

Computer & GNU/Linux Einführung

Teil 1

Simon Haller, Sebastian Stabinger, Nicolas Stolz

Inst. für Informatik
[IFI]

September 29, 2015

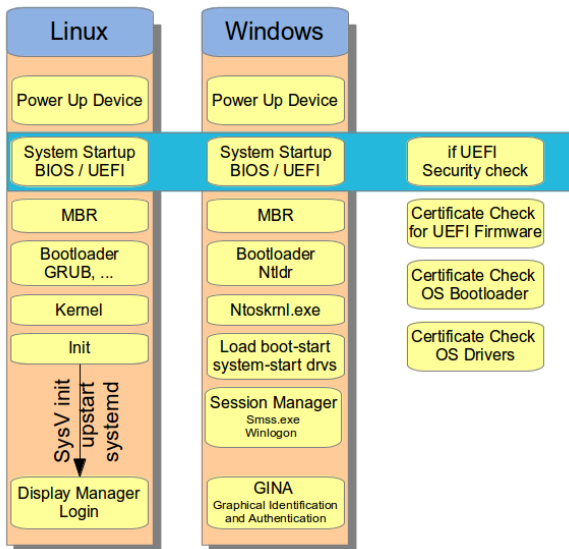


Starten von Windows

- ▶ Windows auf Linux Zugreifen (WinSCP, PuTTY)



Der Bootvorgang



Geschichte

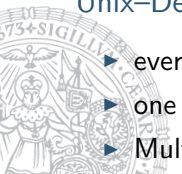
Die Anfänge von Unix

- ▶ MULTICS (1969)
- ▶ UNIX (1971)
- ▶ ab 1974 Spaltung in Derivate
 - ▶ IRIX
 - ▶ Solaris
 - ▶ HP/UX
 - ▶ AIX
 - ▶ BSD

<http://en.wikipedia.org/wiki/Unix>

Unix-Design-Paradigmen

- ▶ everything is a file
- ▶ one tool for one job
- ▶ Multiuser und Netzwerk



Einleitung

Linus und die ersten Schritte

- ▶ Start im Jahr 1991
- ▶ erster unter der GPL veröffentlichter Linux Kernel (Version 0.99) 1992
- ▶ Version 1.0 erst 1994



Kernel

Betriebssystem

- ▶ Verwaltung von Ressourcen (Prozessor, Hauptspeicher, Platte, Netzwerk, . . .)
- ▶ Betriebssystem vergibt diese Ressourcen
- ▶ Betriebssystem benötigt Ressourcen

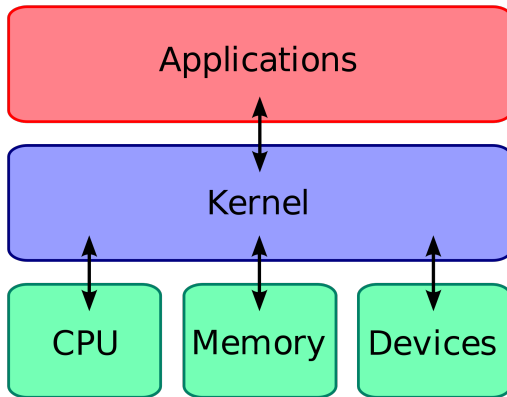
Unterschied zwischen Betriebssystem und Kernel

- ▶ Kernel ist Teil des Betriebssystems und verwaltet die Hardware
- ▶ Ein Betriebssystem besteht zudem noch aus grundlegenden Systemprogrammen, die dem Bootvorgang und der Konfiguration des Systems dienen.



Kernel

Interaktion zwischen Kernel Space, User Space und Hardware



Bobbo (2008)



Vom C-Programm zum Maschinencode

Programm in
einer höheren
Programmiersprache
(in C)

```
swap(int v[], int k)
{int temp;
  temp = v[k];
  v[k] = v[k+1];
  v[k+1] = temp;
}
```

Compiler

Programm in
Assemblersprache
(für MIPS)

swap:

```
multi $2, $5, 4
add $2, $4, $2
lw $15, 0($2)
lw $16, 4($2)
sw $16, 0($2)
sw $15, 4($2)
jr $31
```

Assembler

Programm in bi-
närer Maschin-
sprache
(für MIPS)

```
000000001010000100000000000011000
000000000000110000001100000100001
10001100011000100000000000000000
100011001111001000000000000000100
101011001111001000000000000000000
10101100011000100000000000000100
00000011111000000000000000001000
```

Adapted from Patterson and Hennessy (2011)



Kernel

Monolithischer Kernel

- ▶ Vorteile:
 - ▶ Schnell
- ▶ Nachteile:
 - ▶ Fehlende Flexibilität und Anpassungsfähigkeit

Mikrokern

- ▶ Vorteile:
 - ▶ Klare Schnittstelle und separierte Komponenten
 - ▶ Gerätetreiber mit Benutzerrechten
- ▶ Nachteile:
 - ▶ Langsamer (z.B. Hardwarezugriff)



Runlevels und Initialisierung

Runlevels

- ▶ 0 Shutdown.
- ▶ S Single-User-Runlevel
- ▶ 1 Einzelnutzerbetrieb (ohne Netzwerk, nur lokale Ressourcen)
- ▶ 2 Lokaler Mehrnutzerbetrieb (ohne Netzwerk, nur lokale Ressourcen)
- ▶ 3 Netzwerkbetrieb (keine grafische Oberfläche)
- ▶ 4 FREI
- ▶ 5 Standard (grafische Oberfläche)
- ▶ 6 Reboot



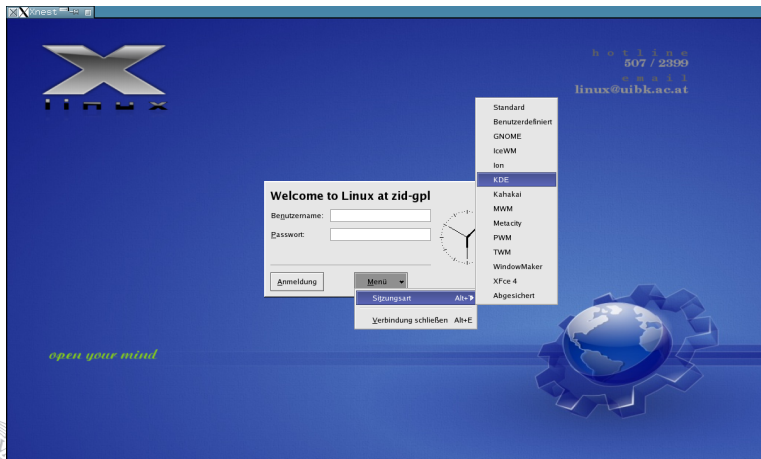
Runlevels und Initialisierung

rc.d, init.d

- ▶ `init` ist der erste Systemprozess
- ▶ In `/etc/inittab` beginnt
- ▶ In `rc[S0-6].d/` finden sich die Start und Stopskripte für die einzelnen Runlevels (Links auf `init.d` skripte)
- ▶ `/etc/rc.local` wird als eines der letzten Skripte ausgeführt



Desktopmanager & Windowmanager



Haller (2010)

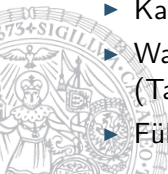
Das X-Window-System

X-Server

- ▶ Kontrolliert die Anzeige, die Tastatur, die Maus sowie andere Eingabegeräte
- ▶ Kommuniziert Ereignisse an den Client
- ▶ Ist verantwortlich für die Präsentation der Fenster und den verfügbaren Zeichensatz

X-Client

- ▶ Ist ein Anwendungsprogramm, das die grafischen Ein- und Ausgabe-Dienste des X-Servers nutzt
- ▶ Kann auf demselben oder einem entfernten Rechner laufen
- ▶ Wartet (passiv) auf Ereignisse (events) des X-Servers (Tastenanschläge, Mausbewegungen, Klicks, . . .)
- ▶ Führt Applikationscode aus



Relative und Absolute Pfadnamen

Relative Pfadnamen

Es wird von aktuellem Arbeitsverzeichnis ausgegangen

- ▶ `cd ./linuxkurs/Beispielverzeichnis`

Absolute Pfadnamen

Beginnen immer mit einem `/`

- ▶ `cd /home/simon/linuxkurs/Beispielverzeichnis`



Pfadnamen II

Das Arbeitsverzeichnis

- ▶ Ist das Verzeichnis, von dem relative Pfadnamen ausgehen.
- ▶ Wenn kein Pfadname angegeben wird, dann wird im CWD (current working directory) gesucht
- ▶ `$ cd Directory` wechselt in das Verzeichnis `Directory` (relativ oder absolut)
- ▶ `$ cd` alleine wechselt ins `$HOME`
- ▶ `$ pwd` print working directory
- ▶ Speziell:
 - ▶ `.` aktuelles Verzeichnis
 - ▶ `..` übergeordnetes Verzeichnis



Groß - und Kleinschreibung

Achtung
auf Groß - und Kleinschreibung!



Verzeichnisstruktur I

Root

<code>/bin</code>	Systemprogramme
<code>/dev</code>	Devices
<code>/etc</code>	Konfigurationsdateien
<code>/root</code>	Home für root
<code>/home</code>	Homes für User
<code>/home/franz</code>	
<code>/home/seopl</code>	



Verzeichnisstruktur II

Root

- `/lib` ‘‘Shared Libraries’’ für dynamisch gelinkte Programme des Betriebssystems

- `/mnt` Verzeichnisse mit Mount Points
- `/media`

- `/usr` (unix system ressourcen), Anwendungen, Manpages, alles wichtige für einen User
 - `/usr/bin`
 - `/usr/lib`
 - ...

- `/opt` meist kommerzielle Software

- `/proc` Prozess-‘‘Dateisystem’’. stellt Kernelinformationen in Form von Dateien dar.

- `/sbin` Systemprogramme, die mehr Rechte zum Ausführen benötigen



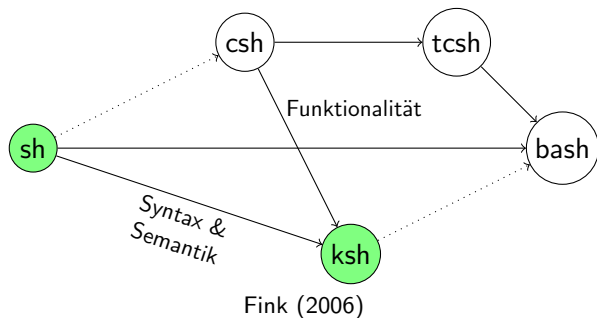
Verzeichnisstruktur III

LPCCS spezifisch

- ▶ `/usr/site` ... Universitätsweit zugängliche Programme
- ▶ `/scratch/cXXX/cXXXXXXX` ... Platz für temporäre Dateien und zum Datenaustausch (Unix-Rechtevergabe)
- ▶ `/scratch/.snapshot` ... Stündliches Backup von `/scratch`
- ▶ `/afs/zid1.uibk.ac.at` ... AFS Filesystem der LFU
 - ▶ `home/cXXX/cXXXXXXX` ... Home
 - ▶ `backup/cXXX/cXXXXXXX` ... Backup des Homes (Daily)
 - ▶ `share/cXXX/cXXXXXXX` ... Share-Verzeichnis für User (AFS-Rechtevergabe)



Shell Entwicklung



Das Bild zeigt eine schematische Entwicklung beginnend bei der Bourne-Shell (`/bin/sh`) (Ende der 70er Jahre) bis zur Bourne-Again-Shell (`/bin/bash`) Anfang der 90er.

POSIX (Portable Operating System Interface): bietet eine standardisierte Schnittstelle zwischen Applikationen und Betriebssystem.

Manpages und Infoseiten

- ▶ `$ man Befehl ...` liefert die Manpage
- ▶ `$ man -k Suchbegriff ...` sucht nach dem Wort in Manpages (alternativ `$ apropos`)
- ▶ `$ info Befehl`
- ▶ in `/usr/share/doc ...` finden sich Paket- u. Distributions spezifische Dokumentation zu Programmen

LPCCS spezifisch

- ▶ <http://www.uibk.ac.at/zid/systeme/linux>
- ▶ Linux-User mailing list: <http://lists.uibk.ac.at>

Im Netz

- ▶ <http://tldp.org>
- ▶ Foren, Usegroups, Chats, ... (Stichwort: HowTo)



References I

Bobbo (2008). Wikipedia.

[http://en.wikipedia.org/wiki/Kernel_\(computing\)](http://en.wikipedia.org/wiki/Kernel_(computing)).

Fink, M. (2006). Unix kurs.

Haller, S. (2010). Lpccs introduction course.

Patterson, D. and J. L. Hennessy (2011).

Computer organization and design. The hardware/software interface. With CD.
München: Oldenbourg Verlag.

