Der User root

- *Superuserprivilegien* geben Zugriff auf alle Rechnerresourcen und Dateien,. . ..
- Die *User-ID 0* besitzt die *Superuserprivilegien*.
- Der User *root* hat die *User-ID 0*.
- Nur benötigte Arbeiten unter *root* durchführen.

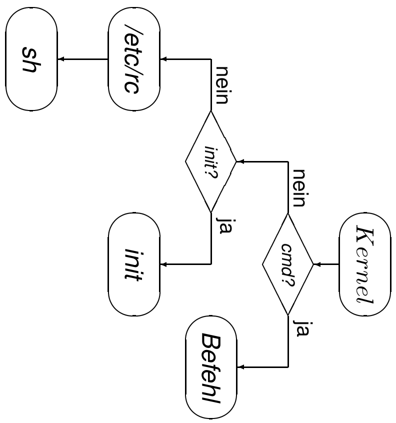
Aufgaben des Systemverwalters

- Systeminstallation
- Systemkonfiguration & Fehlerbehebung
- Installation neuer Systemkomponenten
- Userverwaltung (einrichten, löschen, ändern)
- Verwaltung der Rechnerresourcen (Drucker, NFS, Dateisysteme,. . .)
- System hoch- bzw. herunterfahren

Bootup-Sequence

- Nach dem Booten wird normalerweise das Programm `init` gestartet.
- Alternativ kann der Kernel ein anderes „Init-Programm“ ausführen (`init=`*command*).
- Es wird `/bin/sh` gestartet, falls `init` nicht ausgeführt werden konnte.

Bootup-Sequence



init

- `init` ist der Vater aller Prozesse.
- `init` startet Prozesse, die in der `/etc/inittab` aufgeführt sind
- `init` verwaltet mehrere *Runlevel*.

inittab

Syntax für einen Eintrag in der `/etc/inittab`:

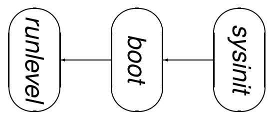
id:runlevels:action:process

Beschreibung der Felder:

id:	Zeichenketten der Länge 1–4 zur Identifikation
runlevels:	0, 1,...,6, A, B, C
action:	Schlüsselwort (u. a. bootwait, once, respawn,...)
process:	Namen des zu startenden Prozesses

inittab — Abarbeitung

1. Abarbeitung des sysinit-Eintrages
2. Abarbeitung aller boot- bzw. bootwait-Einträge
3. Abarbeitung aller once- bzw. respawn-Einräge



inittab — Schlüsselwörter

Eine Auswahl von inittab-Schlüsselwörtern:

respawn:	Der Prozeß wird nach jeder Terminierung erneut gestartet.
wait:	Der Prozeß wird einmal gestartet, init wartet bis dieser terminiert.
once:	Der Prozeß wird einmal gestartet.
boot:	Der Prozeß wird während des Systemboots gestartet.
bootwait:	Der Prozeß wird während des Systemboots gestartet, init wartet bis der Prozeß terminiert ist.
initdefault:	Legt den runlevel fest, der nach dem Booten aktiv ist.
sysinit:	Dieser Prozeß wird während des Systemboots ausgeführt. Er wird vor allen boot- bzw. bootwait-Prozessen gestartet.
ctrlaltdel:	Legt den Prozeß fest, der bei drücken von Ctrl-ALT-Del ausgeführt wird.

inittab — Beispiel

```
# inittab for linux
...
id:1:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit
...
1:1  :respawn:/etc/getty 9600 tty1
2:1  :respawn:/etc/getty 9600 tty2
3:135:respawn:/etc/getty 9600 tty3
4:135:respawn:/etc/getty 9600 tty4
...
```


sysinit-Shellskript

Einmaliger Aufruf — nur beim Start des Systems:

- Setzen der Systemzeit
- Aktivieren des Swap-Spaces (Auslagerungspartition)
- Checken der Filesysteme
- Aktivieren des Quota-Supports
- Mounten der Filesysteme

Runlevels

Der Zustand eines Unix-Systems wird durch Runlevels definiert:

```
...
l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
...
```

0 — Halt
1 — Singleuser Mode
2 — Multiuser Mode (ohne Netzwerk)
3 — Multiuser Mode (Standardmodus, wenn kein X11)
4 — Nicht benutzt
5 — Multiuser Mode mit X11
6 — Reboot

Lage der Startskripte im Dateisystem

/etc — das Verzeichnis für Konfigurationsdateien

```
├── inittab
└── rc.d/
    ├── init.d/
    ├── rc
    ├── rc.sysinit
    ├── rc0.d/
    ├── rc1.d/
    ├── rc2.d/
    ├── rc3.d/
    ├── rc4.d/
    ├── rc5.d/
    └── rc6.d/
```

Struktur des Filesystems

/ — das root Verzeichnis

bin	Zum Booten benötigte Befehle
boot	Dateien für den Bootloader
dev	Devicefiles
etc	Systemweite Konfigurationsdateien
home	Heimverzeichnisse der User
lib	Zum Booten benötigte shared libraries
mnt	Lage der Mountpoints
proc	Lage des proc-Dateisystems
root	Heimverzeichnis von root
sbin	Zum Booten benötigte Befehle
tmp	Temporäre Dateien
usr	Zweite Hauptebene
var	Veränderliche Daten

Die *Linux Filesystem Structure* (`fsstnd-1.2`) definiert die Verzeichnisse, deren Aufgabe und Inhalt.

Die /usr-Hierachie

In der `/usr`-Hierachie befinden sich die meisten Programme.

/usr — die zweite Hauptebene

X11R6	X Window System Version 11 Release 6
X386	X Window System Version 11 Release 5
bin	Die meisten Userprogramme
dict	Wörterbücher
doc	Beschreibungen zu Programmen
etc	Systemweite Konfigurationsdateien
games	Spiele, Spiele, etc.
include	Includefiles für C-Programme
info	Haupverzeichnis für das GNU-Infosystem
lib	Libraries
local	3. Hauptebene (lokale Hierarchie)
man	Online-Handbuch
sbin	Nicht zum Booten benötigte Rootprogramme
share	Systemunabhängige Daten
src	Quellcodes

- Eigene Software sollte immer unterhalb eines eigenen Verzeichnisses, am besten `/usr/local`, installiert werden.

Partitionsaufteilung

Eine geschickte Partitionsaufteilung vereinfacht die Pflege eines Systems:

Verzeichnis	Partitionsgröße	Option
/	20 MB	read-write
/home	Abhänig von der Anzahl der User	read-write
/var	Abhänig von der Anzahl der User	read-write
/usr	Rest (300MB)	read-only
swap	Doppeltes des Hauptspeicher	—

Dateisysteme mounten

- Dateisysteme werden mit `mount` in den Dateibaum eingebunden.
- Standardaufruf: `mount -t` *type device dir*
- Dateien, die von `mount` benutzt werden:

/etc

fstab	Informationen über die Filesysteme
mtab	Verzeichnis der gemounteten Dateisysteme
mtab˜	Lock-File
mtab.tmp	Temporärdatei

mount — Beispiele

Was muß ich tun, um . . .

- das CDROM-Laufwerk zu mounten:

```
# mount /mnt/cdrom
# _
```

- eine read-only Rootpartition schreibbar zu machen:

```
# mount -n -o remount,rw /
# _
```

- alle gemounteten Dateisysteme anzuzeigen:

```
# mount
/dev/sda1 on / type ext2 (rw)
/dev/sda3 on /home type ext2 (rw)
# _
```


fstab

Die `/etc/fstab` legt fest, welche Dateisysteme wo eingebunden werden sollen.

```
# /etc/fstab
#
# <device> <mountpoint> <filesystemtype> <options>                  <dump> <fsck>

/dev/sda1   /           ext2            defaults                         1   1
/dev/sda3   /home       ext2            defaults                         1   2
...
none        /proc       proc            defaults                         0   0
/dev/sr0    /mnt/cdrom  iso9660         noauto,user,nosuid,nodev,noauto 0   0
/dev/fd0    /mnt/floppy ext2            noauto,user,nosuid,nodev,noauto 0   0
...
```

- in der `/etc/fstab` eingetragene Optionen müssen beim Mounten nicht mehr angegeben werden

umount

- `umount` *mountpoint* löst ein Dateisystem aus dem Verzeichnisbaum.
- `umount -a` löst alle Dateisysteme aus dem Verzeichnisbaum.
- „ungemountete“ Dateisysteme können „ge-`fsck`-ed“ werden.
- Kein User darf sich im „zu unmountenden“ Verzeichnis befinden.
  - mit `fuser` *mountpoint* kann festgestellt werden, wer sich im Verzeichnis *mountpoint* befindet.

Benutzerverwaltung

- Useraccounts einrichten
- Useraccounts löschen oder sperren
- Accountinformationen modifizieren (Namen, Paßwörter,. . .)

Userverwaltung — Konzept

/etc

passwd	Userdatenbank im ASCII-Format
group	Verzeichnis der Gruppen

Bearbeiten der Dateien durch ...

- einen Editor
- verschiedene Tools (z. B. `passwd`, `useradd`, `adduser`, `control-panel`)

Userverwaltung — /etc/passwd

Syntax für einen passwd-Eintrag:

```
login_name:passwd:UID:GID:user_name:directory:shell
```

Beispiel:

```
...
test1:f3gfaEWS0v2x.:505:100:Test1:/home/test1:/bin/bash
test2:*f3gfaEWS0v2x.:506:100:Test2:/home/test2:/bin/bash
...
```

Tip: Es gibt **keine** Kommentarzeichen in der `/etc/passwd`

Userverwaltung — /etc/group

In `/etc/group` werden **zusätzliche** Gruppenzugehörigkeiten der Benutzer eingetragen.

Syntax für einen group-Eintrag:

```
group_name:passwd:GID:user_list
```

Beispiel:

```
...
linux::208:zendel,koppen,kleinhen,magnus,kester,pfeffer
...
```


Userverwaltung — Erweiterung des Konzeptes

- User können verschlüsselte Paßwörter lesen $\longrightarrow$ Sicherheitslücke
- Erweiterung des Konzeptes $\longrightarrow$ Shadowpaßwortkonzept

/etc

passwd	Userdatenbank in ASCII-Format ohne Paßwörter
shadow	Paßwörter der User
group	Verzeichnis der Gruppen ohne Paßwörter
gshadow	Paßwörter der Gruppen

- Verallgemeinerung des Shadow-Systems: PAM

Erweiterung der installierten Software

- Programm selbst kompilieren
  - Lesen der relevanten READMEs
  - Anpassen an den Filesystemstandard
  - Anlegen einer Dateiliste
- fertige Binaryversion installieren
  - Anlegen einer Dateiliste
- Packagemanager verwenden (z. B. Red Hat Packagemanager)

shutdown

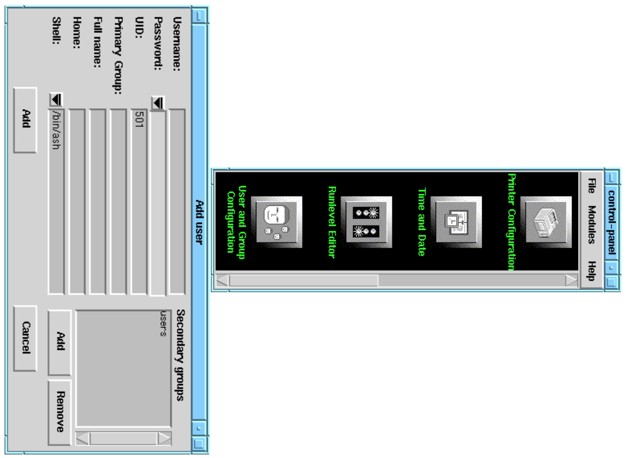
`shutdown` fährt das System herunter

- `shutdown -h` *zeit* wechselt in den *Runlevel 0* (Halt)
- `shutdown -r` *zeit* wechselt in den *Runlevel 6* (Reboot)

Werkzeuge für Systemverwalter

- Arbeiten sollten durch Skripte automatisiert werden (`sh`, `sed`, `awk`, `perl`, . . .)
- Regelmäßige Aufgaben mit `cron` erledigen lassen (z. B. `/tmp` löschen)
- komfortable AdminTools (z. B. `control-panel` von Red Hat)

Red Hat — control panel



Literatur

- UNIX Systemadministration
  von Æleen Frisch
  O'Reilly, ISBN 3-930673-04-5

- TCP/IP Netzwerk Administration
  von Craig Hunt
  O'Reilly, ISBN 3-930673-02-9

- Linux Wegweiser für Netzwerker
  von Olaf Kirch
  O'Reilly, ISBN 3-930673-18-5

- Linux HOWTOs
  `http://www.lunetix.de/docs/HOWTO/`

- diverse `man` pages