

# Wichtige Befehle

- alias: Kann Kurzbefehle definieren
- apropos: Stichwortsuche in man-Texten
- bg: setzt mit Ctrl-Z unterbrochene Prozesse im Hintergrund fort
- cat: Unix-Analogon zu MS-DOS type
- cd: analog zu DOS
- chgrp: ändert Gruppe, siehe weiter hinten
- chmod: ändert Dateiattribute, “
- chown: ändert Benutzer+Gruppe, “
- cp: kopiert Dateien – siehe man-Page!
- dd: kopieren und konvertieren

# Wichtige Befehle

- df: Wieviel Speicherplatz ist noch frei?
- dmesg: Kernmeldungen, auch bei Start!
- du: Zeigt Platzverbrauch von Dateien und Verzeichnissen an
- fdisk: Partitiniert Festplatten
- fg: wie bg, setzt aber Programm im Vordergrund fort.
- file: rät den Dateiinhalt
- find: sucht Dateien
- free: zeigt freien Arbeitsspeicher an
- grep: sucht Zeichenketten

# Wichtige Befehle

- `groupadd`, `groupdel`: fügt Benutzergruppen hinzu oder löscht sie
- `halt`: stoppt den Rechner
- `head`, `tail`: die ersten/letzten Zeilen einer Datei anzeigen
- `kill`: tötet Prozesse
- `killall`: tötet alle Prozesse mit gleichem Namen
- `ldd`: zeigt Bibliotheksinformationen an
- `less`: zum Lesen von ASCII-Dateien

# Wichtige Befehle

- ln: erstellt Links auf Dateien
- lpr: Druckt Dateien aus der Kommandozeile
- locate: sucht Dateien in einer Datenbank
- ls,ll,l: zeigt Dateien an
- man: man-Pages
- mkdir: erstellt Verzeichnisse
- mv: verschiebt Dateien (Achtung mit Wildcards!!!)
- newgrp: Ändert die primäre Gruppe eines Benutzers

# Wichtige Befehle

- nice: Läßt Programme mit niedriger Priorität ablaufen.
- nohup: startet untötbare Prozesse
- passwd: ändert das Passwort
- ps: zeigt laufende Prozesse an.
- pstree: ebenso, aber als Baumstruktur
- reboot: Wer sich wieder nach Windows sehnt....
- renice: nice im nachhinein
- reset: Stellt Zeichensatz wieder her, rebootet das System nicht!

# Wichtige Befehle

- `rm`: löscht Dateien, `rm -r` macht tabula rasa!
- `rmdir`: löscht Verzeichnisse
- `rpm`: installiert Pakete von Hand nach
- `sort`: sortiert Daten
- `su`: ändert den Benutzer
- `tar`: Packt und entpackt Dateien:  
tar -xjvf Datei entpackt `.tar.bz2` Dateien  
tar -xzvf Datei entpackt `.tar.gz` und `.tgz` Dateien
- `top`: Zeigt Prozessorauslastung und mehr an.

# Wichtige Befehle

- mount/umount: bindet Geräte ein
- updatedb: aktualisiert die locate-Datenbank
- useradd: fügt Benutzer hinzu
- userdel: löscht Benutzer
- which: zeigt an, welches Kommando ausgeführt würde
- who: zeigt eingeloggte Benutzer an

# Dateirechte unter Unix

- Unices verwenden ein standardisiertes Rechtesystem, das “Eigentümer”, “Gruppe”, “Andere” bestimmte Rechte zuweisen, eine explizite Zuweisung an einzelne Benutzer wie unter Windows NT ist nicht möglich!
- Seine Gruppenzugehörigkeit kann man mit `groups` erfahren, seine primäre Gruppe mit `newgrp` ändern.
- Seinen eigenen Loginnamen kann man mit `whoami` erfahren.



# Interna

- Der Dateibesitzer und die Gruppe werden unter Unices nicht als Klartext wie bei Windows, sondern als sogenannte User und Group IDs abgespeichert, dies hat den Vorteil, daß sich Benutzer leicht umbenennen lassen.
- Root hat die User-ID 0.
- Systemdienst haben User-IDs  $0 < 500$
- Normale Benutzer haben User-IDs  $> 500$

# Dateirechte im Einzelnen

- Jeder dieser Gruppen kann man die Rechte “read”, “write”, “execute” geben, bei Dateien sind die Rechte selbsterklärend, bei Verzeichnissen steht “execute” für das Recht, in das Verzeichnis zu wechseln.
- Die Dateirechte können mit `chmod` gesetzt werden.
- Der Eigentümer kann mit `chown` gesetzt werden.

# Spezialbits

- Außerdem Zugriffsrechten gibt es noch Spezialbits:
- SUID: Das Programm läuft nicht unter der User-ID des Starters, sondern unter der User-ID des Programms (meist root), dies kann ein Sicherheitsrisiko darstellen!
- GUID: Analog mit der Gruppe
- Sticky: In einem Verzeichnis kann nur der Besitzer einer Datei diese löschen, nicht alle anderen, die in diesem Verzeichnis Schreibrechte haben.

# chown

- chown setzt den Benutzer und evtl. die Gruppe, der Aufruf lautet:  
`chown Benutzer[:Gruppe] Datei`
- die in eckigen Klammern stehende Teile können weggelassen werden (dies entspricht auch der Konvention, die in den man/info-Pages verwendet wird).
- Man kann als Benutzer keinem anderen Benutzer eine Datei zuweisen, ebenso keine Gruppe, der man nicht selber angehört.

# chgrp

- Mit `chgrp` kann man entsprechend die Gruppenzugehörigkeit einer Datei ändern, hier gelten die selben Einschränkungen wie bei `chown`.

# chmod

- chmod setzt die Zugriffs- und Spezialbits für Dateien und Verzeichnisse.
- Dabei sind zwei verschiedene Modi möglich:
- Im Oktalmodus werden alle Bits auf einmal gesetzt
- Im “bildlichen” Modus können die einzelnen Bits durch Symbole gesetzt werden.

# Oktaler Modus

- Die Bits werden direkt in Zahlen gesetzt:  
Das höchstwertige Bit pro Ziffer ist “read” (suid), das mittlere “write” (sgid), das niederwertigste “execute” (sticky), somit steht  
`chmod 4750 Datei`  
für: Benutzer darf, lesen, schreiben und ausführen, Mitglieder der Gruppe dürfen lesen und ausführen (mit suid), alle anderen dürfen gar nichts.
- Führende Ziffern dürfen weggelassen werden.

# “Bildlicher” Modus

- Hier können die einzelnen Bits mit Symbolen gesetzt werden:
- Dazu wird hinter chmod ein Buchstabe angegeben, der entsprechend für user, group, others, all steht, danach folgt jeweils ein + für Setzen oder ein – für Löschen, danach folgt der Kurzbuchstabe des zu setzenden Bits:  
read, write, execute, sticky, set-id



# chmod Beispiel

- `chmod g+rwx Datei`  
erlaubt es Leuten, die in der gleichen Gruppe sind, die Datei zu lesen, schreiben und auszuführen und , alle anderen Attribute bleiben unverändert.

# Switches für chmod und chown

- Sowohl chmod und chown akzeptieren den Switch -R, dieser besagt, daß die Änderungen auf alle Unterverzeichnisse und die darin enthaltenen Dateien angewendet werden.
- Dies unterscheidet sich von rm, bei dem der Switch -r heißt!

# Videoplayer installieren

- Wie schon vorher erwähnt, muß ein sinnvoller Videoplayer am besten selbst beschafft und compiliert werden.
- Der zur Zeit beste Player ist wohl mplayer.
- Er kann unter <http://www.mplayerhq.hu> heruntergeladen werden.
- Zur Verwendung unter KDE sollte das artsd-Entwicklerpaket (arts-devel) installiert werden.

# mplayer

- Die Installation ist unter /DOCS/German/\* erklärt.
- Zu shared libraries siehe Vorlesung!

# Filesystem Hierarchy Standard

- Linux verwendet einen standardisierten Verzeichnisbaum, hier wird nur grob auf die wichtigsten Verzeichnisse eingegangen!

# FHS

- /bin: elementare Programme für alle
- /boot: Dateien zum Booten des Systems
- /dev: Gerätedateien: Finger weg!
- /etc: systemweite Konfigurationsdateien
- /home: alle Homeverzeichnisse, auf die die einzelnen Benutzer Zugriff haben
- /lib: enthält die shared libraries, unbedingt in Ruhe lassen!
- /mnt, /media: Mountpunkte für Geräte
- /opt: zusätzliche Programme

# FHS

- /proc: Alles mögliche zu Prozessen,...
- /root: Das Homeverzeichnis von root
- /sbin: Befehle für root
- /tmp: temporäres Verzeichnis
- /usr: Unix shared resources: Nahezu das ganze System
  - /usr/local: Die meisten Programmpakete
  - /usr/src: Quelltexte von Programmen
- /var: Loggingdateien, Mails,...
- /windows: Mountpunkte für Windowspartitionen