**Linux | CentOS & Ubuntu**

****

**Overview**

Linux is an open-source and free to use, which has led to its widespread adoption in various computing environments, from personal computers to servers, embedded systems, and supercomputers.

Linux kernel originally created by Linus Torvalds in 1991.

Key Features of Linux

Open Source:

Linux is distributed under an open-source license, which means its source code is freely available to the public. This allows users to study, modify, and distribute the software.

Multiuser:

Linux supports multiple users simultaneously. Each user can have their own account with separate files, settings, and permissions.

Multitasking:

Linux is a multitasking operating system, capable of running multiple processes concurrently.

It efficiently manages system resources to ensure smooth operation.

Multiplatform:

Linux runs on various hardware architectures, including x86, ARM, PowerPC, and more. This versatility allows it to be used in a wide range of devices, from smartphones to mainframes.

Networking:

Linux has robust networking capabilities, making it an ideal choice for servers and networking devices. It supports protocols such as TCP/IP, UDP, HTTP, FTP, and more.

Security:

Linux offers strong security features, including user permissions, access control lists (ACLs), encryption, firewalls, and auditing tools. Its open-source nature allows security vulnerabilities

to be quickly identified and patched by the community.

Components of a Linux System

Kernel:

The Linux kernel is the core of the operating system. It manages hardware resources, provides system services, and facilitates communication between software and hardware components.

Shell:

The shell is a command-line interface that allows users to interact with the operating system.

It interprets user commands and executes them by interacting with the kernel.

Utilities:

Linux provides a wide range of command-line utilities for performing various tasks, such as file management, process management, networking, text processing, and more.

Graphical User Interface (GUI):

While Linux is often used via the command line, it also supports graphical desktop environments like GNOME, KDE, Xfce, and others. These environments provide a user-friendly interface with icons, windows, and menus.

Software Packages:

Linux distributions come with pre-installed software packages and package managers for installing, updating, and managing additional software. Common package managers include apt (Debian-based), yum/dnf (Red Hat-based), and pacman (Arch Linux).

Linux Distributions:

Linux distributions, often referred to as "distros," are variations of the Linux operating system tailored for specific use cases or preferences. Each distribution typically includes the Linux kernel,

a package manager, system utilities, and additional software packages.

Popular Linux distributions include:

* + Ubuntu
  + Debian
  + CentOS
  + Fedora
  + Arch Linux
  + Linux Mint
  + openSUSE
  + Slackware

These distributions may vary in terms of installation process, default software selection, package management system, and support model.

Overall, Linux offers flexibility, reliability, and customizability, making it a preferred choice for both individual users and organizations across the globe.

**Linux CentOS and Ubuntu Operational commands and description**

**File Management:**

|  |
| --- |
| **sudo -i**  #is used to start a shell with root privileges |
| **exit** # user privileges |
|  |
|  |
| **ls** #List directory contents. |
|  |
|  |
| **cd** #Change the current working directory. |
|  |
|  |

|  |
| --- |
| **pwd** #Print the current working directory. |
| **mkdir** # Create a new directory. |
| **rmdir** # Remove an empty directory. |
| **Rm** #Remove files or directories. |
| **cp** # Copy files or directories. |
| **mv** #Move or rename files or directories. |
| **touch**  #Create an empty file. |
| **cat** #Concatenate and display files. |
| **more/less** #View file contents one page at a time. |
| **head/tail** #Display the beginning/end of a file. |
| **grep**  #Search for patterns in files. |
| **chmod**  #Change file permissions. |
| **Chown** #Change file ownership. |
|  |

**System Management:**

|  |
| --- |
| **top**  #Display Linux processes. |
| **ps** #Display information about running processes. |
| **kill**  #Terminate processes. |
| **shutdown** #Shutdown or restart the system. |
| **reboot**  #Reboot the system. |
| **df**  #Report file system disk space usage. |
| **du**  #Estimate file space usage. |
| **free**  #Display amount of free and used memory. |
| **uname** #Print system information. |
| **Who** #Display who is logged on. |
| **history** #Display command history. |
| **Date**  #Print or set the system date and time. |
| **uptime**  #Show how long the system has been running. |

**Networking:**

|  |
| --- |
| **Ifconfig** #Configure network interface parameters. |
| **ping**  #Send ICMP echo requests to a host. |
| **Traceroute**  #Print the route packets trace to a network host. |
| **Netstat** #Print network connections, routing tables, interface statistics, etc. |
| **Ssh**  #Secure Shell - remote login protocol. |
| **Scp**  #Secure copy (remote file copy program). |
| **wget**  #Retrieve files from the web via HTTP, HTTPS, or FTP. |
| **Curl**  #Transfer data from or to a server. |

**Package Management:**

|  |
| --- |
| **apt/apt-get** #Package management commands for Debian/Ubuntu. |
| **Yum** #Package management command for CentOS/RHEL. |
| **Dnf** #Package management command for Fedora. |
| **Rpm**  #RPM Package Manager. |

**Text Processing:**

|  |
| --- |
| **Sed**  #Stream editor for filtering and transforming text. |
| **Awk** #Pattern scanning and processing language. |
| **Cut**  #Remove sections from each line of files. |
| **Sort** #Sort lines of text files. |
| **Uniq**  #Report or omit repeated lines. |
| **tr**  #Translate or delete characters. |
| **wc** #Print newline, word, and byte counts for each file. |
| **Diff**  #Compare files line by line. |

**Compression and Archiving:**

|  |
| --- |
| **Tar**  #Archive files. |
| **Gzip**  #Compress or expand files. |
| **bzip2** #Compress or decompress files. |
| **Zip**  #Package and compress (archive) files. |

**User Management:**

|  |
| --- |
| **Useradd**  #Create a new user or update default new user information. |
| **Userdel** #Delete a user account and related files. |
| **Usermod**  #Modify a user account. |
| **Passwd**  #Change user password. |

|  |
| --- |
| **Create Multiple users and provide the permission and add in wheel and adm group** |
|  |
| #adduser tassdaq |
| #passwd test123 |
|  |
| #id tassdaq |
|  |
| screenshot |
|  |
| #usermod -aG adm tassdaq |
| #usermod -aG wheel tassdaq |
|  |
| #id tassdaq |

screenshot

|  |
| --- |
| Provide the SUDO permission to particular users |
| vi /etc/sudoers |

screenshot

|  |
| --- |
| Provide the SUDO permission to particular users |
| #visudo |
| %sudo ALL=(ALL:ALL) ALL |
| newuser ALL=(ALL:ALL) ALL |

**System Information:**

|  |
| --- |
| **Lscpu**  #Display information about the CPU architecture. |
| **Lsblk**  #List block devices. |
| **Lshw**  #List hardware configuration. |
| **Lsusb**  #List USB devices. |
| **Lsmod** #Show the status of loaded kernel modules. |

**File System Operations:**

|  |
| --- |
| **Mount** #Mount a filesystem. |
| **Umount**  #Unmount file systems. |
| **Fdisk**  #Partition table manipulator. |
| **Mkfs**  #Build a Linux filesystem. |
| **Fsck**  #File system consistency check and repair. |

**Process Management:**

|  |
| --- |
| **nice**  #Run a command with modified scheduling priority. |
| **renice**  #Alter priority of running processes. |
| **Jobs**  #Display status of jobs. |
| **Bg**  #Send a job to the background. |
| **Fg**  #Bring a job to the foreground. |

**Shell Scripting:**

|  |
| --- |
| **Bash**  #GNU Bourne-Again SHell. |
| **Sh** #Bourne Shell. |
| **Chmod** #Change file mode bits. |
| **Source** #Execute commands from a file in the current shell. |
| **Export** #Set an environment variable. |
| **Echo**  #Display a line of text. |
| **Read**  #Read a line from standard input. |

|  |
| --- |
| **#chmod a+rwx(read|write|Executeable) file.tx** |
| a=all |
| u=user |
| g=group |
| o=other |

**Text Editors:**

|  |
| --- |
| **vi/vim** #Visual editor. |
| **Nano** #Simple text editor. |
| **Emacs**  #Extensible, customizable text editor. |

**Remote Access:**

|  |
| --- |
| **telnet**  #User interface to the TELNET protocol. |
| **ftp**  #File Transfer Protocol. |
| **Rsh**  #Remote shell. |
| **Rsync**  #Remote file synchronization. |

**System Monitoring:**

|  |
| --- |
| **Htop**  #Interactive process viewer. |
| **Iotop**  #Monitor I/O usage by process. |
| **Nload** #Monitor network traffic bandwidth. |
| **Iftop**  #Display bandwidth usage on an interface by host. |

**Miscellaneous:**

|  |
| --- |
| **Watch**  #Execute a program periodically, showing output Fullscreen. |
| **Xargs**  #Construct argument list(s) and execute utility. |
| **Killall** #Kill processes by name. |
| **Bc** #An arbitrary-precision calculator language. |
| **Echo**  #Display a line of text. |
| **Logout**  #Exit a login shell. |
| **Man**  #Format and display the on-line manual pages. |
| **Whatis**  #Display manual page descriptions. |
| **Whereis**  #Locate the binary, source, and manual page files for a command. |
| **Which**  #Locate a command. |

**Bash Shell Scripting**

Example-1 (display hello Hysec Learning)

|  |
| --- |
| #vi file1.sh  #!/bin/bash  echo "Hello, Hysec Learning!"  chmod 777 file1.sh #give the permission to file  ./hello.sh run the script |

Example-2 (This script takes user input and checks if the input matches a specific value)

|  |
| --- |
| **Vi file100.sh**  **#!/bin/bash**  **echo "Please enter your name:"**  **read name**  **if [ "$name" == "tassdaq" ]; then**  **echo "Hello, tassdaq!"**  **else**  **echo "Hello, stranger!"**  **fi** |

**Linux Bash Scripting**

**Create EC2 amazon linux**

**mkdir hysec**

**display hello world**

**vi test.sh**

**#!/bin/bash**

**echo "Hello World"**

**chmod 777 test.sh**

**./test.sh**

**------------------------------------------------------------**

**add 2 numeric number**

**vi test2.sh**

**#!/bin/bash**

**# Add two numeric value**

**((sum=25+35))**

**#Print the result**

**echo $sum**

**chmod 777 test2.sh**

**./test2.sh**

**-----------------------------------------------------------**

**add 2 numeric number**

**vi test3.sh**

**#!/bin/bash**

**# Add two numeric value**

**((sum=200+300))**

**#Print the result**

**echo $sum**

**chmod 777 test3.sh**

**./test3.sh**

**----------------------------------------------------------------------**

**create Directory than change directory create directory within directory**

**vi test4.sh**

**#!/bin/bash**

**mkdir rojae**

**cd rojae**

**mkdir canada**

**chmod 777 test4.sh**

**./test4.sh**

**---------------------------------------------------------------------------**

**create Directory than change directory create directory within directory**

**vi test5.sh**

**#!/bin/bash**

**mkdir amazon**

**cd amazon**

**mkdir tassdaq**

**chmod 777 test5.sh**

**./test5.sh**

**------------------------------------------------------------------**

**vi sal.sh**

**#!/bin/bash**

**salary=1000**

**expenses=800**

**#Check if salary and expenses are equal**

**if [ $salary == $expenses ];**

**then**

**echo "Salary and expenses are equal"**

**#Check if salary and expenses are not equal**

**elif [ $salary != $expenses ];**

**then**

**echo "Salary and expenses are not equal"**

**fi**

**-------------------------------------------------------------------------------**

**vi update.sh**

**#!/bin/bash**

**sudo yum update**

**sudo yum upgrade**

**------------------------------------------------------------------------------------------**

**vi system.sh**

**#!/bin/bash**

**systemctl start httpd**

**systemctl status httpd**

**------------------------------------------------------------------------------**

**vi httpd.sh**

**#!/bin/bash**

**yum install httpd -y**

**systemctl start httpd**

**systemctl status httpd**

**vi remove**

**#!/bin/bash**

**yum remove httpd -y**

**------------------------------------------------------------------------------------------**

**vi docker**

**#!/bin/bash**

**yum install docker -y**

**systemctl start docker**

**systemctl status docker**

**--------------------------------------------------------------------**

**vi info.sh**

**#!/bin/bash**

**echo "Date"**

**date**

**echo "Uptime"**

**uptime**

**echo "Memory Usage"**

**free -m**

**echo "Network Usage"**

**ip a**

**-----------------------------------------------------------**

**#!/bin/bash**

**display\_disk\_space() {**

**echo "Disk Space Usage:"**

**df -h**

**}**

**display\_top\_summary() {**

**echo "Top Processes Summary:"**

**top -b -n 1 | head -n 20**

**}**

**main() {**

**display\_disk\_space**

**echo ""**

**display\_top\_summary**

**}**

**# Call the main function**

**main**

**-----------------------------------------------------------------------------**