the problem by testing network connectivity.
   a. From **host03**, use the `ping host01` command to test network connectivity to **host01**.

```
[host03]# ping host01
64 bytes from example.com (192.0.2.101): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.101): icmp_seq=1 ttl=64 ...
64 ytes from example.com (192.0.2.101): icmp_seq=1 ttl=64 ...
CTRL-C
```

   - Note the `ping` command verifies that network connectivity to **host01** exists.

2. Attempt to diagnose the problem by checking the `sshd` configuration.
   a. From **host01**, use the `systemctl` command to view the status of the `sshd` service.

```
[host01]# systemctl status sshd
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service...
   Active: active (running) since ...
...
```

   - The `sshd` service is running.

   b. From **host03**, use the `systemctl` command to view the status of the `sshd` service.

```
[host03]# systemctl status sshd
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service...
   Active: active (running) since ...
...
```

   - The `sshd` service is running.

   c. From **host01**, use the `grep` command to search for "`ssh`" in the `/var/log/messages*` files.
   - Repeat the command but search for `port 22`.

```
[host01]# grep ssh /var/log/messages*
...
[host01]# grep "port 22" /var/log/messages*
```

   - Note that neither command returned output that indicates the cause of the problem.

   d. From **host01**, use the `grep` command to search for "`ssh`" in the `/var/log/secure` file.
   - Repeat the command but search for "`port 22`".
   - Enter the `date` command and note the date and time.

```
[host01]# grep ssh /var/log/secure
...
```

```
[host01]# grep "port 22" /var/log/secure
...
[host01]# date
<date time>
```

- Note that the <*date_time*> stamp on the log file entries does not correspond to the <*date_time*> of the problem.
- Note that there are several "pam" messages related to the sshd service.
  - Pluggable Authentication Modules (PAM) is covered in another course.
  - PAM configuration could cause this problem in the real world, but because PAM has not been covered in this course, you can conclude PAM is not the cause of the problem.

3. Attempt to diagnose the problem by checking the firewall (firewalld and iptables) configuration.
   a. From **host01**, use the systemctl command to view the status of the iptables service.

```
[host01]# systemctl status iptables
Unit iptables.service could not be found.
```

   - Note that the iptables service is not running and not even installed on **host01**.

   b. From **host01**, use the systemctl command to view the status of the firewalld service.

```
[host01]# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service...
   Active: active (running) since ...
...
```

   - Note that the firewalld service is running.

   c. From **host01**, use the firewall-cmd command to list the trusted services.

```
[host01]# firewall-cmd --list-services
dhcpv6-client
```

   - Note that the dhcpv6-client service is trusted.
   - Note that the ssh service is not trusted.
     - This is the cause of the problem.
   - To fix the problem, either stop the firewalld service or trust the ssh service.

4. Fix the remote connectivity problem by trusting the ssh service when firewalld is running.
   a. From **host01**, use the firewall-cmd command to trust the ssh service.

```
[host01]# firewall-cmd --add-service=ssh
success
```

   b. From **host01**, use the firewall-cmd command to list the trusted services.

```
[host01]# firewall-cmd --list-services
dhcpv6-client ssh
```

- Note that the `ssh` service is now trusted.

5. Ensure remote connectivity is working.

   a. From **host03**, confirm that you can use the `ssh` command to connect to **host01**.

   - Provide the `root` password when prompted.
   - Use the `hostname` command to confirm that you did connect.

```
[host03]# ssh host01
root@host01's password:
Last login: ...
[host01]# hostname
host01.example.com
```

   b. Use the `logout` command to log off **host01**.

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

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

Practice 20-9: Log File Is Not Getting Updated

Overview

In this practice, you diagnose and fix a problem with a log file not getting updated.

Assumptions

- You are the `root` user on both **host01** and **host03**.
- This practice is performed on both **host01** and **host03**.
- The prompts include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. From **host01**, change the `oracle` user's password to `oracle`.
   a. Use the `passwd oracle` command to set the password.
      - Ignore the "BAD PASSWORD" message.

```
[host01]# passwd oracle
Changing password for user oracle.
New password: oracle
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: oracle
passwd: all authentication tokens updated successfully.
```

2. From **host03**, confirm you can use the `ssh` command to log in to **host01** as the `oracle` user.
   a. Use the `exit` command to log out as the `root` user.
      - Use the `whoami` command to confirm you are the `oracle` user.

```
[host03]# exit
logout
[host03]$ whoami
oracle
```

   b. Use the `ssh` command to log in to **host01**.
      - Use the `hostname` command to verify you were able to log in. Provide the password when prompted.

```
[host03]$ ssh host01
oracle@host01's password:
Last login: ...
[host01]$ hostname
host01.example.com
```

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

```
[host01]$ exit
```

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

3. From **host01**, save a copy of the `/var/log/secure` file.
   a. Use the cp command to copy `/var/log/secure` to `~/secure_before`.

```
[host01]# cp /var/log/secure ~/secure_before
```

   - You review the differences in these copies later in this practice.

4. From **host01**, configure the `pam_nologin` authentication module to prevent non-`root` login.
   - PAM is covered in another course.
     - For the purposes of this practice, PAM checks for the existence of the `/etc/nologin` file, and if the file exists, remote logins by non-`root` users are denied and the content of this file is displayed as an error message.

   Use the `vi` editor and create the `/etc/nologin` file with the following contents:

```
[host01]# vi /etc/nologin
No logins allowed at this time.
```

5. As the `oracle` user on **host03**, attempt to log in to **host01**.
   a. Use the `ssh` command to connect to **host01**. Provide the password when prompted.

```
[host03]$ ssh host01
oracle@host01's password:
No logins allowed at this time.

Authentication failed.
```

   - Note that the connection is denied by the PAM authentication module.

6. From **host01**, view the new entries in the `/var/log/secure` log file.
   a. Use the `diff` command to view the differences in the `/var/log/secure` log file and the copy of the log file you made previously.

```
[host01]# diff /var/log/secure ~/secure_before
< <date_time> host01 [sshd[...]: Failed password for oracle from
192.0.2.103 port ... ssh2
< <date_time> host01 [sshd[...]: fatal: Access denied for user
oracle by PAM account configuration [preauth]
```

   - Note that two entries were written to this file when the `oracle` user attempted to `ssh` from **host03** to **host01**.
   - Also note that one entry specifically references PAM account configuration.

7. From **host01**, execute the `nine.x` program from the `root` user's home directory.

```
[host01]# cd
[host01]# pwd
/root
[host01]# ./nine.x
```

8. From **host01**, save a copy of the `/var/log/secure` file.
   a. Use the `cp` command to copy `/var/log/secure` to `~/secure_before`.
      - Answer **`y`** to overwrite the file.

```
[host01]# cp /var/log/secure ~/secure_before
cp: overwrite '/root/secure_before'? y
```

      - You review the differences in these copies later in this practice.
9. As the `oracle` user on **host03**, attempt to log in to **host01**.
   a. Use the `ssh` command to connect to **host01**. Provide the password when prompted.

```
[host03]$ ssh host01
oracle@host01's password:
No logins allowed at this time.

Authentication failed.
```

      - Note that the connection is denied by the PAM authentication module.
10. From **host01**, view the new entries in the `/var/log/secure` log file.
    a. Use the `diff` command to view the differences in the `/var/log/secure` log file and the copy of the log file you made previously.

```
[host01]# diff /var/log/secure ~/secure_before
```

      - Note that there are no differences in the files.
      - The `/var/log/secure` log file is not getting updated as expected.
11. Diagnose and fix the problem of the log file not getting updated.
      - Review the lesson titled "System Logging."
        - Is the logging daemon running?
        - Is logging configured for the `/var/log/secure` log file?
12. After fixing the problem, ensure the `/var/log/secure` log file is getting updated when you attempt to `ssh` from **host03** to **host01** as the `oracle` user.
13. Return **host01** to its original state.
    a. Use the `rm` command to remove the `/etc/nologin` file.

```
# rm /etc/nologin
rm: remove regular file '/etc/nologin'? y
```

    b. Use the `rm` command to remove the `~/secure_before` file.

```
# rm ~/secure_before
rm: remove regular file '/root/secure_before'? y
```

Solution 20-9: Log File Is Not Getting Updated

Steps

1. Diagnose the cause of the logging problem.
   a. As the `root` user on **host01**, view the status of the `rsyslog` service.

```
[host01]# systemctl status rsyslog
rsyslog.service – System Logging Service
   Loaded: loaded (/usr/lib/systemd/system/rsyslog.service...
   Active: active (running) since ...
...
```

   - The `rsyslogd` service is running.

   b. Use the `grep` command to search for `/var/log/secure` in the `rsyslog` configuration file.

```
[host01]# grep /var/log/secure /etc/rsyslog.conf
#authpriv.*                              /var/log/secure
```

   - Note the configuration file contains an entry for `/var/log/secure`; however, the entry is commented out (preceded by a `#` sign).

2. Fix the problem.
   a. Use the `vi` editor to remove the `#` from the beginning of the line containing `/var/log/secure`.

```
[host01]# vi /etc/rsyslog.conf
...
#authpriv.*            /var/log/secure            (old entry)
authpriv.*             /var/log/secure            (new entry)
...
```

   b. Use the `systemctl` command to restart the `rsyslog` service.

```
[host01]# systemctl restart rsyslog
```

3. Verify the `/var/log/secure` log file is getting updated.
   a. From **host01**, use the `cp` command to copy `/var/log/secure` to `~/secure_before`.

   - Answer **`y`** to overwrite the file.

```
[host01]# cp /var/log/secure ~/secure_before
cp: overwrite '/root/secure_before'? y
```

   b. As the `oracle` user on **host03**, use the `ssh` command to connect to **host01**. Provide the password when prompted.

```
[host03]$ ssh host01
oracle@host01's password:
No logins allowed at this time.
```

```
Authentication failed.
```

- Note that the connection is denied by the PAM authentication module.

c. From **host01**, use the `diff` command to view the differences in the `/var/log/secure` log file and the copy, `~/secure_before`.

```
[host01]# diff /var/log/secure ~/secure_before
< <date_time> host01 [sshd[...]: Failed password for oracle from
192.0.2.103 port ... ssh2
< <date_time> host01 [sshd[...]: fatal: Access denied for user
oracle by PAM account configuration [preauth]
```

- Note that entries are now being written to the `/var/log/secure` file as expected.

4. Return **host01** to its original state.

a. Use the `rm` command to remove the `/etc/nologin` file.

```
# rm /etc/nologin
rm: remove regular file '/etc/nologin'? y
```

b. Use the `rm` command to remove the `~/secure_before` file.

```
# rm ~/secure_before
rm: remove regular file '/root/secure_before'? y
```

Appendix A: Source Code for Problem-Causing Executables

Overview

In this appendix, the source code for the executables that cause the problems is given.

Practice 20-2: System Boots into Single-User Mode

The "`two.x`" script appends the word `single` to the end of the kernel line for the default kernel after backing up the grub.cfg file:

```
#!/bin/bash
cp /boot/grub2/grub.cfg /boot/grub2/grub.cfg.save
sed –i –e '/vmlinuz-4.1.12/s/$/ single/' /boot/grub2/grub.cfg
```

The result of this program is as follows:

```
# cat /boot/grub2/grub.cfg
...
linux16 /vmlinuz-4.1.12-112.16.4... single...
...
```

Practice 20-3: Status Commands Fail

The "`three.x`" script runs the following `umount` command to unmount the `proc` file system:

```
#!/bin/bash
umount –l /proc
```

Practice 20-4: cron Job Fails to Run

The "`four.x`" script stops the `crond` service by running the following command:

```
#!/bin/bash
systemctl stop crond > /dev/null
```

Practice 20-5: User Cannot Log In

The "`five.x`" script inserts the "`#`" sign at the beginning of the `john` line in the `/etc/shadow` file after backing up the original /etc/shadow file:

```
#!/bin/bash
cp /etc/shadow /etc/shadow.save
sed –i –e '/john/s/^/#/' /etc/shadow
```

The result of this program is as follows:

```
# cat /etc/shadow
...
#john:$6$...:0:99999:7:::
```

Practice 20-6: File System Troubleshooting

The "six.x" script runs the following dd command to corrupt the file system superblock.

```
#!/bin/bash
dd if=/dev/zero of=/dev/xvdb1 bs=1024 skip=1000 count=300 2>
/dev/null
```

Practice 20-7: Network Connectivity Problem

The "seven.x" script changes the IP address of eth0 on **host01** from 192.0.2.101 to 192.0.3.101 by running the following commands after backing up the original ifcfg-eth0 file:

```
#!/bin/bash
cd /etc/sysconfig/network-scripts
/bin/cp ifcfg-eth0 ~/ifcfg-eth0.save
sed –e 's/2.101/3.101/' ifcfg-eth0 > ifcfg-eth0.new
/bin/mv ifcfg-eth0.new ifcfg-eth0
cd
systemctl restart network > /dev/null
```

The result of this program follows:

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
...
IPADDR=192.0.3.101
...
# ip addr
...
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 192.0.3.101/24 brd 192.0.3.255 scope global eth0
......
```

Practice 20-8: Remote Access Problem

The "eight.x" script removes the firewalld rule on **host01**, which trusts the sshd service, by running the following command:

```
#!/bin/bash
firewall-cmd --remove-service=ssh > /dev/null
```

Practice 20-9: Log File Is Not Getting Updated

The "nine.x" script inserts the "#" sign at the beginning of the secure line in the /etc/rsyslog.conf file and restarts the rsyslog service by running the following commands after backing up the original rsyslog.conf file:

```
#!/bin/bash
cp /etc/rsyslog.conf /etc/rsyslog.conf.save
```

```
sed –i –e '/secure/s/^/#/' /etc/rsyslog.conf
systemctl restart rsyslog > /dev/null
```

The result of this program is as follows:

```
# cat /etc/rsyslog.conf
...
#authpriv.*                                    /var/log/secure
```

Appendix: Remote Access Options

Appendix: Remote Access Options – Overview

Appendix Overview

This appendix describes the connection process to **dom0** using Oracle's recommended VNC viewer, **TigerVNC**. **TigerVNC** is available with various Linux distributions, including Oracle Linux and Red Hat Enterprise Linux.

Other VNC viewers/clients can be used and students must follow the connection process for these viewers. For example, viewers from the following products have been shown to connect, but performance and clarity specifics may vary:

- RealVNC
- TightVNC
- UltraVNC
- TurboVNC

Appendix: Using the TigerVNC Viewer to Connect to dom0

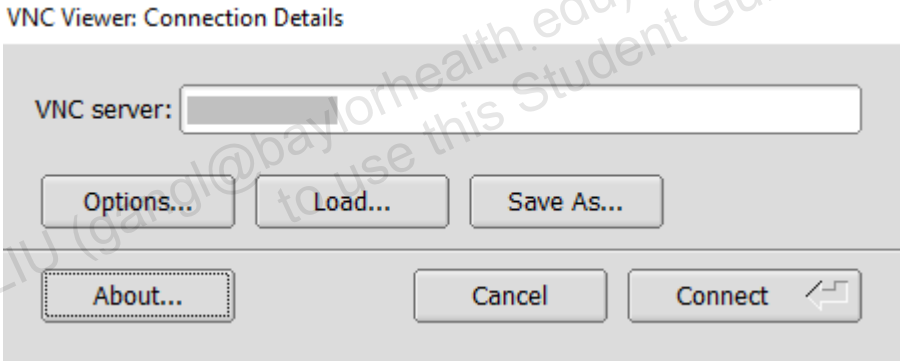
Overview

This appendix describes accessing your student PC (**dom0**) remotely using the **TigerVNC Viewer**.

Steps

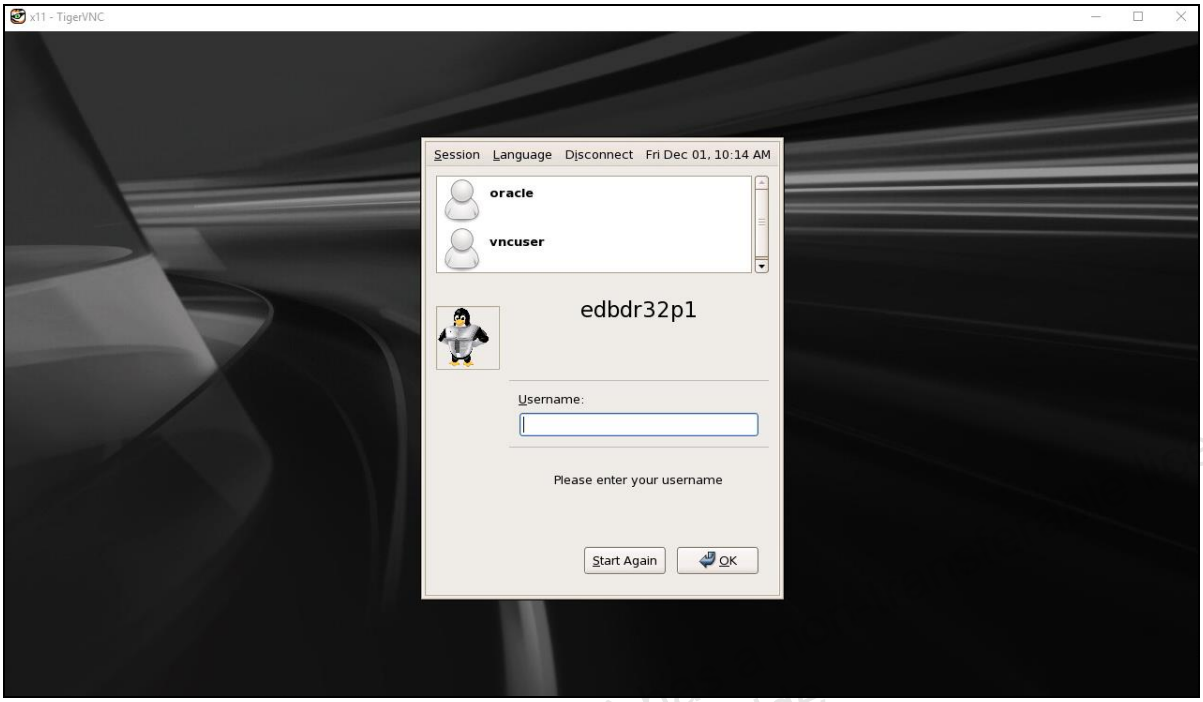
1. If you need to obtain the **TigerVNC Viewer**, the following site provides links to binaries: https://github.com/TigerVNC/tigervnc/releases.
2. Connect to the **dom0** VNC server.
   a. Run **vncviewer**. This is the **TigerVNC Viewer** icon:

   - The following window appears. A previously entered IP address for a VNC server is grayed out in this example.

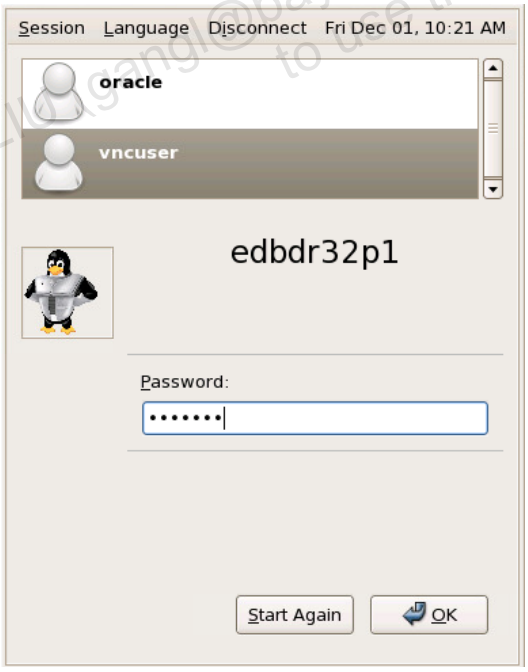


   b. Enter the IP address of **dom0** (provided to you) in the VNC server box. Click **Connect**.

- The following window appears. Your system name will be different than "edbrdr32p1," which is shown here.



3. Log on to **dom0**.
   a. Click **vncuser** and provide the password in the box as shown. See the “Security Credentials” section at the beginning of the Activity Guide for the lesson titled “Course Introduction” to obtain the password.



b. Click **OK**. The **dom0** GNOME virtual desktop appears:

- From here, you can open a terminal window and proceed with the practices as needed.