Linux Boot Process

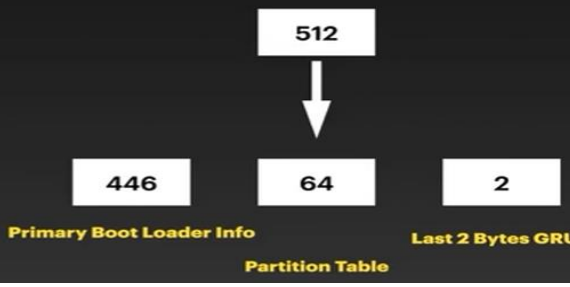
BIOS	Basic Input/Output System executes MBR
MBR	Master Boot Record executes GRUB
GRUB	Grand Unified Bootloader executes Kernel
Kernel	Kernel executes the /sbin/init program
Init	Init executes Runlevel programs
Runlevel	Runlevel programs are executed from /etc/rc.d/rc*.d/

BIOS — Basic Input/Output System executes MBR

- Basic Input/ Output System
- Perform Integrity check, searches load and execute boot loader program
- Searches - CD /DVD , SD Card and HDD
- Boot Sequence - F12, F2

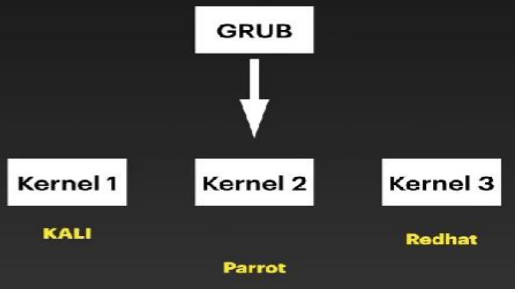
MBR — Master Boot Record

- Located in the first sector of bootable disk
- /dev/sda or /dev/hda
- less than 512 bytes in size. It has 3 components



GRUB — Grand Unified Boot loader

- Choose the kernel image to load and executed.
- Loads splash screen and executes default image.
- Loads and executes the kernel



Kernel

Kernel executes the /sbin/init program

- Mounts the root file system as in grub.conf
- Executes the init program located in sbin folder
- Loads the file system

INIT

Init executes Runlevel programs

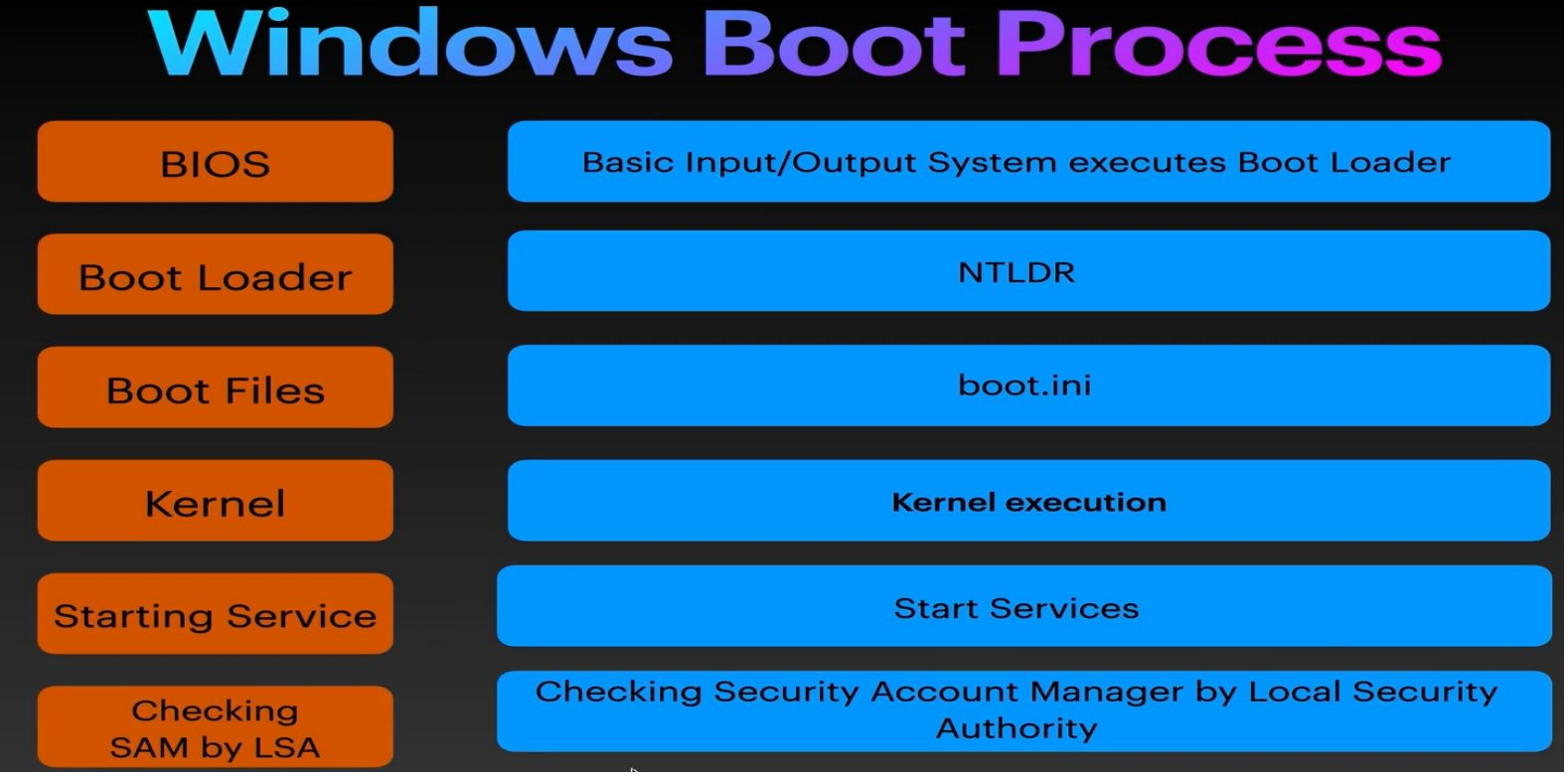
- Decides the Runlevels
- Which programs to be loaded at startup
- Runlevels 0 to 6

Runlevel 0	SHUTDOWN/ HALT
Runlevel 1	SINGLE USER NO NETW
Runlevel 2	MULTI USER NO NETW
Runlevel 3	MULTI USER + NETW
Runlevel 4	UNDEFINED
Runlevel 5	X11
Runlevel 6	REBOOT

RUNLEVELS

Runlevel programs are executed from /etc/rc.d/rc*.d/

- System now executes programs based on run levels
- Here are directories of programs - /etc/rc.d/rc*.d/
- Programs start with s used for startup
- Programs start with k used for shutdown



Bios:- Execute the boot loader

Boot Loader:- Check if the NTLDR files are available or not

Boot Files:- Check the important require file like boot.ini which is used to successfully boot the system

Kernel:- Responsible for Kernel execution of the operating system, also known as heart of the system

Starting service:- Check all the required initial services and start all the required services by the operating system to function.

Checking SAM by LSA:- when we login our password are hashed and saved to a secure location by SAM, so here its check if the password we are entering is right or wrong

Steps to Reset Root Password without password

- ➔ Booting into the GRUB menu
- ➔ After selecting the second option, you need to press the “e” in order to edit the boot entry
- ➔ Find the Keyword “Linux”, search for the “ro” and replace it with “rw”. Find quite splash and replace with init=/bin/zsh (for old kali init=/bin/bash).
- ➔ To check RW permission on the root partition, press the “F10”.
- ➔ Type “mount” and check rw is granted.

Resetting kali linux root password

- ➔ Passwd root
- ➔ exec /sbin/init

Steps to Setup Grub Password

Generate Encrption Password: (pbkdf2 is a cryptography encryption method)

- ➔ grub-mkpasswd-pbkdf2
- ➔ cp /etc/grub.d/40_custom /etc/grub.d/40_custom.old
- ➔ vim /etc/grub.d/40_custom
- ➔ set superusers=”root”
- ➔ password_pbkdf2 root HASH_HERE
- ➔ grub-mkconfig -o /boot/grub/grub.cfg
- ➔ init 6

Package Managers

Why is it so important?

- Working with file archivers to extract package archives.
- Ensuring the integrity and authenticity of the package.
- Verifying checksums and digital certificates of packages.
- Downloading, Installing, or Updating existing software from a software repository.
- Managing dependencies to ensure a package is installed with all packages it requ
- Grouping packages by function.

Different Package Managers

Yum:- Red hat based operating systems /

sudo yum install package_name / sudo yum remove package_name

Brew:- Mac os based operating systems

brew install package_name / brew uninstall package_name

Rpm:- Red hat based operating systems

rpm -I path (/root/Downloads/curl.rpm)

rpm-e package_name

Dpkg:- Debian based operating systems

dpkg -i path (/root/Downloads/curl.deb)

dpkg -r package_name (remove without configuration file)

dpkg -purge package_name (remove with configuration file)

Apt:- Debian based operating systems

apt install package_name

apt remove package_name

apt search search_text (search softwares, search_text is searching phrase eg – graphical ftp)

apt show package_name (shows information about software)

apt update

apt upgrade

Git:- git clone github_path

Linux Hardware information commands

- ➔ history (see history of all the commands we have used)
- ➔ dmesg (display messages in the kernel ring buffer)
- ➔ cat /proc/cpuinfo (display cpu information)
- ➔ cat /proc/meminfo (display memory information)
- ➔ free -h (see how much memory is free and how much is used, -g see in gb, -m see in mb)
- ➔ lspci -tv (display the pci devices if connected)
- ➔ lsusb -tv (display the usb devices if connected)
- ➔ dmidecode (display bios information)
- ➔ du (disk usage)
- ➔ df (disk usage but in a much proper way and short understandable way)

Performance monitoring commands

- ➔ top (see top process running on the system) q (to quit)
- ➔ htop (see and manage interactive process management)
- ➔ vmstat 1 (see information about virtual memory)
- ➔ cat /var/log/syslog (display system logs)
- ➔ lsof (list of open commands/file on my operating system)

Userinfo and management commands

- ➔ id (display the user and group id of current user)
- ➔ last (display the last users that have logged into the systems)
- ➔ who (display who is currently logged in to the system)
- ➔ w (display current user and what task they are doing)
- ➔ useradd -c “users name” -m username (create user)(useradd -c “Dipanshu Chhanikar” -m deep)
- ➔ cat /etc/passwd (check if the user is created or not)
- ➔ userdel username (delete user) (userdel deep)
- ➔ groupadd group_name (create group) (groupadd teamx)
- ➔ usermod -aG group_name username (add user to a group)
- ➔ su username (switch to different users)

Network commands

- ➔ ifconfig (display the current network interface configuration information)
- ➔ iwconfig (display network interface)
- ➔ iwconfig wlan0 mode monitor (change the adapter to monitor mode)

File, Directory, and Deletion commands

- ➔ ls (list all the content present in that directories) (la -al to
- ➔ ls -al (detail list) ls -all (detail list)
- ➔ pwd (display current directory)
- ➔ cd directory_name (go to that directory) cd .. (goes one step back)
- ➔ mkdir directory_name (create directory)
- ➔ touch file_name (create empty file) touch {1..10} (create 10 empty files)
- ➔ nano file_name (edit file in nano editor) gedit file_name (edit file in gedit editor)
- ➔ cat file_name (print all the text written in a file)
- ➔ cp file_name destination_folder (copy the file to destination folder)
- ➔ cp -r source_ directory_name destination_ directory_name (copy directory)(cp -r dir2 dir1)
- ➔ rm file_name (delete the file)
- ➔ rmdir directory_name (delete the directory)
- ➔ rm -rf directory_name (delete the directory even if the directory consist some files)
- ➔ tree directory_name (display directories and files in a tree format)
- ➔ mv source_file_or_folder destination_folder (move file)

Permission commands

- ➔ chmod a+rwx filename (giving user, group and other read, write, execute permission)
- ➔ chmod u+rwx filename (giving user read, write, execute permission)
- ➔ chmod g+rw filename (giving group read, write permission)
- ➔ chmod o+r filename (giving user read permission)
- ➔ chmod a-rwx filename (removing user, group and other read, write, execute permission)
- ➔ chmod u-rwx filename (removing user read, write, execute permission)
- ➔ chmod g-rw filename (removing group read, write permission)
- ➔ chmod o-r filename (removing user read permission)
- ➔ chmod 777 filename (giving all the permission in a different way)
- ➔ chown username filename (change user ownership for a file)
- ➔ chown :groupname filename (change group ownership for a file)
- ➔ chown -R username:groupname directory_name(change user/group ownership for a directory)

SSH

- ➔ ssh ipaddress (remote connection using ssh) (ssh 192.168.12.1)
- ➔ ssh user@ipaddress (remote connection with specific username)(ssh root@192.168.12.1)
- ➔ ssh user@ipaddress -p portnumber (remote connection using specific port number by default 22) (ssh ipaddress -p 22)