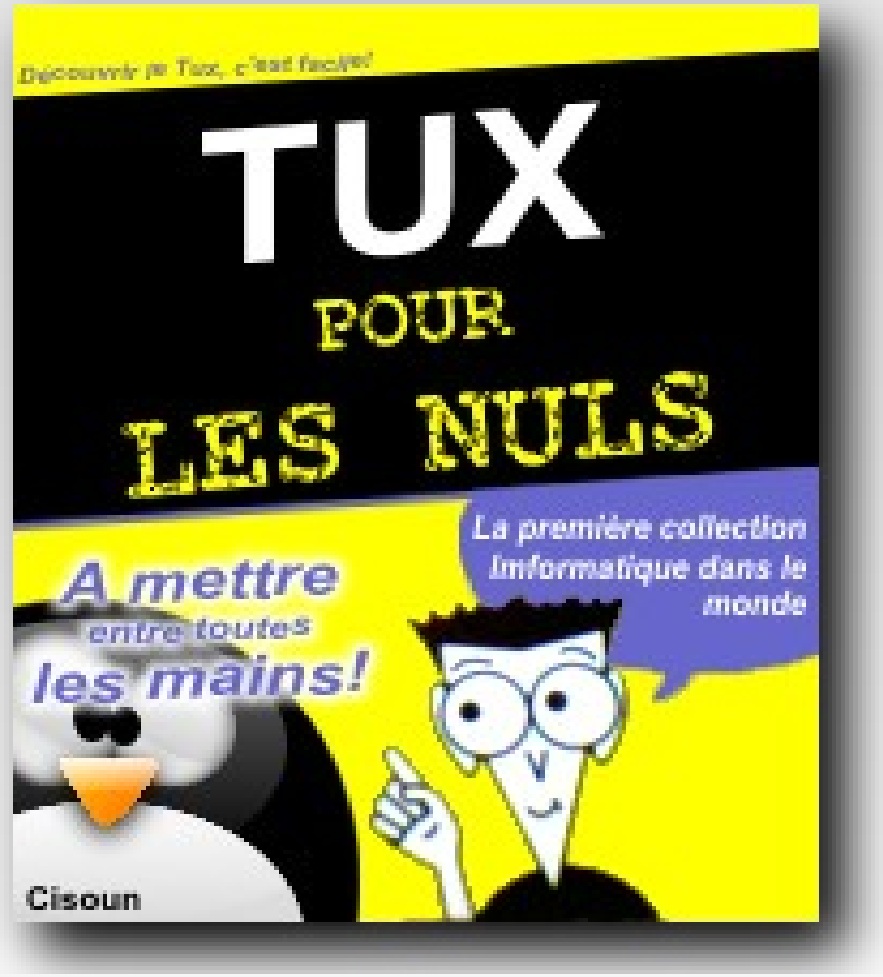
Tutorium Fortgeschrittene



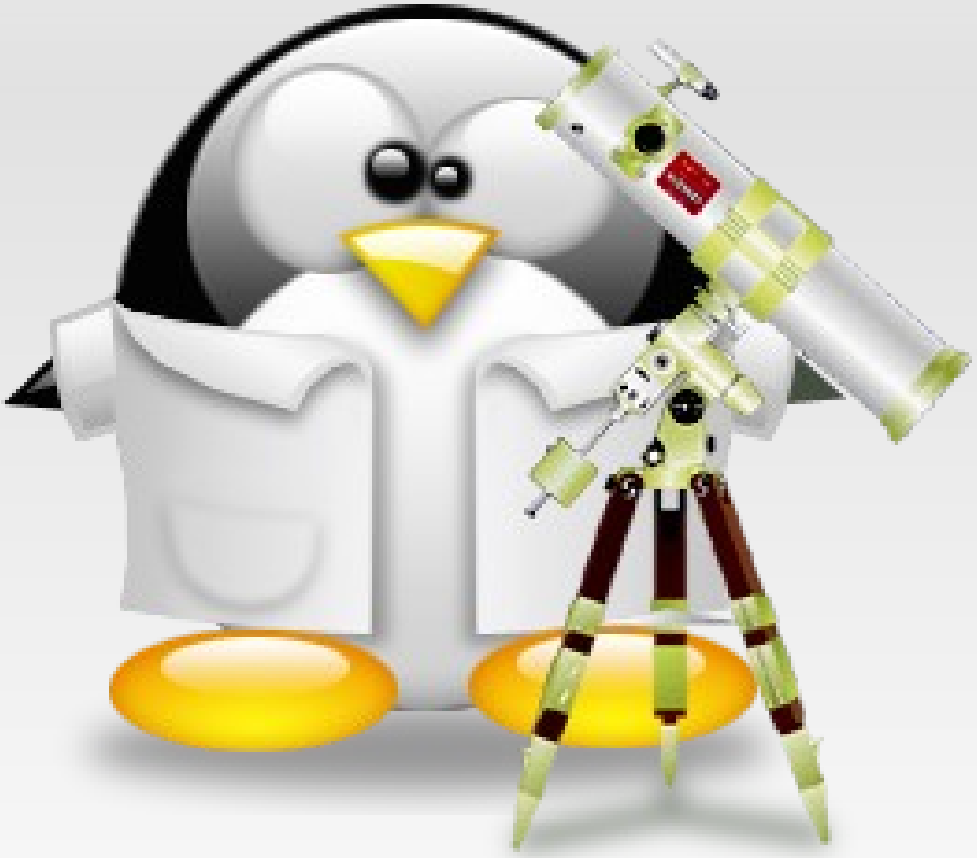
ubuntu

Übersicht

- Bootvorgang
- Kernel

ubuntu

Bootvorgang



Der Hardwareboot

- Basic Input-Output System → B.I.O.S.?
- PowerOn Selftest → P.O.S.T. ?
- Suche nach bootbaren Medium
- Hardwarespezifische Einstellungen im B.I.O.S.
- Aufbau des M.B.R. (hier erster Sektor der FP):
  - Bootloader
  - Partitionstabelle (64 Byte)
  - Signatur, auch Magic Number genannt, bedeutet, dass der erste Sektor (MBR) "ausführbar" ist.

Bootcode (Bootloader)

- Besteht aus Code zum Laden des B.S. oder weiteren Bootcodes (z.B. GRUB-Verzeichnis)
- Damit von einen Medium gebootet werden kann, muss dieser Code gültig sein
- Der Initiale Bootloader wird (meist) aus einer Referenz zum M.B.R. geladen
- Der Bootloader ist in den ersten 446 Byte untergebracht (z.B. GRUB referenziert, weil er größer ist)
- Die nächsten 64 Byte sind für Partitionstabelleneinträge belegt

Die Partitionstabelle

- Besteht aus vier 16 Byte langen Einträgen (stellv. für die max. 4 primären Festplattenpartitionen)
- Ein Partitionseintrag enthält:
  - ID
  - Partitionssektor (Anzahl, Beginn- und Endsektor)

Bootmanager

- Beispiel LILO:

Bootmanager

- Beispiel GRUB

ubuntu

Bootmanager

- Realisierung des Betriebssystemstart
- Er startet den Kernel des Betriebssystems und übergibt Parameter
- Wird in der Regel im MBR platziert
- Wird meist über eine Textdatei gesteuert werden
- Bekannte Vertreter unter Linux sind GRUB und bei älteren Systemen LILO

LILO

- LILO steht für LInux LOader
- kann verschiedene Betriebssyteme starten
- wird meist auf älteren Systemen verwendet
- Konfiguration über /etc/lilo.conf
- hat so gut wie keine Bedeutung mehr

GRUB

- GRUB bedeutet GRand Unified Bootloader
- Dieser universelle Bootloader wird von allen „großen“ Distributionen unterstützt
- Kann neben Windows und Linux auch noch viele weitere BS laden (z.B. Solaris, BSD etc...)
- Chainloader: Bedeutet, dass GRUB andere Bootloader laden kann, ähnlich wie Kettenelemente
- Partitionsänderung kann zum „Totalversagen“ von GRUB führen

GRUB – menu.lst

- enthält die komplette GRUB Konfiguration
- Beispiel:

```
title Ubuntu 9.10, kernel 2.6.28-11
uuid 51c3987080989876
kernel /boot/vmlinuz
initrd /boot/initrd.img
```

GRUB – menu.lst Parameter

- default 0 – zu ladendes Standard-BS
- timeout 10 – Zeit bis zum Start des Std.-BS
- title Linux – Titel des BS (kann frei gewählt werden)
- uuid 12-34-56-8 – Festplatten ID
- Kernel /boot/vmlinuz - Kernelimage
- initrd /boot/initrd.img – Image des Initrd
- root (hd0,0) – Partition des zu ladenden BS
- makeactive – Betriebssystem „Aktiv“ machen
- chainloader +1 – Übergibt an den nächsten Loader
- Aufbau steht im Beispiel

GRUB - Installation

- grub-install (--recheck) hd0 – Installiert Grub im MBR der ersten Festplatte
- update-grub – Durchsucht die Festplatte nach bootbaren Partitionen und schreibt sie in den MBR

Start des Kernels

- Prüfen des BIOS Registers und lädt entsprechende Hardwaretreiber
- Kernel swap daemon wird gestartet
- Rootdateisystem wird gestartet → wenn nicht gefunden → Kernelpanic
- Überprüfung des Hauptspeichers, CPU
- Auswertung der übergebenen Bootparameter

Init.d – und die Runlevel

- /sbin/init – Erster Prozess der vom Kernel initialisiert wird, erhält die PID 1
- Ist der Elternprozess aller anderen Prozesse
- Hauptaufgabe:

  Initialisieren von Runlevelscripts
- Bei Runlevelscripts handelt es sich um Scripts die beim Start oder Beenden eines Runlevels ausgeführt werden

Runlevel

- Runlevel: Ist ein Systemstatus bei dem bestimmte Dienste laufen
- Entsprechend starten/stoppen Init-Scripts die entsprechende Dienste zu den Runleveln
- In der Regel sind diese Scripte in der Syntax der BASH

Die Ausprägungen der Runlevel

- 0 halt (Do NOT set initdefault to this)

  bewirkt das Abschalten des Rechners

- 1 Single user mode

  Der Single-User-Mode ist gewissermassen ein Wartungmode in dem nur der Systemadministrator (also root) Zugriff hat.

- 2 Multiuser, without NFS

  normaler textorientierter Multi-User-Modus, aber ohne die Möglichkeit des "Filesharings". Andere Netzwerkfunktionen sind aber aktiv.

Die Ausprägungen der Runlevel

- 3 Full multiuser mode

  wie 2, nur das jetzt auch noch das "Filesharings" über das NFS (Network File System) unter UNIX/Linux-Maschinen möglich ist.

- 4 unused

- 5 X11

  wie 3. Nun läuft aber endlich auch die graphische Oberfläche X-Window (oder kurz X11).

Die Ausprägungen der Runlevel

- 6 reboot

  bewirkt einen Neustart.

- Die einzelnen Runlevels können mit dem Befehl ”init” aufgerufen werden
  - z.B.

    $ init 6

    Bewirkt einen Neustart des Systems

inittab

- Sagt den Bootloader, welcher Runlevel Standardmäßig geladen werden und welche Aktion ausgeführt werden sollen
- liegt in /etc/inittab (Ausnahme Ubuntu)
- Wenn init einen Prozess starten soll, wird auf /etc/initscript geprüft, ist dem so, wird der Prozess gestartet
- Die jeweiligen Runlevelscripts liegen in /etc/rc.d/rcX (X ∈ {0,1,2,3,5,6})

inittab - Aufbau

- ID: definiert eine ID, 1 – 4 Zeichen die zur Identifikation der Aktion
- Runlevel: Der für diese Aktion aufgerufen werden soll
- Aktion: legt die Aktion selbst fest, dabei wird ein Schlüsselwort angegeben um Init zu sagen was getan werden soll (z.B. boot, der Prozess wird während des Bootvorgangs ausgeführt).
- Kommando: Welcher Prozess gestartet werden soll

initab ? bei ubuntu

- bei ubuntu liegen die Startscripte für die einzelnen Runlevel im Verzeichnis ”/etc/event.d”
- ein Beispiel:
  - in der inittab wird ein Faxserver gestartet

    mo00:23:respawn:usr/local/sbin/faxgetty ttyIAX0
  - geht aber nicht bei ubuntu da die inittab nicht vorhanden ist

initab ? bei ubuntu

- Lösung
  - im Ordner /etc/event.d die entsprechenden Runlevelscripte suchen und folgenden Code hinzufügen

```
exec /usr/bin/iaxmodem ttyIAX0
```

initab ? bei ubuntu

- das entsprechende Script sieht dann wie Folgt aus
  - z.B. Runlevel 2

```
start on runlevel 2

stop on runlevel [!2]

console output

script
        set $(runlevel --set 2 || true)
        if [ "$1" != "unknown" ]; then
            PREVLEVEL=$1
            RUNLEVEL=$2
            export PREVLEVEL RUNLEVEL
        fi
      exec /etc/init.d/rc 2   // Starte Runlevel 2
        exec /usr/bin/iaxmodem ttyIAX0  // Starte Faxserver
end script
```

/etc/init.d

- in dem Ordner init.d sind alle Start- und Stop-Scripte abgelegt
- erstellen eines neuen Start-/Stop-Scriptes
  - erstellen der Datei mit einem beliebigen editor
  - Speichern der Datei im Ordner init.d
  - erstellen der entsprechenden Rechte für die Datei
    - chmod 755 beispieldatei
  - Die Datei hat folgenden Aufbau

/etc/init.d

```
#! /bin/sh
### BEGIN INIT INFO
# Provides:          Was macht das Skript?
# Required-Start:
# Required-Stop:
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: Kurze Beschreibung
# Description:       Längere Bechreibung
### END INIT INFO
# Author: Name <email@domain.tld>
```

ubuntu

/etc/init.d

```
# Aktionen

case "$1" in
    start)
        /opt/beispiel start
        ;;
    stop)
        /opt/beispiel stop
        ;;
    restart)
        /opt/beispiel restart
        ;;
esac

exit 0
```

ubuntu

/etc/init.d

- Anschließend mit dem Befehl update-rc.d die Links in den entsprechenden Runlevelordner anlegen
- in den einzelnen Runlevelordnern wird ein Link zu dem Init-Script erstellt
- es gibt zwei Arten von Links
  - S als Anfangsbuchstabe steht für Start in diesem Runlevel
  - K (Kill) als Anfangsbuchstabe steht für Stopen in diesem Runlevel
- die Nachfolgenden Ziffern legen die Startreihenfolge fest

Der init-Prozess im System

- Wie Stoppen wir einen init-Prozess im laufendem System?

ubuntu

Der init-Prozess im System

- Konsole öffnen
- mit cd /etc/init.d in das init.d Verzeichnis wechseln
- mit ls den Ordnerinhalt anzeigen lassen
- durch die Eingabe folgender Syntax den Prozess stoppen

  ./[Proz] stop //Stoppt den Prozess

  ./[Proz] start //Startet den Prozess

  ./[Proz] restart //Neustart des Prozesses

Kernel

ubuntu

Fragen?!

ubuntu

Hinweise

- Bei Problemen sofort nachfragen
- Linux-Stammtisch (Jeden 2. Mittwoch im Monat um 19 Uhr)
- Dr. Tux – Die Sprechstunde für Linux-Interessierte
  - Sprechzeiten Mittwochs

    15.00 bis 17.00 Uhr
  - Ort: Database Competence

    Center (Raum F0001)

ubuntu