LINUX BOOTING PROCESS

The boot process is made up of a series of actions that a computer has to go through ranging from powering on to the complete loading of the operating system (OS) for optimal functionality.

- BIOS (Basic Input/Output System) or UEFI (Unified Extensible Firmware Interface
- POST (Power-on-self-test)
- Boot loader
- GRUB (Grand Unified boot loader)
- Kernel
- Init
- System Daemons

Here is a brief explanation as to how this works;

Stages	Description.
• **BIOS or UEFI** (Basic input/output system)	Right after clicking the power button on your computer, a program called a "BIOS or UEFI" boots up, these are basically pieces of software that gets all the main part of the computer i.e. keyboard, monitor, mouse, hard drives and so on, ready for action. The key difference between BIOS and UEFI is their approach to disk storage and accuracy.
• **POST** (Power-on-self-test)	This test makes sure all the hardware(s) are up and running before fully turning everything on. If there is any hardware malfunction. It detects and populates an "error message" on the screen notifying the user of underlying risk(s)
• Boot Loader	If everything looks good after the POST, the boot loader runs certain checks on the hard-drives first, or USD or Disk drive, if doesn't find anything on the hard-drive. In a BIOS system, the boot loader can be located in the hard-drive called the "*Maser boot record" (MBR)* but for the UEFI,

	there is a separate partition for this called the *"efi. Boot loader file"*. The key function of a boot loader is: - Locate the operating system kernel on the disk - Load the kernel into the computer's memory - Start running the kernel code.
• **GRUB** (Grand unified boot loader)	This is found inside the boot loader, alternative to this is the LILO which is now phased-out. The GRUB can handle multiple operating systems expressing them with graphical and/or text-based menus and has a bunch of variety options. After the GRUB loads itself up, it inserts the Linux kernel into memory
• **Kernel**	The kernel is the core of the operating system, managing hardware resources, providing and controlling interactions between hardware and software. It initializes the system hardware, mounts the root file system, and begins the user-space initialization process. It does this by decompressing itself into the memory and checks the hardware, then load device drivers and loads kernel modules.
• **Initialization (Init)**	The initialization process (init) is a fundamental part of the booting process in Unix-like operating systems. Its primary responsibility is to initialize the system and bring it to a functional state. Linux systems, including those based on Red Hat Enterprise Linux (RHEL), have transitioned to using systemd, which serves as a replacement for init, introducing a more flexible and efficient approach to managing system initialization.
• **System Daemons**	This is also known as background processes or services, play a vital role in the booting process and ongoing operations. These daemons are specialized programs that run in the background, providing essential services to the system and users. They perform specific tasks, such as managing hardware, handling system events, or providing network services.