

File Systems and Security

In this activity, students will learn about the Linux file system and how disk file systems can be created and attached to different points within the Linux file system. This section also works through file management and security scenarios.

For each of the scenarios in this activity, you will capture the commands and the console output as explained in the scenarios. Each of the commands used in the following scenarios are described in the reading. For more information about a particular command, including available command line options, use the [man](#) command.

Videos

Tasks

[Scenario 1 - Scavenger Hunt Linux Style](#)

[Scenario 2 - Linux File System Lockdown](#)

[Scenario 3 - Putting the Squeeze on Files](#)

[Scenario 4 - Room to Grow](#)



Scenario 1 - Scavenger Hunt Linux Style

This scenario provides experience working with the Linux file system and viewing file and directory permissions. This scenario will use the following commands:

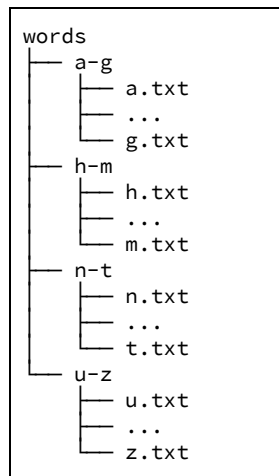
- Commands: [echo](#), [cd](#), [ls](#), [pwd](#), [which](#), [du](#)
1. Start a new konsole and set it to have unlimited scrollback by following directions here: [setting up konsole with unlimited scrollback](#).
 2. Use the [echo](#) command to display "Scenario 1 - Scavenger Hunt Linux Style" in the console
 3. Use the [pwd](#) command to determine your current location on the Linux file system
 4. Use the [which](#) command to identify where on the Linux file system the following commands are located:
 - [cp](#), [mv](#), [ls](#), [cd](#), [echo](#), [pwd](#), [touch](#), [cat](#), [which](#)
 5. It seems as though the most commonly used commands are located in `/usr/bin`. Use the [cd](#) command to change to the `/usr/bin` directory then use the [ls](#) command to see what else is there.
 6. Use the [du](#) command to see how much storage space is used by the files stored in `/usr/bin`.
 7. While in the `/usr/bin` directory, use the [rm](#) command to try and delete the [cp](#) command.
 - NOTE: Do not attempt this with `sudo`. :)
 8. While still in the `/usr/bin` directory, use the [ls -l](#) command to view the full details of each file in the directory. Pay special attention to the following:
 - What owner and group are assigned to each file?
 - What are the permissions assigned to Owner, Group and Other?
 - What does the "->" mean on some entries in the right hand column?
 - Do these files have the execute bit set? Why do you think that is?
 9. Use the [cd](#) command to change to the `/etc` directory then use the [ls -l](#) command to see what is there. Pay special attention to the following:
 - What owner and group are assigned to each file?
 - What are the permissions assigned to Owner, Group and Other?
 - Why do the directories have the execute bit (x) set while the files do not? What does this mean?
 10. Use the [ls -l](#) command specifically on the `/etc/passwd` file
 - What are the permissions?
 11. Use the [ls -l](#) command specifically on the `/etc/shadow` file
 - What are the permissions?
 - How do they differ from the `passwd` file?
 12. Save the session by clicking on [File](#) menu in konsole and then [Save Output As...](#) and save in a file named `module3-s1.txt` (make sure to remember where you saved the file!)
 13. Close the konsole by typing [exit](#) or by sending the End Of File (EOF) character: Press [CTRL-d](#)



Scenario 2 - Linux File System Lockdown

Knowing how to secure your personal workstation is important. This scenario provides practice working with the `chgrp` and `chmod` tools to become familiar with how they work using a sample dataset on your Linux VM.

- Commands: `echo`, `chgrp`, `chmod`, `ls`, `cd`, `wget`, `tar`, `groupadd`, `sudo`
1. Start a new konsole and set it to have unlimited scrollback by following directions here: [setting up konsole with unlimited scrollback](#).
 2. Use the `echo` command to display "Scenario 2 - Linux File System Lockdown" in the console
 3. Use the `mkdir` command to create a folder named `tmp` in your home directory and use the `cd` command to change to that folder.
 4. Use the following `wget` command to download the sample dataset to your `tmp` directory
 - `wget https://github.com/BoiseState/CS-HU153-resources/raw/master/activities/cs-hu153-words-dataset.tgz`
 5. Use the `tar` command to extract the `cs-hu153-words-dataset.tgz` dataset. Check the result with the command `ls -R words` (or use `tree words`)
 - `tar xvfz cs-hu153-words-dataset.tgz`



6. Use the `groupadd` command to create a new local group called `developers`
 - NOTE: This operation will require administrative (`sudo`) privileges
7. Use the `chgrp` command to change the group for the entire `words` directory, and all files and directories within it, to `developers`
 - NOTE: This operation will require administrative (`sudo`) privileges
8. Use the `chmod` command to modify the group privileges for the `words` directory structure to the following:
 - The `developers` group should have read and traverse (or execute) privileges to all directories in the words dataset: `words`, `a-g`, `h-m`, `n-t`, `u-z`
 - The `developers` group should only have read privileges to all text files in the words dataset: `*.txt`
9. Use the `ls -lR` command to recursively list the contents of the words directory structure to verify the permissions have been set correctly. The output will be similar to what is shown [here](#).
10. Save the session by clicking on `File` menu in konsole and then `Save Output As...` and save in a file named `module3-s2.txt` (make sure to remember where you saved the file!)
11. Close the konsole by typing `exit` or by sending the End Of File (EOF) character: Press `CTRL-d`



Scenario 3 - Putting the Squeeze on Files

For software developers, the days of pushing paper around are nearing their end, the 21st century equivalent is pushing bits around in the form of files. This scenario is all about packing up the bits to get them ready for transfer or archival. There are several different formats for packing up and compressing files. Common formats include: 7z, rar, bz, tar, tgz and zip. This scenario focuses specifically on working with the tgz and zip formats

Commands: [tar](#), [zip](#), [unzip](#), [wget](#), [scp](#), [ssh](#), [rm](#), [du](#)

1. Start a new konsole and set it to have unlimited scrollback by following directions here: [setting up konsole with unlimited scrollback](#).
2. Use the [echo](#) command to display "Scenario 3 - Putting the Squeeze on Files" in the console
3. Use the [mkdir](#) command to create a *tmp* directory (if it doesn't already exist) and cd to this directory
4. Use the [wget](#) command to download the CS121-resources repository
 - o [wget https://github.com/BoiseState/CS121-resources/archive/master.zip](#)
5. Use the [unzip](#) command to extract the contents from the cs121 archive
 - o [unzip master.zip](#)
6. Use the [cd](#) command to change to the examples directory
7. Use the [du -h](#) command to see how space is used by the files in the *chap07* directory
8. Use the [zip](#) command to archive (zip up) the *chap07* directory
9. Use the [tar](#) command to archive (tar up) the *chap07* directory
10. Use the [ls](#) command to compare the size of the *chap07.zip* and *chap07.tgz* files with the size of the uncompressed chap07 directory (which we found out in Step 7 above with the [du -h](#) command)
11. Use the [scp](#) command to copy the *chap07.tgz* file from your Linux VM to onyx
12. Use the [ssh](#) command to connect to onyx.boisestate.edu
13. Use the [tar](#) command to extract the *chap07.tgz* archive to a temporary directory under your account on onyx.
14. Use the [exit](#) command to leave onyx and return to your Linux VM
15. Use the [rm](#) command to remove the temporary directory you created for this scenario
16. Save the session by clicking on *File* menu in konsole and then *Save Output As...* and save in a file named [module3-s3.txt](#) (make sure to remember where you saved the file!)
17. Close the konsole by typing [exit](#) or by sending the End Of File (EOF) character: Press [CTRL-d](#)



Scenario 4 - Room to Grow

Developers often run into the situation where they run out of storage space. Common reasons a developer might run out of storage space include working with large datasets, generating lots of output data for debugging, or downloading the latest Lady Gaga album. Regardless of the reason, it is important to understand how to provision additional storage space. This scenario walks through how to attach a (virtual) hard disk to your Linux VM, partition and create a disk file system on it, and attach it to your Linux file system.

- Commands: `fdisk`, `mkfs`, `mount`, `df`, `lsblk`, `sudo`, `ls -l`, `chown`
1. Power off your Linux VM, open the VM's settings and add a new 8GB virtual hard disk to your VM. The Linux VM video in Activity 1 showed you how to add a new hard disk drive to your VM.
 2. Power on your Linux VM and open the Konsole
 3. Start a new konsole and set it to have unlimited scrollback by following directions here: [setting up konsole with unlimited scrollback](#).
 4. Use the `echo` command to display "Scenario 4 - Room to Grow" in the console
 5. Use the `df` command to view the available storage space. The `-h` option makes the output easier to understand.
 6. Use the `lsblk` command to view the block devices (storage devices) attached to the Linux VM.
 - What is the name of the new virtual hard disk? `sda`? `sdb`?
 - At this point, the new device should have no partitions (`sda1`, `sda2`, etc..) and it should not have a mount point. The size should be 8G (Gigabyte).
 7. Use the `fdisk` command to create a new partition on the new virtual hard disk. If the new device is named `sdb`, the following would be the appropriate `fdisk` command to get started.
`sudo fdisk /dev/sdb`
Then use the commands for `fdisk` to create a new partition and write it to the disk. Make sure to review Section 4.7.5 on `fdisk` in the class notes to do this step correctly.
 8. Use the `lsblk` command again to view the newly created partition.
 - NOTE: It should appear as `sdb1`
 9. Use the `mkfs` command to create a new `ext4` disk file system on the new partition
 - `sudo mkfs -t ext4 /dev/sdb1`
 10. Use the `mkdir` command to create a directory named *workspace* in your home directory. This will be the mount point for your new storage device
 11. Use the `mount` command to connect the new `ext4` partition to the workspace directory
 - `sudo mount -t ext4 /dev/sdb1 ~/workspace`
 12. Use the `lsblk` command to verify that the new storage is properly mounted
 13. Use the `df -h` command to verify that there is 8GB of space available in the workspace mountpoint
 14. Use the `ls -l` command to view the owner and group of the new Linux file system branch mounted in the workspace folder
 15. Use the `chown` command to change the owner and group of the workspace folder to your user account instead of root.
 16. Use the `umount` command to disconnect the storage from the workspace mount point



17. Edit the `/etc/fstab` file to automatically mount the new storage to your workspace folder at boot.
 - `sudo kwrite /etc/fstab`
(Make sure to add the correct file system type, that is `ext4`, for the new entry in `/etc/fstab`. Or your system may no longer reboot!)
18. Save the session by clicking on *File* menu in konsole and then *Save Output As...* and save in a file named `module3-s4.txt` (make sure to remember where you saved the file!)
19. Close the konsole by typing `exit` or by sending the End Of File (EOF) character: Press `CTRL-d`
20. Reboot the VM, then resume session in a new konsole.
21. Use the `df` and `lsblk` command to verify that the new storage is mounted
22. Save the session by clicking on *File* menu in konsole and then *Save Output As...* and save in a file named `module3-s4-extra.txt` (make sure to remember where you saved the file!)
23. Combine with the original session output file with the following command (The `>>` operator appends the first file to the the end of the second file):
`cat module3-s4-extra.txt >> module3-s4.txt`
24. Close the konsole by typing `exit` or by sending the End Of File (EOF) character: Press `CTRL-d`

