**Linux Operating System**

Linux is a Unix-like, open source and community-developed operating system (OS) for computers, servers, mainframes, mobile devices and embedded devices. Linux has been around since the mid-1990s and has since reached a user-base that spans the globe. It is supported on almost every major computer platform, including x86, ARM and SPARC, making it one of the most widely supported operating systems. Every version of the Linux OS manages hardware resources, launches and handles applications, and provides some form of user interface. The enormous community for developers and wide range of distributions means that a Linux version is available for almost any task. Linux is highly configurable and depends on a modular design that enables users to customise their own versions of Linux.

**Use of Linux in CyberSecurity:**

Linux is the OS of choice for many hackers. Why, you may ask? Because it’s open-source, less prone to malware, lightweight, portable, and very compatible with multiple hacking tools.

Linux has many distros to choose from and most can be modified as the user pleases without any restrictions. A number of distros commonly used by hackers are Kali Linux, Parrot, BlackArch, and Archstrike. But don’t stop there, the options are unlimited.

Hackers use Linux because:

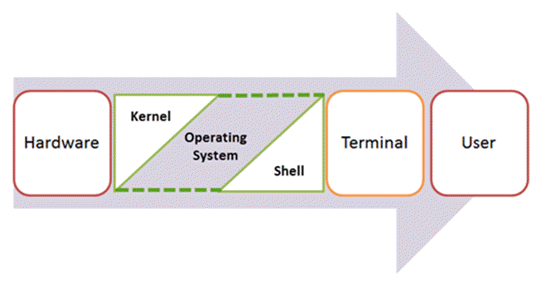
1. Linux is Open Source
2. Linux is Transparent
3. It offers granular control, ie almost infinity amount of control over the system
4. Most hacking tools are written for Linux

Kali Linux was developed by Offensive Security as a hacking operating system built on a distribution of Linux called Debian. Kali was designed for penetration testers and hackers and comes with a significant complement of hacking tools. It has hundreds of tools pre installed, saving you the hours it would take to download and install them yourself.

**Components of Linux OS:**

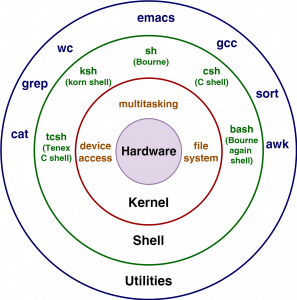
An Operating is made of many components, but its two prime components are –

* Kernel
* Shell



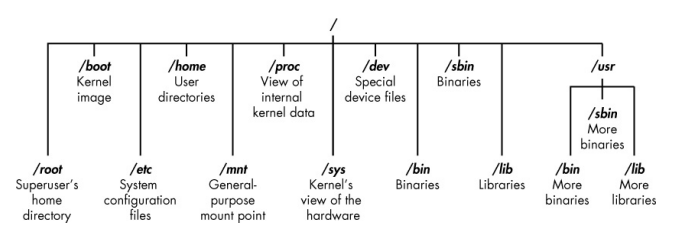
*Kernel*- A Kernel is at the nucleus of a computer. It makes the communication between the hardware and software possible. While the Kernel is the innermost part of an operating system, a shell is the outermost one.

*Shell*- This is an environment and interpreter for running commands in Linux. The most widely used shell is bash, which stands for Bourne­again shell, but other popular shells include the C shell and Z shell. The Shell wraps around the delicate interior of an Operating system protecting it from accidental damage. Hence the name Shell. The command line interface that forms a connection between the shell and the user is called Terminal



**LINUX FILE STRUCTURE:**

The file system in Linux is somewhat different from other OS and uses a logical filesystem instead. At the very top of the filesystem structure is /, which is often referred to as the root of the filesystem (note: this is different from the root user).



Under this directory, the following are some of the most important subdirectories:

| /root | The home directory of the all powerful root user |
| --- | --- |
| /etc | Generally contains the Linux configuration files—files that control when and how  programs start up. The etc folder (short for etcetera) is a commonplace location to store system files that are used by your operating system |
| /proc | It doesn't contain 'real' files but runtime system information (e.g. system memory, devices mounted, hardware configuration, etc). For this reason it can be regarded as a control and information centre for the kernel |
| /home | The user’s home directory |
| /mnt | Where other filesystems are attached or mounted to the filesystem |
| /media | Where CDs and USB devices are usually attached or mounted to the filesystem |
| /bin | Where application binaries (the equivalent of executables in Microsoft Windows)  reside |
| /lib | Where you’ll find libraries (shared programs that are similar to Windows DLLs) |
| /var | The "/var" directory, with "var" being short for variable data, is one of the main root folders found on a Linux install. This folder stores data that is frequently accessed or written by services or applications running on the system. |
| /tmp | This is a unique root directory found on a Linux install. Short for "temporary", the /tmp directory is volatile and is used to store data that is only needed to be accessed once or twice. |

**LINUX SHELL COMMANDS:**

These are instructions given to the Shell to interact with the operating system and perform various tasks. The basic structure of a command is as follows:

command [option] [arguments]

*Command*- It is the actual action you want to execute

*Options*- These are additional and optional means to modify the behaviour of a command. They are usually preceded by a single (-) or double hyphen (--)

*Arguments*- These are inputs or data that the command operates on. They include files, directories or other items that the command needs to process. Some commands do not require arguments

#Some basic commands:

1. **whoami:**

It returns the user account you are logged in as.

1. **echo:**

It is used to display a message or print the value of a variable to the terminal.

Syntax: echo [string]

1. **clear**:

It is used to clear the contents of the terminal screen, providing a clean and empty console. When you run the clear command, it doesn't delete any command history or terminal session; it simply clears the visible output from the screen.

1. **man (Manual):**

It is used to display the manual pages for various commands, utilities, and system calls. Manual pages, often referred to as man pages, provide detailed documentation and information about the usage, options, and functionality of commands and programs on a Unix-based system.

Syntax: man [command]

1. **help:**

It provides a brief overview of the command's usage and available options. When you append --help to a command, it typically prints a help message to the terminal, giving you information about how to use the command and what options are available.

Syntax: [command] --help

#Commands for Navigating File System:

1. **pwd (Present Working Directory)**:

It returns your location within the directory structure, i.e. your current working directory.

1. **cd (Change Directory)**:

It is used to change the current working directory to another location (given as an argument by the user).

Syntax: cd [path]

Note: The directory path can be an absolute (starting from the root directory) or relative (relative to the current directory) path. In file paths, certain symbols have special meanings:

* ‘~’ (Tilde): Represents the home directory of the current user
* ‘-’ (Hyphen): Represents the previous working directory
* ‘.’ (Dot): Represents the current directory
* ‘..’ (Double Dot): Represents the parent directory

1. **ls (Listing):**

It is used to list the files and directories in a specified directory. If no directory is specified, it lists the contents of the current working directory.

Syntax: ls [options] [directory]

* ls -l : It is used to display detailed information about files, including permissions, ownership, and modification times.
* ls -a : The -a option stands for all. This command is used to list all files and directories in the current working directory, including hidden ones (whose names start with a dot)

#Creating and Handling Files:

1. **touch**:

It is used to create empty files and update the access and modification timestamps of existing files.

Syntax: touch [options] filename(s)

1. **cat (concatenate)**:

It is used to concatenate and display the contents of files. It is a versatile command that can be used for various purposes, such as displaying the contents of a file, combining multiple files, creating new files, and more.

Syntax: cat [options] filename(s)

Note: Using > and >> operators:

* cat > filename : This command is used create a file, or overwrite an existing file with given string
* cat >> filename : This command is used to append to a new or an existing file

1. **mkdir (make directory):**

It is used to create new directories (folders).

Syntax: mkdir [options] directorynames(s)

1. **rmdir (remove directory):**

It is used to remove empty directories (folders).

Syntax: rmdir [options] directoryname(s)

Note: rmdir can only remove directories that are empty

1. **rm (remove):**

It is used to remove files or directories.

Syntax: rm [options] [directory/filename]

* rm -r : The -r option stands for "recursive" and is used to remove directories and their contents. This command removes the specified directory and its contents. Thus this command can be used to remove a non-empty directory.
* rm -f : The -f option with the rm command stands for "force," and it is used to forcefully remove files without prompting for confirmation, even if the files are write-protected or if the user does not have the necessary permissions.

1. **cp (copy):**

It is used to copy files and directories from one location to another

Syntax: cp [options] source(s) destination

* cp -r : to copy all the contents of a directory recursively into another directory
* cp -i : prompts for confirmation before overwriting an existing file in the destination

1. **mv (move):**

It is used to move or rename files and directories. It can be used to relocate files or directories from one location to another or to change their names.

Syntax: mv [options] source(s) destination

* mv -v : the option -v (verbose) provides more detailed information, including the names of files as they are moved.

1. **head:**

It is used to display the first few lines of a text file. By default, it prints the first 10 lines of each specified file to the standard output.

Syntax: head [options] file(s)

* head -n [number] file.txt : The -n option allows you to specify the number of lines to display
* head -d ‘,’ file.txt : The -d option allows you to specify a custom delimiter for lines. This command displays lines separated by a comma in a CSV file.

1. **tail**:

It is used to display the last few lines of a text file. By default, it prints the last 10 lines of each specified file to the standard output.

Syntax: tail [option] file(s)

Similar to head, tail command can also use the same options to specify the number of lines and other options to customise

1. **file:**

It is used to determine the type of a file or the format of a file system. It examines the actual contents of a file and provides information about its type, such as whether it is a text file, binary file, image, archive, etc.

Syntax: file [options] file(s)

1. **ln (link):**

It is used to create links between files. It can create two types of links: hard links and symbolic (or soft) links.

A hard link is essentially another name for an existing file. Changes to one hard link are reflected in all hard links to the same file because they all point to the same inode (data structure on a filesystem).

Syntax: ln [options] source target

A symbolic link is a separate file that contains a path to the target file or directory. It is more flexible than a hard link and can link across different filesystems.

Syntax: ln -s [options] source target

Note: Deleting the original file does not delete hard links to that file. The file is only truly deleted when the last hard link is removed. Deleting the original file or moving it to another location can break symbolic links, as they reference the file by its path.

#Finding and Filtering Files:

1. **whereis**:

It is used to locate the binary, source, and manual page files for a specified command. It provides information about the locations where the executable binary, source code, and manual pages for a given command are stored on the system.

Syntax: whereis [binary file]

1. **find:**

It is the best tool used for searching for files and directories in a directory hierarchy based on various criteria. It is a versatile command with many options, allowing you to specify conditions such as file names, types, sizes, and modification times.

Syntax: find [start\_directory] [options] [expression]

* find [start\_directory] -name [file\_name] : This command searches for “file\_name” in the specified directory “start\_directory”
* find [start\_directory] -type f : This command finds all regular files in the specified directory.

Note: Wildcards are characters used to represent one or more other characters in a string. They are widely used in command-line interfaces to specify patterns for matching filenames or commands.

* \* (Asterisk): Matches any sequence of characters (including none).
* ? (Question Mark): Matches any single character.
* [ ] (Square Brackets): Matches any one of the characters enclosed in the brackets.
* ! (Exclamation Mark): Used for negation (excluding a pattern)
* { } (Curly Braces): Generates multiple patterns separated by commas.

1. **grep (Global Regular Expression Print):**

It is a powerful utility used for searching text patterns in files.

Syntax: grep [options] pattern [file(s)]

* grep -i pattern file(s) : This option is used to ignore case distinctions in both the pattern and input files
* grep -r pattern directory : This is used to recursively search subdirectories
* grep -n pattern file(s) : Displays line numbers along with the lines that match the pattern.
* grep -e pattern1 -e pattern2 file(s) : Allows you to specify multiple patterns.
* grep -c pattern file(s) : Displays the count of matching lines

#Installing and package management:

1. **apt (Advanced Package Tools):**

It is a set of tools used for package management in Debain-based Linux Distributions. It is used for installing, updating, upgrading, and managing software packages on the system.

* sudo apt update : This command updates the local package database with the latest information about available packages and their versions.
* sudo apt upgrade : This command upgrades all installed packages to their latest versions. It may prompt you for confirmation before proceeding.
* sudo apt install packagename : This command installs a new package.
* sudo apt remove packagename : This command removes a package from the system. However, it leaves the configuration files intact
* sudo apt purge packagename : This command removes a package and its configuration files from the system.
* apt search searchterm : This command searches for packages containing the specified search term in their name or description.
* apt show packagename : This command displays detailed information about a specific package.

Note: yum (Yellowdog Updater, Modified) is a package management tool used in Red Hat-based Linux distributions. It includes tools for package management similar to apt, like update, install, remove, search, info

1. **wget (Web Get):**

It is used for downloading files from the web.

Syntax: wget [options] URL

* wget -O desired\_file\_name URL : The -O option allows you to specify a different name for the downloaded file.
* wget -b URL : The -b option runs wget in the background, allowing you to continue using the terminal.
* wget -i file\_list\_of\_URL : The -i option allows you to specify a file containing a list of URLs to download.

#Security and File Permissions:

1. **sudo (Super User Do):**

It allows you to temporarily elevate your current user account to have root privileges without a need to change your identity. It is used as a prefix to other commands to run that command as the root user.

Syntax: sudo [command]

* sudo -i : This command simulates a login as the root user, including setting the root user's environment variables and working directory. It essentially starts a new shell session as if you had logged in directly as the root user.
* sudo -s : This command starts a new shell with root privileges but does not simulate a full login. It does not read the root user's login scripts or set the environment variables exactly as if you had logged in as root. It maintains the current user's environment and working directory.
* sudo -l :

Note: The user running the sudo command must be configured in the sudoers file (which is usually located at /etc/sudoers or in the /etc/sudoers.d/ directory) to have the necessary privileges, and the user's password is used for authentication.

1. **su (Switch User):**

It is used to switch to another user account but maintains the current user's environment.. When used without specifying a username, it defaults to switching to the superuser (root).

Syntax: su [username]

* su - : The option (-) hyphen is used to simulate a full login with the environment variables and working directory of the target user. This also defaults to root if no username is passed after (-)

The working of [su -] command is very similar to [sudo -i], however, the first asks for root’s password whereas the latter requires the current user’s password and checks sudo privileges

#Process Management:

1. **ps:**

It is used to provide information about the currently running processes on a system. It displays information such as process IDs (PIDs), resource usage, and other details.

Syntax: ps [options]

* ps aux : Provides a detailed list of all processes running on the system.
* ps -e : Display information about other users' processes as well.

1. **kill:**

It is used to terminate or signal processes. It sends signals to processes, allowing you to manage their execution.

Syntax: kill [options] ProcessID

Note: The pkill command is a convenient way to send signals to processes based on their name rather than their Process ID.

#System and Hardware Information:

1. **uname (Unix Name):**

It is used to display system information. It stands for "Unix Name" and provides various details about the system, including the operating system name, kernel version, network node hostname, machine hardware architecture, and more. When you run the uname command without any options, it typically provides information about the operating system.

Syntax: uname [options]

* uname -a : Used to display all information available
* uname -s : Used to show the kernel name
* uname -r : Used to show the kernel release
* uname -v : Used to display the kernel version
* uname -m : Used to show the machine hardware architecture

1. **lsb\_release :**

It provides LSB (Linux Standard Base) and distribution-specific information.

Syntax: lsb\_release [options]

* lsb\_release -a : With the "-a" option, provides detailed information about the Linux distribution, including the distributor ID, description, release number, and codename.

Note: uname -a gives details about the kernel and machine hardware, while lsb\_release -a provides information about the Linux distribution.

1. **lshw (List Hardware):**

It is a powerful and comprehensive tool that provides detailed information about various hardware components of your system. lshw can display information about the system's CPU, memory, storage devices, network interfaces, and more.

Syntax: sudo lshw [options]

* sudo lshw -class cpu : Used to show detailed information about a specific hardware category (e.g., CPU)
* sudo lshw -class network : Used to show information about the network interfaces

1. **free:**

It is used to display information about the system's memory usage, both physical and swap memory.

Syntax: free [options]

#Networking:

1. **ip:**

It is used for configuring and displaying network interfaces. It is part of the iproute2 package and provides functionalities that were traditionally handled by multiple commands like ifconfig, route, and others.

* ip link show : Show information about all network interfaces

1. **ping:**

It is used to test the reachability of a host (usually a computer or server) on an Internet Protocol (IP) network. The ping command sends Internet Control Message Protocol (ICMP) Echo Request messages to the target host and waits for Echo Reply messages

Syntax: ping [options] host

1. **dig:**

It is a DNS (Domain Name System) tool used to query DNS servers and retrieve information about domain names, IP addresses, and other DNS records. It is commonly used for troubleshooting DNS-related issues and obtaining detailed DNS information.

Syntax: dig [options] [domain]

* dig -x [IP address] : Used to perform a reverse DNS lookup by querying the PTR (Pointer) record for an IP address:

1. **netstat:**

It provides information about network connections, routing tables, interface statistics, masquerade connections, and more.

Syntax: netstat [options]

1. **tcpdump:**

It is a powerful packet analyser. It allows users to capture and display network traffic on a network interface. tcpdump is widely used for network troubleshooting, monitoring, and security analysis.

Syntax: tcpdump [options] [optional expression to specify which packets to capture]