

```

FFFFFFFFFFFFFFFFFFFF 00000000 SSSSSSSSSSSSSSS SSSSSSSSSSSSSS
F::::::::::::F 00::::::::00 S::::S::::S::::S S::::S::::S::::S
F::::::::::::F 00::::::::00 S::::S::::S::::S S::::S::::S::::S
FF::::::::FFFFFO::::0000::::05::::S SSSSSSSSSSSSS SSSSSSS
F::::F FFFFFFFO::::0 0::::05::::S S::::S
F::::F 0::::0 0::::05::::S S::::S
F::::::::FFFFFFFFFF 0::::0 0::::0 S::::SSSS S::::SSSS
F::::::::FFFFFFFFFF 0::::0 0::::0 SS::::SSSSSS SS::::SSSSSS
F::::::::FFFFFFFFFF 0::::0 0::::0 SSS::::SS SS::::SSSSSS
F::::::::FFFFFFFFFF 0::::0 0::::0 SSSSSSS::S SSSSSSS::S
F::::F 0::::0 0::::0 S::::S S::::S
F::::F 0::::0 0::::0 S::::S S::::S
FF::::::::FF 0::::0000::::05SSSSSS S::::SSSSSSSS S::::S
F::::::::FF 00::::::::00 S::::SSSSSSSSSS S::::SSSSSSSS
F::::::::FF 00::::::::00 S::::SSSSSSSSSS S::::SSSSSSSS
FFFFFFFFFFFF 00000000 SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

```

```

AAA GGGGGGGGGGGG
A:::A GGG::::::::G
A::::A GG::::::::G
A:::::A G::::GGGGGGG::G
A:::::A G::::G GGGGG
A:::::A A::::A G::::G
A:::::A A::::A G::::G GGGGGGGGG
A:::::A A::::A G::::G G::::G
A:::::A AAAAAAAA::A G::::G GGGGG::G
A:::::A AAAAAAAA::AG::::G G::::G
A:::::A AAAAAAAA::AG::::G G::::G
A:::::A A::::AG::::GGGGGGG::G
A:::::A A::::AGG::::G
A:::::A GGG::::GGG::G
AAAAAA AAAAAA GGGGG GGG

```

Terminal Basics Cheat Sheet

Jan-Marius Lenk, Christoph Parnitzke, Josef Schneider, Yannick Bungers

Free and Open Source Software AG
 Department of computer science
 5. Dezember 2016

Terminal vs. Shell

- Terminal: zeilenweise Eingabe von Befehlen
- TTY: Terminals via STRG+ALT+F[1-7]
 - auch die grafische Oberfläche läuft auf einer Shell
 - meist auf F7
- VTerm: virtuelles Terminal in grafischer Umgebung
- Shell: interpretiert Benutzereingabe

switch user

- Syntax: su [user]
- wechseln des Benutzers via Terminal

super user do

- Syntax: sudo [command]
- ausführen von [command] mit erweiterten Rechten
- Parameter:
 - -b (Ausführen von [command] im Hintergrund)
 - * Vorsicht bei interaktiven Programmen!!
 - -e (öffnet Datei zum editieren, erstellt tmp. Backup)
 - * gut beim Konfigurieren des Systems
 - * beugt Datenverlust vor
 - * Root wird neuer Besitzer der Datei
 - * statt 'sudo -e' 'sudoedit' möglich
 - -s (öffnet Shell)
 - -u [user] (führt Befehl als [user] aus)
 - -i (gibt root die .profile und .login des Benutzers)
 - * übergebene Befehle werden in Environment des Benutzers gestartet
 - * ohne '-c' wird interaktive Root-Shell geöffnet

apt-get und aptitude

- immer mit 'sudo' ausführen
- apt statt apt-get möglich
- aptitude ist etwas grafischer und hat Zusatzfunktionen
- wichtigste Befehle:
 - sudo apt install [pkg] ([pkg] installieren)
 - sudo apt update (Datenbank der Pakete aktualisieren)
 - sudo apt upgrade (mit Datenbank die Pakete aktualisieren)
 - sudo apt dist-upgrade (wie upgrade, löst dependency Probleme)
 - sudo apt remove [pkg] ([pkg] deinstallieren)

- sudo apt search [RegEx] (Paket suchen)
- sudo apt show [pkg] (zeigt Informationen zu [pkg])
- sudo apt list (zeigt Liste von Paketen)
- sudo apt list --installed (zeigt installierte Pakete)

man page

- Benutzerhandbuch des Systems
- Erklärung und sonstige Informationen zu installierten Paketen
- Sektionen:
 - NAME (Name des Befehls)
 - SYNOPSIS (Syntax des Befehls)
 - CONFIGURATION
 - DESCRIPTION (Beschreibung des Befehls)
 - OPTIONS (Optionen, die beim Aufruf mit angegeben werden können)
 - EXIT STATUS
 - RETURN VALUE
 - ERRORS
 - ENVIRONMENT
 - FILES
 - VERSIONS
 - CONFORMING TO
 - NOTES
 - BUGS (bekannte Fehler)
 - EXAMPLE (Beispiele für den Aufruf)
 - AUTHORS
 - SEE ALSO

Suchverlauf

1. ausführbare Programme und Shell Kommandos
2. Systemaufrufe (vom Kernel bereitgestellt)
3. Bibliotheksaufrufe (innerhalb von Programm-Bibliotheken)

4. spezielle Dateien (in /dev)
 5. Dateiformate und Konventionen (/etc/passwd)
 6. Spiele
 7. Sonstiges (Macropakete und Konventionen)
 8. System administrations Kommandos (nur Root)
 9. Kernel Routinen (kein Standard)
- Syntax: cd [path]
 - Wechsel in Parent-Directory '..' bzw. '../'
 - Wechsel in Home-Directory 'cd'

make directory

- Erstellen von Verzeichnissen
- Syntax: mkdir [option] [path]
- Optionen:
 - -p (erstellt Parent-Directory, falls es nicht existiert)
 - --mode=[Mode] (Modus mit dem Ordner erstellt wird)

list

- listet alle Dateien und Ordner im gegebenen Verzeichnis auf
- aktuelles Verzeichnis ist dabei Default
- Syntax: ls [option] [path]
- Parameter:
 - -a (listet auch versteckte Dateien und Verzeichnisse)
 - -l (listet auch Rechte, Besitzer, Gruppe, Größe, Name, last modified)
 - -c (listet Zeit der letzten Modifikation)
 - -f (gibt Inhalte in unsortierter Reihenfolge)
 - -g (listet wie '-l' nur ohne Namen; '-G' ohne Gruppen)
 - -h (gibt die Größe der Dateien für Menschen leserlicher aus)
 - -m (gibt Liste durch Kommata getrennt)
 - -r (reversive Auflistung der Inhalte)
 - -R (rekursive Auflistung der Inhalte, auch Unterordner)
 - -s (größe der Dateien)
 - -l (Auflistung zeilenweise)
 - -sort=[var] (sortiert mit [var]=none (keine Sortierung); [var]=size (nach Größe sortiert); [var]=time (Sortierung nach Zeit); [var]=version (Sortierung nach Version), [var]=extension (Sortierung nach Anhang))

change directory

- Navigieren durch Dateisystem

Gruppenverwaltung

- groups [user] (zeigt Gruppen von [user] an)
- groupadd [group] (legt Gruppe an) [nur Root]
- groupdel [group] (löscht Gruppe) [nur Root]
- groupmod [option] [group] (modifizieren der Gruppen Eigenschaften) [nur Root]
- usermod -ag [user] [group] ([user] zu [group] hinzufügen) [nur Root]
- deluser [user] [group] ([user] aus [group] entfernen) [nur Root]

change mode

- Ändern von Zugriffsrechte von Ordnern und Dateien
- Syntax: chmod [mode] [path]
- Elemente von [mode]:
 - u= (Rechte des Besitzers)
 - g= (Rechte der zugeordneten Gruppe)
 - a= (Rechte aller anderer)
- Rechte:
 - r (Erlaubnis zum Lesen)
 - w (Erlaubnis zum Schreiben)
 - x (Erlaubnis zum Ausführen)

change owner

- Ändern des Besitzers und der Gruppe einer/s Datei/Ordners
- Benutzer ändern: `chown [user] [file]`
- Gruppe ändern: `chown [user]:[group] [file]`
- Parameter '-R' ändert Zugehörigkeit rekursiv

remove directory

- Löschen von leeren Verzeichnissen
- Vorsicht: auch leere Verzeichnisse können wichtig sein
- leere Ordner in dem Ordner werden auch gelöscht

echo

- Syntax: `echo [option] [text]`
- gibt eingegebenen Text wieder
- Ausnahme: `echo $c` (gibt Wert von Variable 'c' aus)
- Parameter:
 - `-n` (Ausgabe ohne neue Zeile)
 - `-e` (Verwendung von Formatierungsoptionen)

stdin, stdout, stderr

- jeder Prozess hat 3 Datenströme
- `stdin` (Standard Input [number: 0])
- `stdout` (Standard Output [number: 1])
- `stderr` (Standard Error [number: 2])

<, > und |

- Umleiten der Ein- und Ausgabe
- `<` Laden der Eingabe aus Datei
- Syntax von `<`: `[command] < [path]`
- `>` Schreiben der Ausgabe in Datei
- Syntax von `>`: `[command] > [path]`

- Pipe '|': `[command1] | [command2]`
- Ausgabe von `[command1]` wird zu der Eingabe von `[command2]`
- `stderr` umleiten:
 - `'2>'` es muss die Datenstromnummer angegeben werden
 - Bsp. umleiten von `stderr` auf `stdout`:
`[command] > [path] 2> &1'`

more

- Filereader
- kein Editieren möglich
- Betrachten immer von oben nach unten
- Navigation ist etwas hinderlich

less

- intelligenter Nachfolger von `more`
- unterstützt Scrolling
- Navigation:
 - Pfeil- und Bildlauf Tasten
 - `'/pattern'` durchsucht Dokument nach Muster
 - `'?pattern'` wie `'/pattern'` nur umgekehrt
 - `'&pattern'` zeigt nur Zeilen mit Muster
 - `'n'` spring zum nächsten Mustervorkommen
 - `'n'` ruft nächste Dokument auf

concatenate

- konkateniert und zeigt Inhalte von Datei an
- Syntax: `cat [option] [path]`
- Optionen:
 - `-n` (nummeriert ausgegebenen Zeilen)
 - `-s` (unterdrückt Ausgabe von wiederholenden Leerzeichen)

head und tail

- Textbetrachter auf bestimmte Bereiche der Datei
- **head** gibt ersten 10 Zeilen einer Datei aus
- **tail** gibt letzten 10 Zeilen einer Datei aus
- Parameter:
 - head [option] [path]
 - * -n [val] (gibt ersten [val] Zeilen aus)
 - tail [option] [path]
 - * -n [val] (gibt letzten [val] Zeilen aus)
 - * -f (gibt immer aktuelle Zeilen aus)

Sortieren und Suchen

- Syntax: sort [option] [path]
 - liefert Datei und sortiert Inhalt
 - -r (Ausgabe umdrehen)
 - -c (checkt ob Inhalt sortiert ist)
- Syntax: find [option] [path] [pattern]
 - sucht nach Dateien im System via Muster
 - -m (sucht Dateiname, nicht Pfad)

copy und move

- Syntax:
 - mv [path1] [path2]
 - cp [path1] [path2]
- Parameter:
 - -i (fragt vor Überschreiben)
 - -f (überschreibt **immer**)
 - -n (überschreibt **niemals**)
 - -backup (erstellt Backup vor Überschreiben)
 - -S, -suffix=[suf] (alternatives Suffix [suf] für Backup)
 - -r (verschieben von Ordnern [nur 'cp'!!!!])
 - -s (erstellt symbolische Links statt Dateien [nur 'cp'!!!!])

remove

- Löschen von Dateien und Ordnern
- Parameter:
 - -f (löschen ohne nachfragen)
 - -i (Nachfragen bei jeder Löschung)
 - -r (löscht rekursiv [für Ordner])
 - -d (leere Verzeichnisse löschen)
- 'rm' löscht nicht endgültig

tar

- Syntax entpacken: tar [option] [path]
- Syntax verpackern: tar [option] [path-archiv] [path-files]
- Parameter:
 - -c (erzeugt Archiv)
 - -x (extrahiert Dateien)
 - -v (Fortschritt auflisten)
 - -z (komprimieren als .gz, etc.)
 - -f (erzeugt beim entpacken einen Ordner mit Archivnamen)
 - -j (bzip2 format)
 - -r (Daten an Archiv anfügen)
 - -d (Unterschiede zwischen 2 Archiven anzeigen)

Prozesse verwalten

- top:
 - gehört zu Standardinstallation
 - -h (Befehle anzeigen)
- htop:
 - meist nicht Standardinstallation
 - leichtere Bedienung als 'top'
- Optionen für 'htop':
 - -d, -delay=[n] (Delay [n] zwischen Updates)
 - -C, -no-color (htop ohne Farbe)
 - -h (zeigt mögliche Optionen)

- p, -pid=[PID1],[PID2],.. (zeigt nur Prozesse mit gegebener PID)
- u, -user=[name] (zeigt nur Prozesse von Benutzer [name])

- Ansicht:

- Auslastung einzelner CPU-Kerne und Arbeitsspeicher
- Anteil von Swap und bisherige Systemlaufzeit
- PID (Prozess-ID), User, PRI (Priorität), NI (Nice-Wert)
- VIRT und RES (benutzer virtueller und physischer Speicher)
- S (State; S ∈ {S[leeping], R[unning], D[ist sleep], Z[ombie], T[race/suspend], W[Paging]})
- CPU% und MEM% (genutzer CPU- und RAM-Anteil)

nice

- starten von Prozessen mit modifizierter Priorität
- -20 ist höchste, 19 ist niedrigste Priorität
- Syntax: nice [option] [command] [args]
- Parameter:
 - n, -adjustment=N (aufaddieren von N auf Nice-Wert)

ping

- prüft die Verfügbarkeit einer Adresse
- Syntax: ping [option] [addr]
- -c [num] (Anzahl der Pings setzen)

renice und nohup

- Syntax: renice [-n] priority [-g | -p | -u] identifier
 - nachträglich den Nice-Wert eines Befehls einstellen
 - n [num] (neuer Nice-Wert)
 - g [group-id], -p [process-id], -u [user-id]

- Syntax: nohup [command] [args]
 - Programm läuft weiter auch nach Logout

&

- Angabe am Ende von Prozess um diesen in Hintergrund zu verschieben
- stdout ist immernoch das Terminal
- Prozess blockiert Shell nicht
- **Achtung!!** && ≡ Prozesse hintereinander ausführen
 - p1 && p2 : p2 ausgeführt, wenn p1 erfolgreich

Texteditoren

- ed:
 - nicht Teil der Standardinstallation
 - erster Texteditor; Release: 1970
 - a (append), i (insert), w (write), q (quit)
 - Grundlage für grep
- pico:
 - Release: 1992
 - sehr minimalistischer Editor
 - reine Textverarbeitung, mehrere Dokumente möglich, Syntaxhighlighting
 - kein Highlighting von einem Wort, kein Text-Splitscreen, keine RegEx Suche
- nano:
 - Release: 1999
 - sehr minimalistischer Editor
 - Nachfolger von pico
 - bietet mehr Funktionalität, z.B.: Text-Splitscreen, RegEx Suche
- vim:
 - Release: 1991
 - Großmufti unter den Editoren
 - verbesserte Suchfunktionen

- Standard bei 'git'

- jed:

- Release: 1992
- Editor für Programmierer
- liefert Templates für verschiedene Anwendungsfälle
- weniger mächtig als vim

shutdown

- Ausschalten oder Neustart
- Syntax: shutdown [options] [time]
- Parameter:
 - -P (poweroff)
 - -r (reboot)
 - -h (halt; kein Timer möglich)

ip addr

- Anzeigen von lokalen IP-Adressen
- Parameter:
 - -4 Anzeigen der IPv4-Adressen
 - -6 Anzeigen der IPv6-Adressen

nmon

- Shortcuts:
 - c (zeigt Auslastung aller CPU Kerne)
 - m (zeigt Auslastung Arbeitsspeicher [mit Swap])
 - d (zeigt Auslastung der Festplatte)
 - k (zeigt Kernel Statistiken)
 - r (Details zum System und Prozessor)
 - l (CPU Auslastung als Graph)
 - n (Auslastung der Netzwerk Schnittstellen)
 - j (Auslastung des Dateisystems)
 - t (wie top, alle laufenden Prozesse)
 - + bzw. - (Refresh erhöhen bzw. verringern)

- H (Hilfe)

RegEx

- zeigen Grundstruktur von gesuchtem Begriff
- Angeben immer in Anführungszeichen
- Syntax:
 - c (konstantes Zeichen c)
 - . (genau **ein** beliebiges Zeichen)
 - .* (Folge von beliebigen Zeichen [auch keines])
 - * (Zeichen vor * ist beliebig oft [auch gar nicht])
 - ? (Zeichen vor ? ist **genau** einmal oder gar nicht)
 - + (Zeichen vor + min. einmal)
 - {n,m} (Zeichen wird n bis m mal wiederholt)
 - {n} (Zeichen wird genau n mal wiederholt)
 - '[...]' (Zeichen kommt in den Klammern vor) [Hochkommata ignorieren]
 - '[...-...]' (Zeichen ist aus Zeichenklasse) [Hochkommata ignorieren]
 - '[...]' und '[...-...]' (verneinen von Zeichenklasse) [Hochkommata ignorieren]
 - \$ (Zeilenende)
 - < (Wortanfang)
 - > (Wortende)
 - a1|a2 (alternative Ausdrücke; a1 oder a2)
 - (...) Gruppe von Ausdrücken
 - \ Sonderbedeutung von nächstem Symbol ignorieren

grep

- durchsucht eine Eingabe nach einem RegEx
- Syntax: grep [path] [RegEx]
- Selektoren:
 - -E (Interpretiert Suchwort als erweiterten RegEx)
 - -F (Interpretiert Suchwort als fixed String)

- -G (Interpretiert Suchwort als basic RegEx)

- Kontrolle:

- -f [file] (liest RegEx aus [file])
- -v (invertiert das Matchmaking)
- -w (gibt nur Zeilen aus, in denen ganze Worte dem RegEx entsprechen)
- -x (gibt nur Zeilen aus, die ganz dem RegEx entsprechen)

- Ausgabe:

- -c (gibt Anzahl der passenden Zeilen aus)
- -o (gibt nur gematchten Teile einer Zeile aus)
- -q (keine Ausgabe, gibt Exitcode 0 aus, falls erfolgreich)

Alias

- Setzen von Alias für einen Befehl

- Vorgehen:

- Bearbeite `/.bashrc`
- füge `alias [aliasname]='[Befehl]'` hinzu
- speichern
- neu einloggen oder `source /.bashrc`

- Bsp. Zeile: `alias ll='ls -l'`