

## Inhaltsverzeichnis

Systemstart und Initialisierung.....	2
Ablauf .....	2
SysInit vs. SystemD .....	2
Verzeichnisbaum – „Alles ist eine Datei“ .....	2
Virtuelle Dateisysteme .....	2
Dateitypen .....	3
Inode (Index node) .....	3
Terminalbefehle .....	3
Apropos .....	3
Buchseiten.....	3

## Systemstart und Initialisierung

### Ablauf

1. Bootloader: Kernel und initramfs in RAM laden
2. Kernel laden: Hardware wird initialisiert
3. InitRamFS: Temporäres Root-Dateisystem zum laden von Modulen und Zugriff aufs echte FS
4. Mount Root FS: Kernel mountet Dateisystem nach /etc/fstab
5. Init starten: Erstes Prozessprogramm wird ausgeführt (init/systemd)
6. Dienste starten
7. Boot abgeschlossen: Bereit zur Anmeldung

### SysInit vs. SystemD

	<b>SysInit</b>	<b>SystemD</b>
<b>Grundidee</b>	- Traditionelles Init-System	- Modernes Init-System
<b>Startverhalten</b>	- Serielles Starten über Skripte	- Startet Parallel - Schnellere Startzeiten
<b>Konfiguration</b>	- Skripte in /etc/init.d	- Units-Dateien in /etc/systemd/system
<b>Abhängigkeiten</b>	- Kaum Abhängigkeitslogik	- Umfangreiche Abhängigkeitsverwaltung - Abbrüche bei Fehlern
<b>Logging</b>	- Über Syslog	- Zentralisiertes Logging über journalctl
<b>Resources &amp; Sicherheitsfeatures</b>	- Kaum integrierte Features	- Sandbox-Funktionen - Restriktionsmöglichkeiten
<b>Ökosystem</b>	- In vielen alten Distros - Leicht, aber weniger Funktionen	- Weit verbreitet - Oft Standard
<b>Vor- &amp; Nachteile</b>	Vorteile - Einfach, Transparent - Geringer Overhead Nachteile - Langsamer Start - Wenig Abhängigkeitslogik	Vorteile - Schneller Start - Robuste Abhängigkeitsverwaltung - Viele Funktionen Nachteile - Komplex - Kritik an Zentralisierung

### Verzeichnisbaum – „Alles ist eine Datei“

- Keine Laufwerksbuchstaben sondern Mountpoints
- Jeder Benutzer hat ein Home-Verzeichnis (~)

### Virtuelle Dateisysteme

- Belegen keinen Speicherplatz
- Zum Austausch zwischen Kernel und Anwendungen
  - o Nicht zum speichern
- Beispiele
  - o /dev -> Zugang zu Hardware (Festplatte)
  - o /proc -> Alle laufenden Prozesse

## Dateitypen

Dateityp	Kürzel	Informationen
Reguläre Datei	-	Texte, Grafiken, Audiodateien, Programme
Verzeichnisdateien	d	Ordner; Tabelldatei mit Dateiname -> Inode-Nummer
Blockdateien	b	Schnittstelle zu Geräten (z.B. Laufwerk); Zugriffe werden an das Gerät weitergeleitet
Zeichengeräte-dateien	c	Serieller Strom von Ein- & Ausgaben (z.B. Terminal)
Pipe-Datei	p	Ähnlich wie Pipes/Shell; Direkte Kommunikation zwischen Programmen ohne Zwischenspeicher
Symbolic link file	l	Enthält Pfadangabe (Verknüpfung)
Socket-Files	s	Bidirektionale Interprozesskommunikation innerhalb eines Rechners

## Inode (Index node)

- Datenstruktureintrag im Dateisystem
- Enthält Metadaten über eine Datei
  - o Dateigröße
  - o Dateityp
  - o Benutzerrechte
  - o Zeitstempel
  - o ...
- Jede Datei hat eine eindeutige Inode-Nummer

## Terminalbefehle

- Siehe Buch für die meisten Befehle

## Apropos

- Befehl um nach anderen Befehlen zu suchen anhand von Begriffen in deren Dokumentation
- Apropos kernel: Liefert alle Befehle, die „kernel“ in der Dokumentation enthalten

## Buchseiten

Seite	Inhalt
180	SystemD
28	Verzeichnisbaum
161	Sudo
63	Ls-Befehl
90	Touch-Befehl
65	Cp-Befehl
66	Mv-Befehl
67	Rm-Befehl
72	Mkdir-Befehl
41	Umgebungsvariablen
60	Man-Befehl
109	Grep-Befehl
124	Awk-Befehl
125	Sed-Befehl
100	Find-Befehl