

# Linux Startup

Proces spuštění počítače

- [Spuštění GNU/Linux](#)

# Spuštění GNU/Linux

Dovolte mi nejprve vyjasnit firmware a partition schéma.

## Firmware

Firmware je to, co se spustí, když stisknete tlačítko napájení, jeho hlavním účelem je inicializovat váš hardware a poté načíst bootloader. Dva nejpoužívanější jsou:

**BIOS:** To je to, co najdete ve starých počítačích (téměř všechny počítače před rokem 2010), byl velmi základní až do té míry, že postrádá standardy, takže jinými slovy nepořádek. Vše, co dělá, je inicializovat základní hardware a načíst první sektor na spouštěném disku.

**UEFI:** Standardizováno v roce 2007, aby nahradilo BIOS, má nad hardwarem mnohem větší kontrolu, než jen jeho inicializaci (jako přetaktování, nastavení křivek ventilátoru,...). UEFI může běžet v režimu kompatibility (CMS) pro zpětnou kompatibilitu se stávajícími instalacemi MBR, kde funguje jako BIOS, který načítá pouze první sektor spouštěcího disku.

Podle specifikací UEFI:

- Musí být schopen číst souborový systém FAT (FAT12/16/32)
- Musí mít oddíl naformátovaný pomocí FAT 12/16/32 a označený ESP (EFI System Partition), který funguje jako kontejner pro bootloadery.
- Musí být schopen číst zaváděcí položky

## Partition Schéma

Partition schéma je to, co organizuje celý váš disk (budu mu říkat disk, protože to znamená pevné disky, disky SSD, flash disky, optické disky...), a řídí, kolik oddílů může mít disk a dokonce jak velké jsou. Dva, které se používají hlavně:

**GPT:** Toto je moderní schéma rozdělení, které řeší předešlé chyby staré varianty (hlavně limit oddílů a disk větší než 2 TB), většinou je spojeno s UEFI.

**MBR:** Toto je tradiční schéma rozdělení dat do éry IBM, MBR je první sektor (512 bajtů), ve kterém je uložen primární zavaděč (zavaděč fáze 1, více později) a tabulka oddílů, která organizuje 4 primární oddíly disku. Nepleťte si to s tabulkou souborového systému, která organizuje vaše data uvnitř jednoho oddílu.

Nyní pojďme k těm fázím:

**V čistém UEFI/GPT** může UEFI číst tabulku GPT a najít oddíl ESP (podle příznaku ESP) a poté spustit GRUB (**fáze 1**), GRUB pak načítá OS (**fáze 2**) pomocí svého konfiguračního souboru ( `/boot/grub/ grub.cfg` ).

**V systémech BIOS/MBR a BIOS/GPT** BIOS načte spouštěcí kód do MBR (fáze 1), ale protože je tak malý, aby se vešel do jakéhokoli zavaděče, obsahuje pouze ukazatel na základní grub (fáze 1.5), který se nachází:

- V případě BIOSu/MBR: volné místo bez oddílů hned po MBR
- V případě BIOSu/GPT speciální oddíl bez souborového systému (obvykle po 1. oddílu)

Protože BIOS je tak základní, že nezná žádnou představu o oddílech nebo souborových systémech, tento zavaděč **fáze 1.5** ( `core.img` ) obsahuje primární ovladače potřebné ke čtení tabulky souborového systému oddílu `/boot` a vyhledání konfiguračního souboru grub ( `/boot/grub /grub.cfg` ), který pak GRUB volitelně používá, nabízí uživateli zaváděcí nabídku se seznamem nainstalovaných OS/kernelů, poté načte jádro výchozího OS nebo to, co uživatel vybere v zaváděcí nabídce (**stage2** ).

## Co přijde po GRUB?

GRUB načte jádro a `initramfs` do paměti ( `initrd` byl používán ve starších systémech, ale nahradit jej `initramfs` kvůli jeho omezením). Jádro poté připojí obsah `initramfs` (po jeho dekomprimaci) a připojí jej jako počáteční kořenový souborový systém (**který je uložen v paměti**), tento počáteční kořenový souborový systém obsahuje data potřebná pro jádro k inicializaci zařízení a spuštění programu nazvaného `init`, že připojení skutečný kořenový souborový systém ( `/` ) a převezmeme kontrolu nad zaváděním zbytku systému.

## Starší obrázek boot procesu

Tady ještě příkládám **starý obrázek boot procesu**, který by se mohl hodit pro pozdější úpravy. Některé části jsou pravdivé i dnes.



