**Module: 1 - Linux server - Understand and use essential tools**

**1. Minimum Number of Partitions Needed to Install Linux**

* **At least one partition (/ root partition)** is required to install Linux. This partition contains the entire filesystem including system files, user files, and applications.
* **Optional partitions:**
  + **Swap partition:** Used for virtual memory (paging). Recommended if you have limited RAM.
  + **Separate /home partition:** Keeps user data separate from system files.
  + **Separate /boot partition:** Stores bootloader files (sometimes recommended in specific setups).

**Summary:**

* **Minimum:** 1 partition (/)
* **Recommended:** 2 partitions (/ and swap)

**2. Explain the chmod Command**

* chmod changes the **permissions** (access rights) of files or directories.
* Permissions control who can:
  + **Read (r):** View the contents
  + **Write (w):** Modify or delete
  + **Execute (x):** Run the file or access the directory
* Permissions are divided into three categories:
  + **Owner (user)**
  + **Group**
  + **Others**
* Permissions can be set in two ways:

1. **Symbolic mode:** Using letters to add/remove permissions
   * chmod u+x file → Add execute permission for user (owner)
   * chmod go-r file → Remove read permission for group and others
2. **Numeric (octal) mode:** Using numbers to set permissions
   * Each permission has a numeric value:
     + Read = 4
     + Write = 2
     + Execute = 1
   * Sum the values for each category:
     + chmod 755 file means:  
       User = 7 (4+2+1 = rwx)  
       Group = 5 (4+0+1 = r-x)  
       Others = 5 (r-x)

**3. How to Check Linux Memory Utilization**

* free -h: Shows total, used, free memory including swap in human-readable format (MB/GB).
* top or htop: Interactive tool showing real-time CPU and memory usage by processes.
* vmstat: Provides details about memory, CPU, I/O stats in a tabular form.
* View detailed memory info from the system file:
* cat /proc/meminfo
* Example output of free -h:
* total used free shared buff/cache available
* Mem: 7.7G 2.1G 3.5G 152M 2.1G 5.0G
* Swap: 2.0G 0B 2.0G

**4. Use grep to Search for Specific Patterns in Files**

* grep searches text files for lines matching a specified pattern.
* Basic syntax:
* grep "pattern" filename
* Useful options:
  + -i : Case-insensitive search.
  + -r : Recursive search through directories.
  + -n : Show line numbers where pattern occurs.
  + -v : Invert match (show lines not matching).
* Example: Find "error" in system logs.
* grep -i "error" /var/log/syslog

**5. Connecting to a Linux Server Using SSH**

* SSH (Secure Shell) allows you to remotely access a Linux machine securely.
* Syntax:
* ssh username@hostname\_or\_IP
* Example:
* ssh root@192.168.1.50
* If SSH runs on a custom port (not 22):
* ssh -p 2222 user@host
* On connection, you’ll be prompted for the user’s password (or use key-based authentication).

**6. Create 5 Files in /tmp and Bundle & Compress Using tar and gzip**

* Step 1: Create files in /tmp
* cd /tmp
* touch file1.txt file2.txt file3.txt file4.txt file5.txt
* Step 2: Archive the files using tar
* tar -cvf myfiles.tar file1.txt file2.txt file3.txt file4.txt file5.txt
  + c: create archive
  + v: verbose (list files)
  + f: specify file name
* Step 3: Compress the archive with gzip
* gzip myfiles.tar
* Final compressed archive: myfiles.tar.gz
* To extract later:
* tar -xvzf myfiles.tar.gz

**7. Describe the Root Account**

* The **root account** is the **superuser** with **unrestricted access** to all commands and files.
* Can perform any operation including:
  + Installing/removing software
  + Changing system configuration
  + Managing users
  + Accessing all files (even other users’ files)
* Root should be used **carefully** to avoid accidental damage to the system.
* Normal users use sudo to run commands with root privileges temporarily.

**8. What is a Shell?**

* A **shell** is a program that interprets user commands and interacts with the operating system.
* It acts as an interface between the user and the OS kernel.
* Shells provide command execution, scripting, and automation capabilities.
* Common shells:
  + **Bash** (Bourne Again Shell)
  + Zsh
  + Fish
  + Tcsh

**9. What is Linux?**

* Linux is a **free, open-source, Unix-like operating system kernel**.
* Combined with GNU utilities, it forms a complete OS known as GNU/Linux.
* It is widely used for servers, desktops, embedded systems, and smartphones.
* Known for stability, security, and multiuser multitasking.

**10. What is Bash?**

* **Bash** is the default command-line shell in most Linux distributions.
* It is a command interpreter and scripting language.
* Supports programming constructs like variables, loops, conditionals, and functions.
* Bash configuration files include .bashrc, .bash\_profile.

**11. First Step with a New Empty Hard Drive for Linux**

* **Partition the disk** to create usable sections using tools like:
  + fdisk (for MBR partitioning)
  + parted (supports GPT)
* Example:
* sudo fdisk /dev/sdb
* Create partitions, set types, and write changes.
* Then **format** partitions with a filesystem, e.g., ext4:
* mkfs.ext4 /dev/sdb1

**12. Command to Show Current Working Directory**

* Displays the absolute path of the current directory.

‘ Pwd ‘

**13. Command to Get Help With Various Options**

* Manual pages:
* man command\_name
* For example:
* man ls
* Alternatively:
* command\_name --help
* Example:
* ls --help

**14. Command to Display What All Users Are Currently Doing**

‘ W ’

* Shows who is logged in, their terminal, login time, idle time, and current processes.

**15. Command to Get Information About the Operating System**

* Kernel info:
* uname -a
* OS release info:
* cat /etc/os-release
* Example output includes distro name, version, and ID.

**16. Command to Create a Hard Link of a File**

ln source\_file hard\_link\_name

* Hard links point to the same inode.
* They behave like duplicates; deleting one does not affect the other.

**17. Command to Create a Soft Link (Symbolic Link) of a File and Directory**

* File:
* ln -s source\_file symlink\_name
* Directory:
* ln -s /path/to/directory symlink\_dir
* Soft links are like shortcuts, referencing the original path.

**18. Command to Search for Specific Pattern in a File**

grep "pattern" filename

* Finds and prints lines matching the pattern.

**19. Command to Show Basic Regular Expressions Using grep**

Examples:

* Lines starting with "root":
* grep "^root" /etc/passwd
* Lines ending with "bash":
* grep "bash$" /etc/passwd
* Lines containing numbers:
* grep "[0-9]" filename
* Lines containing "user" followed by one or more digits (extended regex):

grep -E "user[0-9]+" filename