# Modern Linux Boot Process

## Power On

### BIOS/UEFI Initialization

- When you power on your computer, the BIOS or UEFI firmware is loaded from non-volatile RAM (NVRAM).

- The BIOS/UEFI is responsible for initializing hardware components, performing a Power-On Self-Test (POST), and starting the boot process.

### Probe for Hardware

- The BIOS/UEFI probes or detects the hardware components connected to the system, including the CPU, memory, storage devices, and peripherals.

### Select Boot Device (HDD, USB, PXE,...)

- After hardware detection, the BIOS/UEFI allows you to select the boot device from which the operating system will be loaded.

- You can choose from options like booting from a local disk, a network server, or other storage media.

### Load the Selected Kernel

- The boot loader loads the selected Linux kernel into memory.

- The kernel is the core of the operating system and is responsible for hardware initialization and managing system resources.

### Determine Which Kernel to Boot

- GRUB or the chosen boot loader determines which Linux kernel to load.

- This decision is typically based on the kernel's version and configuration specified in the boot loader's configuration files.

### Load Boot Loader (e.g., GRUB2)

- The BIOS/UEFI or UEFI firmware loads the chosen boot loader. In many Linux systems, GRUB (Grand Unified Bootloader) is commonly used as the boot loader.

- GRUB provides a menu to select the operating system to boot or automatically loads the default Linux kernel.

### Identify EFI System Partition

- If the system is using UEFI firmware, it identifies the EFI System Partition (ESP) on the boot device.

- The ESP contains boot loaders and other essential boot-related files.

### Instantiate Kernel Data Structures

- After loading, the kernel initializes its data structures, sets up memory management, and prepares for the transition to user space.

### Start Init/systemd as PID1

- The kernel starts the init system or systemd as the first user-space process (PID 1).

- In modern Linux distributions, systemd has become the default init system, responsible for managing system services and processes.

### Execute Startup Scripts

- The init system or systemd executes startup scripts and initializes system services and daemons.

- These scripts and services include those responsible for setting up networking, mounting filesystems, and other system-level tasks.

### Running System

Once all initialization and startup tasks are completed, the Linux system is fully booted and ready to use.

Users can log in, and the system is in a running state, allowing users to run applications and perform tasks.