

1 Linux Bootvorgang

Nach dem Einschalten eines Computers wird durch das BIOS (Basic input output system) der Power on self Test durchgeführt. Der POST dient dazu, Hardware zu erkennen und ansprechbar zu machen. So wird unter anderem ermittelt, welcher Prozessor verbaut ist, ob ein Bildschirm oder eine Tastatur angeschlossen ist, wie viel Arbeitsspeicher (RAM) zur Verfügung steht und welche Laufwerke vorhanden sind. In dieser Phase des Bootvorgangs ist es möglich, durch Drücken einer bestimmten Taste in die BIOS-Einstellungen zu kommen. Diese Einstellungen sind je nach System mehr oder weniger umfangreich. Hier lässt sich zum Beispiel die Bootreihenfolge der Festplatten ändern oder die primäre Grafikkarte umstellen. Je nach Modus, verläuft der weitere Bootvorgang etwas anders ab.

Ist ein Monitor angeschlossen, werden Fehler auf diesem ausgegeben. Da das Grafiksystem nicht funktionieren oder nicht vorhanden sein kann, werden Fehlermeldungen als akustische Tonfolgen ausgegeben. Nach dem erfolgreichen POST sucht das BIOS auf dem ersten Sektor (Master-Boot-Record) der eingestellten Bootfestplatte nach dem Bootloader und gibt die Kontrolle an diesen weiter.

Da der Platz im Bootsektor mit 440 Bytes sehr gering ist, arbeiten die meisten Bootloader (bei Ubuntu ist das derzeit GRUB 2) in mehreren Stufen. Die erste Stufe kennt dabei nur die genaue Lage und die Länge der zweiten Stufe auf der Festplatte. Die zweite Stufe enthält Dateisystemtreiber und kann damit zum Beispiel Dateien aufrufen und weitere Stufen laden und/oder ein Bootmenü anzeigen. GRUB 2 kann auch andere Bootloader per Chainload aufrufen. So ist es möglich, das GRUB 2 den Bootloader von Windows oder anderen Betriebssystemen lädt.

Der Bootloader oder Bootmanager hat folgende Aufgaben:

- Er bietet die Möglichkeit, Bootoptionen an den Kernel zu übergeben
- Den Betriebssystemkernel (/boot/vmlinuz-KERNELNUMMER) laden und starten
- Die initiale RAM-Disk (Initramfs) im Speicher erstellen

Danach übernimmt der Kernel den weiteren Bootprozess. Nachdem der Kernel gestartet wurde, wird der Inhalt von /boot/initrd.img-KERNELNUMMER-generic in die RAM-Disk entpackt und als erstes das Root-Dateisystem eingehangen. Somit kann der Kernel auf die in diesem Dateisystem enthaltene Verzeichnisstruktur zugreifen und startet als ersten Prozess das Programm /sbin/init. Init steuert nun den weiteren Startvorgang und lädt benötigte Kernelmodule (Treiber). zu erkennen.

Werden beim Ablauf von init schwerwiegende Fehler entdeckt, startet BusyBox. Busybox ist ein Minibetriebssystem und stellt eine Shell zur Eingabe und Ausgabe bereit. Der Befehlssatz von busybox ist jedoch eingeschränkt und die enthaltenen Befehle können von denen des eigenen Systems abweichen. Durch die Eingabe von help kann man sich alle vorhandenen Befehle anzeigen lassen. Nach der Überprüfung der in der Datei /etc/fstab enthaltenen Partitionen wird das Root-Dateisystem durch pivot-root von der RAM-Disk auf die Root-Partition des realen Datenträgers verlagert. Die RAM-Disk ist nach Abschluss

der Verlagerung nicht mehr notwendig und der benutzte Speicher wird wieder freigegeben. Nach dem Start von `init` und der Dienste ist der eigentliche Bootvorgang abgeschlossen.

Werden beim Ablauf von `init` schwerwiegende Fehler entdeckt, startet Busy-Box. Busybox ist ein Minibetriebssystem und stellt eine Shell zur Eingabe und Ausgabe bereit. Der Befehlssatz von busybox ist jedoch eingeschränkt und die enthaltenen Befehle können von denen des eigenen Systems abweichen. Durch die Eingabe von `help` kann man sich alle vorhandenen Befehle anzeigen lassen.