



ZUSAMMENFASSUNG - LINUX GRUNDLAGEN

«Software is like Sex, it's better when it's free.» ~Linus Torvalds

Kreienbühl Mika – Studierender Techniker HF System- &
Netzwerktechnik mit Cyber-Security

Mika Kreienbühl
mika.kreienbuehl@student.ipso.ch

Inhalt

Übersicht Zusammenfassung Linux Grundlagen	5
Fächerinfos	5
Informationen über Linux/Bash	6
Die Entstehungsgeschichte	6
Linux – Was ist das eigentlich?	6
GNU/Linux – Was ist der Unterschied zu Linux?	6
BASH – Die Shell unter Linux	6
Richard Stallman- Der Mann hinter der freien Software	6
GNU/Linux – Wie sieht die Architektur aus?	7
Linux Distributionen – Was ist das?	7
Server – Ganz allgemein	8
Servermotto	8
Labor - Debian Linux	9
Das Aufsetzen	9
Konfigurationen unter Linux	9
Case Sensitivity	9
Mirroring-Server	9
Probleme mit den Systemupdates	9
Source.list	9
Prozessübersicht	10
Hostname-Definition	11
Apt-Programm	11
Informationen über Programme	11
MBR	12
Backup	12
Benutzerverwaltung	12
Wer ist angemeldet?	12
Wie werde ich den Benutzer?	12
Wie wechselt man das Passwort?	12
Übersicht der User	12
Übersicht der Passwörter	12
Userangaben ändern	12
Übersicht der Gruppen	12
Übersicht der PWs der Gruppen	12
Anpassungen	13

Übersicht eines Users	13
User zu einer Gruppe adden	13
User löschen	13
Gruppenmanagement	13
Konsistenz prüfen	13
Berechtigungen	14
Change Permissions	14
Berechtigungszahlen	14
Change Grouppermissions	14
Change owner	14
Soft- & Hardlinks	15
Editoren	15
Dateien erstellen	15
Netzwerk	15
Portdefinition	15
Lokale Nameresolution	15
DNS-Definition	15
LDAP-Definition	15
Resolveorder-Definition	15
DNS-Konfiguration	15
Netzwerkkonfiguration auslesen	16
Netzwerksetting editieren	16
Default Gateway	16
Routing	16
IP Neighbours	16
ARP-Table	16
Aktive Internetverbindungen	16
Ping	16
Nslookup	16
who is	16
Systemadministration	16
Systemweite Konfigurationen	16
Zeitzone	17
Landeseinstellungen	17
Zeit setzen	17
Hwclock	17

Automatische Zeiteinstellungen mittels NTP	17
Logs.....	17
Befehle als Root ausführen	17
Sudo-Berechtigung	17
Skripts ausführen.....	18
Dateiendungen	18
Directoryverwaltung.....	18
VI(M).....	18
Grep.....	19
Sed.....	19
Jobautomatisierung.....	19
Anacron	19
Dateikomprimierung	20
Tar.....	20
Gzip.....	20
Bzip2	20
GNU/Linux – Filesystem Hierarchy Standard	20
Linux Shutdown.....	21
Funktionszustände von Linux	21
System-V vs. Systemd.....	22
Bootvorgang - Linux.....	22
Programm/Prozess-Management.....	22
Programm im Hintergrund/Vordergrund	23
Prioritäten der Prozesse	23
Systemadministration	24
Kernel-Module.....	24
Systemzustände.....	24
Systemumgebung.....	24
Sprache	24
Konsolen-Setup/Kodierung	24
Shared Libraries	24
X-Window (X11).....	25
Was ist ein X11?	25
Was ist ein Window-Manager?	25
Was ist ein Display-Manager?	25
Desktop-Umgebungen	25

Laufwerk-Management	25
Laufwerksnamen	25
Partitionen	25
Partitionierung	26
GPT-Limits	26
Dateisysteme	26
Partitionen formatieren	26
Partitionen mounten	26
Speicherplatz	26
Quotas	27
Quotas erstellen	27
Quotas starten	27
Quota Stats	27
Wichtige Konfig-Files und Pfade	28
Übersicht der Befehle	28
Bash Befehle	30
Befehl-Informationen	30
Gruppenarbeiten Bash Grundbefehle	30
Programm Ein- & Ausgabe	30
Aliase	30
Variablen	31
Rechnen mit Variablen	31
Sequenzielle Commands	31
Bash skripting	31
Aufbau	31
Variable definieren	31
Skript ausführbar machen	31
Skript ausführen	32
Variabelintegration	32
Array	32
Schleifen	32
Funktionen	33
Quellenverzeichnis	34

Übersicht Zusammenfassung Linux Grundlagen

Im Rahmen der Techniker HF an der IFA (ipso Bildung) werden im 1. Semester Linux Grundlagen erarbeitet, welche im Laufe der weiteren Weiterbildung wichtig für weitere Fächer wie Pen Testing unter Kali Linux sind.

Als Referenz dieser Zusammenfassung dienen die Unterrichtsmaterialien, welche im Rahmen der Techniker HF im Herbst/Winter Semester 2021/22 an der IFA ausgehändigt wurden. Ausserdem werden die Notizen sowie selbst erarbeitete Informationen verwendet.

Sämtliche Angaben in diesem Dokument sind ohne Gewähr und jegliche Haftung wird abgelehnt.

Das Copyright liegt alleinig bei Mika Kreienbühl, 12.11.2000 und darf ohne ein schriftliches Einverständnis weder kopiert noch editiert werden.

Fächerinfos

Schule	IFA, Bern
Lehrgang	Techniker HF System- & Netzwerktechnik mit Cyber-Security
Dozierender	Christian Kiss
Unterrichtszeitraum	11.2021 bis 05.2022

Informationen über Linux/Bash

Die Entstehungsgeschichte

Der Vater von Linux, Linus Torvalds, entwickelte im Jahre 1991 ein Kernel, welcher heute unter dem Namen «Linux» bekannt ist. Ursprünglich war die Intention dahinter, den Zugriff auf Unix-Server seiner Universität zu ermöglichen, ohne dabei jedes Mal in die Uni fahren zu müssen.

Linux – Was ist das eigentlich?

Linux in der Umgangssprache von heute, ist ein sehr hardwarenahes unix-ähnliches Mehrbenutzer-OS, welches auf dem Linux-Kernel basiert. Wobei der eingesetzte Begriff «Linux» z.T. irreführend ist, da mit Linux «nur» das Kernel gemeint ist. Viel mehr wäre der korrekte Begriff des umgangssprachlich genannten Betriebssystems [GNU/Linux](#).

GNU/Linux – Was ist der Unterschied zu Linux?

Unter Linux versteht man korrekterweise der Kernel auf welchem die unzähligen Linux Distributionen basieren. Der Kernel arbeitet äusserst hardwarenah und stellt Funktionen/Softwares wie Scheduling, Multitasking, Treiber, Speicherverwaltung usw. für die Distributionen zur Verfügung.

GNU-Pakete sind Softwares welche weitere Funktionen für das umgangssprachliche «Linux» zur Verfügungstellen. Dabei sind der Phantasie keine Grenzen gesetzt. Es sind das Shells, Coreutils, Compiler und weitere Software-Pakete.

Zusammen ergeben der Linux Kernel sowie die GNU-Pakete ein Unixartiges Betriebssystem. In verschiedener Kombination dieser GNU-Pakete entstehen die verschiedenen Distributionen wie Kali, Ubuntu uvm. Wobei diese alle für einen gewissen Einsatzzweck optimiert sind und weiter mit freier bzw. proprietärer Software ausgestattet sind.

BASH – Die Shell unter Linux

Die BASH (kurz für Bourne-again shell) ist eine Unix-Shell welche frei verfügbar ist und unter GPL (GNU Public Licence) betrieben wird. Es ist eine typische Mensch-Maschinen-Schnittstelle. BASH ist ein elementarer Bestandteil von GNU und ist meist auch standardmässig Teil von GNU/Linux basierten OS'. Interessant dabei ist, dass der Major-Releaser 3.X von BASH ebenfalls die Shell von Mac OS war (Jahre 2003-2019). Aus lizentechnischen Gründen wurde jedoch der Major-Release 4.X oder höher eingesetzt.

Richard Stallman- Der Mann hinter der freien Software

Richard Stallman ist DER Verfechter und Pionier von freier Software und somit von Open-Source.

GNU/Linux – Wie sieht die Architektur aus?

Die Architektur eines GNU/Linux Systems besteht im Wesentlichen aus drei Bestandteilen. Als Basis dient, wie bei jedem System, die Hardware. Die Hardware wird durch den Linux-Kernel bzw. durch seine Treiber verwaltet und angesteuert.

Über dem Kernel arbeiten dann die oben genannten GNU-Pakete.

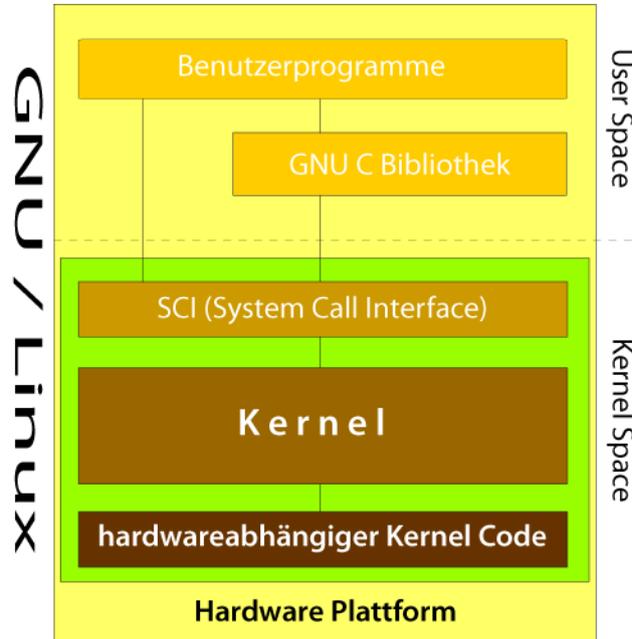


Abbildung 1 GNU/Linux Architektur

Linux Distributionen – Was ist das?

Linux Distributionen sind einfach gesagt, Einsatzzweck bezogene Betriebssysteme welche für einen gewissen Einsatzzweck optimiert sind. Diese sind mit freier- bzw. mit proprietärer Software ausgestattet und erfüllen dadurch standardmässig einen bestimmten Einsatzzweck.

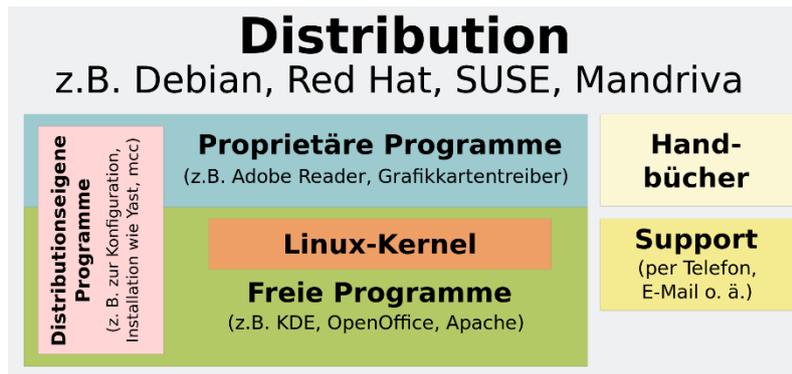


Abbildung 2 Linux Distribution - Der Aufbau

Es gibt weit über 700 Linux Distributionen. Eine, nicht zwingendermassen abschliessende Übersicht dieser Distributionen findet man hier:

[Linux Distribution Tree](#)

Server – Ganz allgemein

Servermotto

Das Servermotto ist ein ungeschriebenes Gesetz der System-Administratoren welche aussagt, dass nur die Tools installiert werden sollen, welche im Betrieb bzw. der Administration auch benötigt werden.

Labor - Debian Linux

Das Aufsetzen

Das Aufsetzen von Linux ist in der Regel selbsterklärend und trivial. Wobei das Aufsetzen nach dem Server-Motto «Installiere nur das, was du benötigst» geschieht. Heisst die standardmässig mitinstallierenden Tools werden über das manuelle Aufsetzen deaktiviert.

Jedes Tool kann anschliessend nachinstalliert werden, so hat der Administrator die Übersicht, was und wie es installiert wurde.

Beim Aufsetzen sollte darauf geachtet werden, dass ein UTF-8 Zeichenformat gewählt wird. Im Gegensatz zu ISO, beinhaltet UTF-8 sämtliche Sonderbuchstaben wie z.B. è,é,à usw.

Konfigurationen unter Linux

Konfigurationen unter Linux werden alle in einem dafür vorgesehenen Konfigurationsfile definiert. Mittels dem Pfad, sofern man diesen weiss, kann man jede Konfiguration abrufen.

Case Sensitivity

Linux ist case sensitive, heisst also die Datei /etc/hostname ist nicht dieselbe wie /etc/Hostname. Zu Deutsch bedeutet das, dass zwischen Klein- und Grossschreibung unterschieden wird. Dies kann zu Problemen führen, vor Allem für Benutzer, welche sich an Windows gewöhnt sind.

Mirroring-Server

Der Mirroring-Server ist einfach gesagt die Referenz bzw. der Ort wo das System seine Updates herbekommt. Verschiedene Organisationen auf der ganzen Welt stellen einen solchen zur Verfügung. In der Schweiz z.B. die ETH.

Probleme mit den Systemupdates

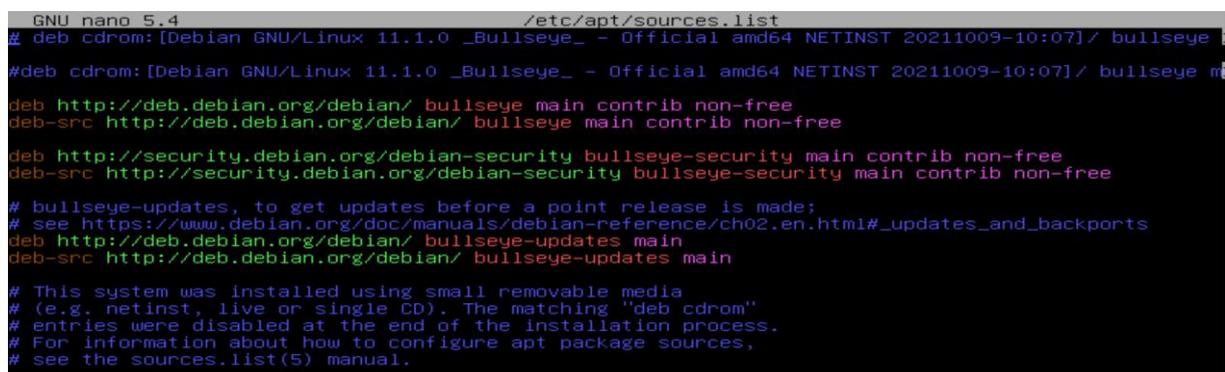
Mögliche Probleme mit Systemupdates könnten auf den Mirroring-Server zurückzuführen sein.

Source.list

Die Source.list ist eine sehr wichtige Liste in der Systemadministration. Sie verwaltet, auf welche Source-Server für Installationen sowie Updates das System zugreift.

Diese findet man unter folgendem Pfad:

/etc/apt/sources.list



```
GNU nano 5.4 /etc/apt/sources.list
# deb cdrom:[Debian GNU/Linux 11.1.0 _Bullseye_ - Official amd64 NETINST 20211009-10:07]/ bullseye
#deb cdrom:[Debian GNU/Linux 11.1.0 _Bullseye_ - Official amd64 NETINST 20211009-10:07]/ bullseye m
deb http://deb.debian.org/debian/ bullseye main contrib non-free
deb-src http://deb.debian.org/debian/ bullseye main contrib non-free
deb http://security.debian.org/debian-security bullseye-security main contrib non-free
deb-src http://security.debian.org/debian-security bullseye-security main contrib non-free
# bullseye-updates, to get updates before a point release is made;
# see https://www.debian.org/doc/manuals/debian-reference/ch02.en.html#_updates_and_backports
deb http://deb.debian.org/debian/ bullseye-updates main
deb-src http://deb.debian.org/debian/ bullseye-updates main
# This system was installed using small removable media
# (e.g. netinst, live or single CD). The matching "deb cdrom"
# entries were disabled at the end of the installation process.
# For information about how to configure apt package sources,
# see the sources.list(5) manual.
```

Abbildung 3 sources.list

Wie oben ersichtlich, besteht die sources.list aus URLs sowie zugehörige Parameter. Wobei der Parameter «bullseye main contrib non-free» hinter den jeweiligen Sources definiert, dass ebenfalls nicht freie Software abgerufen/installiert werden kann.

Prozessübersicht

Um die Prozesse sehen zu können, kann man mit dem Befehl ps die Prozesse übersehen.

Mit weiteren Parametern kann man auch dann noch die Übersicht erweitern und tunen:

A -> wählt alle Prozesse aus, identisch mit -e

N -> wählt alle Prozesse aus, die die angegebenen Bedingungen nicht erfüllen (negiert die Auswahl)

T -> wählt alle Prozesse die mit dem aktuellen Terminal verbunden sind aus. Identisch der t Option, ohne jedes Argument

-a -> wählt alle Prozesse, außer den Session-Leitern und Prozessen, die nicht mit einem Terminal verbunden sind, aus

a -> hebt die BSD-Stil "nur du selbst" Beschränkung auf, die durch die Verwendung von BSD-Optionen auf die Auswahl der Prozesse angewendet wird. Diese Option listet also alle Prozesse eines Terminals

-d -> wählt alle Prozesse außer Session-Leitern aus

-e -> wählt alle Prozesse aus, identisch mit -A

r -> beschränkt die Auswahl auf laufende Prozesse

-C PROZESSLISTE -> zeigt nur die Prozesse an, die in PROZESSLISTE aufgeführt sind.

x -> hebt die BSD-Stil "muss ein tty haben" Beschränkung auf, die durch die Verwendung von BSD-Optionen angewendet wird. Diese Option listet also alle Prozesse, die dem aufrufenden Benutzer gehören

p -> Zeigt die Prozesse inkl. Deren Relationen an

u -> Zeigt die Prozesse einer Users an

pstree zeigt die Prozesshierarchie an

top zeigt Realtime-Infos an wie der Taskmanager unter Windows.

Hostname-Definition

Der Hostname wird ebenfalls über ein Konfigurationsfile definiert. Dies befindet sich unter folgendem Pfad:

`/etc/hostname`

Wobei der Hostname mittels «`more /etc/hostname`» bzw. `hostname` abgerufen werden kann.

```
root@ifadebian:~# hostname
ifadebian
root@ifadebian:~# more /etc/hostname
ifadebian
root@ifadebian:~# _
```



Abbildung 4 `hostname`

Apt-Programm

Das apt-Programm umfasst viele Möglichkeiten für die Verwaltung von Software.

Wobei der bekannteste der «`apt-install`»-Befehl ist, für die Installation von Softwarepaketen. Für das Entfernen von Installationen würde der «`apt-remove`»-Befehl verwendet.

Der Update-Befehl aktualisiert die Updateliste von Mirroring Server auf die lokale Liste.

Wobei der Upgrade-Befehl die verfügbaren Updates dann installiert. Für das korrekte Updaten der Softwares, müssen beide Befehle nacheinander ausgeführt werden, da einer allein keine aktuellen Updates installieren kann.

Weiter gibt es den «`Apt install dist-upgrade`»-Befehl, dieser wird dazu verwendet die Distribution und nicht nur die installierte Software zu aktualisieren.

Der Befehl «`apt purge`» Bereinigt ebenfalls die Einstellungen einer Installation und nicht nur die Installation wie dies bei `remove` der Fall ist.

Mittels apt kann ebenfalls der Mirroring Server durchsucht werden, wobei dies über folgende Befehle geschieht:

`apt list php` -> Zeigt alle Pakete an, welche mit php zutun haben

`apt search php` -> Sucht explizit Pakete an, welche php enthalten.

Da nach Jahren von Laufzeit sich verschiedene Softwarepakete und Download-Archives ansammeln können diese auch mit dem apt-Programm bereinigt werden. Dafür verwendet man folgende Befehle:

`apt autoremove` -> Bereinigt nicht mehr verwendete Installationen

`apt autoclean` -> Bereinigt alte Download archives

Informationen über Programme

Das `man`-Programm zeigt mit folgendem Befehl; `man <program>` Informationen über ein Programm an. Weiter können mit dem Befehl `apt --help <program>` weniger detaillierte Informationen angezeigt werden.

MBR

Backup

Da der MBR für die korrekte Funktionalität des Systems essentiell ist, bietet es sich an, diesen zu backupen. Mittels des Befehls «dd if=/dev/sda of=/dev/sdb/mbr.backup kann das Backup erstellt werden.

Benutzerverwaltung

Die Benutzerverwaltung unter Linux umfasst die Bedienung aber auch die Benutzeradministration

Wer ist angemeldet?

```
root@ifadebian:~#
```

root -> angemeldeter Benutzer

ifadebian -> Maschine

~ -> Location (~ heisst das Userverzeichnis bei einem User /home/user beim root /root

-> Benutzerrechte (# für rootkontext, \$ für userkontext)

Wie wechselt man den Benutzer?

Der Benutzer kann mit folgendem einfachem Befehl gewechselt werden:

```
su -user
```

Wobei user für den Benutzernamen steht.

Gibt man keinen Parameter nach su an, so wechselt das System zum root.

Wie wechselt man das Passwort?

Das Passwort eines Benutzers kann mit folgendem Befehl geändert werden:

```
passwd -> Das Passwort wird für den aktuellen Benutzer geändert
```

```
passwd user -> Das Passwort wird für einen definierten user geändert
```

Übersicht der User

```
etc /passwd
```

```
user:x:1000:1000:user,,,:/home/user:/bin/bash
```

```
username:pw:uid:gid:gecko:userhome:std.Shell
```

```
finger <user> (benötigt zusätzliche Installation)
```

Übersicht der Passwörter

```
/etc/shadow
```

```
user:pw-hash:timestamp:mind. PW-Days:max. PW-Days:alert
```

Userangaben ändern

```
chfn user
```

```
chage -l user
```

```
chage user
```

Übersicht der Gruppen

```
/etc/group
```

Übersicht der PWs der Gruppen

```
/etc/gshadow
```

Anpassungen

/etc/adduser.conf

/etc/default/useradd.conf

Übersicht eines Users

Id <user>

User zu einer Gruppe adden

Usermod -aG<group> <user>

User löschen

Userdel <user> -rf

Gruppenmanagement

groupadd <group> -g xxxx -> Gruppe erstellen wobei xxxx durch die Group-ID ersetzt wird.

groupmod <group> -n <newname>

gpasswd <group> -> Neues Gruppenpasswort

groupdel <group>

Konsistenz prüfen

pwck

grpck

Berechtigungen

-rwxrwxrwx				
-	rwx	rwx	rwx	S
L = Link D = Directory - =File	User	Group	Other	SUID-Bit

R = Read

W = Write

X = Execute

Das SUID-Bit definiert, dass die Ausführung des Files immer als Root gemacht wird und hat den Stellenwert 4.

Also: `chmod 4777`

Ist es bei einem Ordner gesetzt, so gilt die Definition als Vererbung aktiviert.

Das Sticky-Bit schützt vor versehentlichem Löschen bei others. Nur Root und der Owner können das File löschen.

Change Permissions

`chmod +rwx <file>`

`chmod -rwx <file>`

`chmod g=rw <file>` (Nur die angegebenen Rechte zählen)

`chmod u+x <file>`

`chmod u=rw,g-w,o+r <file>`

Berechtigungs zahlen

Zahl	Berechtigung
0	None
1	X
2	W
3	WX
4	R
5	RX
6	RW
7	RWX

Change Grouppermissions

`chgrp <grp> <file>`

Change owner

`chown <user> <file>`

`chown <user>:<group> <file>`

Soft- & Hardlinks

Softlinks verweisen auf einen anderen Ort bzw. eine Inoden Nummer über dem verlinkten Dokument. Hardlinks hingegen verweisen auf die Inodenummer des eigentlichen Files. Wird das original Dokument gelöscht, so geht der Softlink nicht mehr, der Hardlink jedoch schon.

Editoren

Unter Linux gibt es mehrere Editoren wie man dies unter Windows z.B. mit notepad/editor kennt. Folgende Editoren gibt es dabei: vi, vim, nano usw.

Das Management in den Editoren ist von Editor von Editor unterschiedlich. In der Regel ist die Wahl mehr eine persönliche/unternehmerische und nicht eine funktionelle Entscheidung.

Dateien erstellen

Dateien erstellen unter Linux ist sehr trivial mit folgendem Befehl:

`touch /x/x/x` wobei die Datei unter dem angegebenen Pfad erstellt wird, sofern diese noch nicht existiert. Wird kein Pfad angegeben so wird die Datei im aktuelle Directory erstellt.

Wichtig zu wissen ist ebenfalls, dass der Touch-Befehl eigentlich nur zum «Berühren» der Datei vorhanden ist, erstellt diese jedoch bei nicht vorhanden sein. Ist die Datei vorhanden, so aktualisiert er den Zeitstempel in den Metadaten der Datei.

Netzwerk

Portdefinition

`/etc/services`

Lokale Nameresolution

`/etc/hosts`

DNS-Definition

`/etc/resolv.conf`

LDAP-Definition

`/etc/nsswitch.conf`

Resolveorder-Definition

`/etc/host.conf`

DNS-Konfiguration

Die DNS-Konfiguration befindet sich ebenfalls in einem dafür vorgesehenen Konfigurationsfile. Dieses befindet sich unter `/etc/resolv.conf` und muss mit root-Rechten sowie einem Editor bearbeitet werden. Z.B: `sudo vim /etc/resolv.conf`



```
nameserver 8.8.8.8
~
~
~
~
~
~
~
```

Abbildung 5 `/etc/resolv.conf`

Netzwerkconfiguration auslesen

Um die Netzwerkconfiguration auslesen zu können gibt es grundsätzlich mehrere Wege, wobei bei einigen zuvor noch Tools installiert werden müssen.

Der Befehl «ip a» funktioniert jedoch standardmässig bei allen Linux Distributionen

Will z.B. mittels des sehr prominenten Befehls «ifconfig» die Netzwerkconfiguration aufgelesen werden, so müssen zuvor die nettools installiert werden.

Netzwerksetting editieren

```
ifconfig <int> down
```

```
ifconfig <int> <ip addr> netmask <netmask>
```

```
ip addr add <ip addr>/<CIDR> dev <int>
```

```
ip link set <int> up
```

Default Gateway

```
route -n
```

```
route add default gw <gw>
```

Routing

```
route add -net <net> netmask <netmask> gw <gw>
```

```
route add -host <host> gateway <gw>
```

IP Neighbours

```
Ip neighbour show
```

ARP-Table

```
arp -v
```

Aktive Internetverbindungen

```
netstat (-tlpn)
```

Ping

```
ping -c 3 google.com
```

Linux macht ohne den C-Parameter ICMP-Abfragen bis diese manuell abgebrochen werden.

Nslookup

```
host -t <type> ifa.ch
```

```
dig -t a ifa.ch
```

who is

who is fragt ab, wer hinter einer Domäne bzw. ab einer gewissen Anzahl IPs pro Firma kann auch gecheckt werden, wer hinter der IP steckt.

Systemadministration

Systemweite Konfigurationen

Mit dem Befehl dpkg-reconfigure können Konfigurationen geändert werden welche Systemweit und z.T. während des Aufsetz-Vorganges definiert wurden oder einfach Standardmässig gesetzt wurden.

z.B. kann mittels dpkg-reconfigure dash kann die Standard-Shell von bash auf dash geändert werden
Weiter kann z.B. mit dpkg-reconfigure tzdata die Zeitzone geändert werden

Zeitzone

tzselect

dpkg-reconfigure tzdata

/etc/timezone

/etc/localtime -> /usr/share/zoneinfo/Europe/Zurich

Landeseinstellungen

locale (-a) -> (Zeigt alle installierten)

Zeit setzen

date -s 1031

date -s 20220214

-> Diese Settings könnten vom Hypervisor überschrieben werden

Hwclock

BIOS Zeit

sudo hwclock --systohc/--hctosys

Automatische Zeiteinstellungen mittels NTP

Linux unterstützt ebenfalls das NTP (Network Time Protocol), benötigt jedoch eine zusätzliche Installation.

Mit dem bereits bekannten Befehl «apt install» kann das Programm installiert werden.

Wobei der Service anschliessend allenfalls noch gestartet werden muss.

Mit dem Befehl `systemctl status ntp` kann der Status des Service erfragt werden. Gestartet bzw. gestoppt wird ein Service mit «`systemctl start ntp`» bzw. `systemctl stop ntp`».

/etc/ntp.conf

ntpq -p -> Abfrage von NTPs

Logs

/var/log

Kern.log -> Kernel-Log

Syslog -> SW/System-Log

Mittels `tail -f` werden die letzten 10 Einträge angezeigt

Logrotate erstellt in gewissen Abständen neue Logfiles und löscht alte nach einer gewissen Zeit.

Befehle als Root ausführen

Befehle können von einem normalen User als root ausgeführt werden ohne, dass der Benutzer gewechselt werden muss.

Der Befehl ist dabei «`sudo <befehl>`».

Sudo-Berechtigung

Wichtig ist, dass der ausführende Benutzer einmalig für sudo berechtigt werden muss. Dies geschieht folgendermassen:

`usermod -aG sudo user` oder aber mittels `adduser user sudo` (ist zweckentfremdet, funktioniert jedoch ebenfalls)

Grep

Grep ist ein Befehl, welcher das Durchsuchen von Dateien ermöglicht. Die wichtigsten Parameter und Suchmöglichkeiten sind nachfolgend aufgelistet.

Grep 300[1-5] gemeinde.txt -> Sucht im gemeinde.txt alle Einträge mit 300 zu Beginn und 1-5 am Schluss also 3001,3002,3003,3004 und 3005.

Grep ^[A-C] gemeinde.txt -> Sucht alle Einträge mit A, B oder C am Anfang.

Grep 09\$ gemeinde.txt -> Sucht alle Einträge mit 09 am Ende.

Grep \<<Bern> gemeinde.txt -> Sucht alle Einträge mit «Bern» exkl. Berneck o.Ä.

Grep «St\.» gemeinde.txt -> Sucht alle Einträge mit St.

Mit dem Backslash wird definiert, dass es sich um ein Zeichen und nicht ein Aufruf einer Funktion, Variable oder Platzhalter handelt.

Grep -i -> Case Sensitivity off

. -> Platzhalter mit dem Wert 1

* -> Platzhalter mit undefiniertem Wert

? -> Platzhalter mit dem Wert 1 für den Einsatz im Dateisystem.

Sed

Mittels des Befehles «sed» wird das Suchen und Ersetzen von Wörtern ermöglicht.

Sed s/Solothurn /Basel /g sedtext.txt -> Ersetzt Solothurn mit Basel

Sed '/^#/d' sedtext.txt -> Löscht alle # am Anfang der Zeilen. Wobei ^ definiert, dass nur am Anfang von Zeilen gesucht werden soll.

Zu beachten ist, dass sed ein Streamingeditor ist, wobei nur die Ausgabe editiert wird, nicht jedoch die angesprochene Datei.

Jobautomatisierung

Jobs können unter Linux automatisiert werden dies mit Hilfe von crontab.

Man findet es unter vim /etc/crontab bzw. crontab -e bzw. /var/spool/cron/crontabs.

Anacron

Skripts, welche durch Stromausfall nicht ausgeführt wurden, werden so nicht nachgeholt. Für diesen Fall gibt es anacron, (/etc/anacron oder /var/spool/anacron/crondaily).

Damit definiert wird, wer Jobs für anacron aufgeben darf, gibt es at.allow/at.deny und cron.allow/cron.deny.

Wird jemand in allow definiert, können alle anderen nicht. Wird jemand in deny definiert, können alle anderen.

Dateikomprimierung

Dateikomprimierung war früher, bei dem Mangel an Speicherplatz deutlich wichtiger, jedoch auch heute noch nicht zu vernachlässigen. Wobei mehrere Programme unter Linux eine Komprimierung ermöglichen.

Tar

Mit dem Tar dem Tape archiver wird eine Komprimierung unter Linux erreicht, welche früher für Tapes eingesetzt wurde.

Parameter:

- czvf archive.txt.gz archive.txt-> unzipping der Datei
- c -> create > etwas packen
- z -> compresses the file
- v -> Verbose-Mode (Mehr Infos)
- f -> file, wobei immer am Ende
- x -> unpack

Gzip

Liefert eine gute Komprimierung.

gzip archive.txt -> Komprimiert zu archive.txt.gz
mit dem Paramet -k wird die Sourcdatei nicht gelöscht.
mit gunzip wird die Datei entpackt

Bzip2

Liefert eine bessere Komprimierung.

Bzip2 archive.txt -> Komprimiert zu archive.txt.gz
mit dem Paramet -k wird die Sourcdatei nicht gelöscht.
mit bunzip wird die Datei entpackt

GNU/Linux – Filesystem Hierarchy Standard

GNU/Linux hat eine Standard Dateisystem Hierarchie. Dort wird definiert, welche Dateien wo und wie abgelegt werden. Nachfolgend die Übersicht der Standardhierarchie:

- home -> Alle Userhomes
- mnt -> Mountpoints für USB-Sticks usw.
- media -> Mountpoint für CDs (Dies unterscheidet sich bei gewissen Distributionen)
- etc -> Alle Konfigurationen
- bin -> Applikationen für alle User
- sbin -> Applikationen welche nur der Admin sieht
- dev -> Alle Geräte welche wir ansteuern können
- boot -> Alle, zum booten benötigten, Dateien
- lib -> Libraries von den Programmen
- opt -> Optionale Programme z.B: von GitHub
- proc -> Informationen für Einstellungen
- root -> home des root-Benutzers
- tmp -> Temp Dateien
- usr -> Alle Unix-Ressourcen
- var -> Variable Dateien wie z.B. log (Ändern sich regelmässig)

Linux Shutdown

Unter Linux gibt es verschiedene Möglichkeiten das System Herunterzufahren oder Neuzustarten. Folgende Befehle werden am meisten verwendet.

- Neustarten: `Shutdown -r -f -t 0` (wobei r für Reboot, f für Force sowie t für Zeit in Sekunden steht)
- Neustarten: `init 6` (wobei der 6. [Funktionszustand](#) initiiert wird.)
- Neustarten: `reboot`
- Herunterfahren: `init 0` (wobei der 0. [Funktionszustand](#) initiiert wird.)
- Herunterfahren: `poweroff`
- CPU anhalten: `halt` (Die Maschine läuft, die Prozesse werden jedoch gekillt.)

Sollte es unter Linux zu einem kompletten Systemcrash kommen so kann man mittels folgender Tastenkombination das System komplett neustarten:

`alt + prt + REISUB`

Die Buchstaben werden in diesem Fall jeweils nach einander gedrückt und bedeuten folgendes:

- R -> Die Tastatur wird vom Raw-Mode in den XLATE Mode geschaltet
- E -> Das SIGTERM Signal wird an alle Prozesse geleitet exkl. init
- I -> Das SIGKILL Signal wird an alle Prozesse geleitet exkl. init
- S -> Synchronisiert alle gemounteten Dateisysteme
- U -> Remountet alle Dateisystem im Read-Only Mode
- B -> Rebootet das System sofort ohne unmounting oder syncing

Funktionszustände von Linux

Der Funktionszustand von Linux kann 7 Zustände haben:

- 0 -> Halt
- 1 -> Single User Mode
- 2 -> Local Multiuser Mode mit Remote Netzwerk
- 3 -> Full Multiuser Mode mit Netzwerk
- 4 -> Custom
- 5 -> Full Multiuser Mode mit Netzwerk & GUI
- 6 -> Reboot

Die Funktionszustände greifen unter `systemd` auf sogenannte Targets zu, welche unter `/lib/systemd/system` liegen zu. In diesen ist das Verhalten der einzelnen Targets definiert. Grundsätzlich heisst das also, dass für jeden Funktionszustand ein `runlevel.target` definiert wurde. Z.B. für `init 5` -> `runlevel5.target`.

Das runlevel kann über den Befehl «`runlevel`» abgefragt und über «`init X`» geändert werden wobei X für den Funktionszustand steht.

Wechselt man den Funktionszustand und fragt anschliessend das runlevel ab, so stehen 2 Zahlen da:

`#runlevel`

`5 3`

Wobei die erstgenannte Zahl den vorherigen Funktionszustand beschreibt.

Um den Standard abzufragen gibt man «`systemctl get-default`» ein.

Um den Standard zu definieren gibt man «`systemctl set-default multiuser.target`» ein

System-V vs. Systemd

Früher wurden die System-V Befehle verwendet unter Linux. Heute werden die Systemd Befehle verwendet, wobei die eigentlichen Befehle von System-V nach wie vor funktionieren, jedoch im Hintergrund in Systemd übersetzt werden und somit anders funktionieren.

Dies wurde so implementiert, um alt-ingesessenen System Administratoren die Arbeit zu erleichtern.

Bootvorgang - Linux

Der Bootvorgang eines Systems besteht grob gesagt aus 7 Schritten unter Linux sieht das folgendermassen aus:

1. Im ersten Schritt wird das BIOS/UEFI gestartet
2. Weiter wird vom BIOS/UEFI anschliessen der MBR initialisiert
3. Im dritten Schritt kann anschliessend der Bootloader (Unter Linux GRUB) starten
4. Dieser initiiert im 4. Schritt die ramdisk (Temporäres Dateisystem auf dem RAM)
5. Von da aus kann anschliessend der Kernel gestartet werden
6. Wo anschliessend der Init-Prozess/Systemd gestartet wird
7. Im 7. Und letzten Schritt werden dann die benötigten Dienste des Systems gestartet.

Programm/Prozess-Management

Um ein Programm im Hintergrund zu starten fügt man dem Programm noch «&» hinzu.

Beispiel: «vim &»

Um ein Prozess zu killen gibt man folgendes ein:

kill -9 <pid>

Wobei es mehrere Killvarianten gibt:

Num ber	Name (short name)	Descri ption	Used for
0	SIGNULL (NULL)	Null	Check access to pid
1	SIGHUP (HUP)	Hangup	Terminate; can be trapped
2	SIGINT (INT)	Interrup t	Terminate; can be trapped
3	SIGQUIT (QUIT)	Quit	Terminate with core dump; can be trapped
9	SIGKILL (KILL)	Kill	Forced termination; cannot be trapped
15	SIGTERM (TERM)	Termin ate	Terminate; can be trapped
24	SIGSTOP (STOP)	Stop	Pause the process; cannot be trapped. This is default if signal not provided to kill command.
25	SIGTSTP (STP)	Termin al	Stop/pause the process; can be trapped
26	SIGCONT (CONT)	Continu e	Run a stopped process

Abbildung 7 Kill-Codes

Um einen User komplett zu killen braucht man «kill -p user»

Killall -s 9 vim -> Killt alles was mit VIM zutun hat.

jobs -> Übersicht des im Hintergrund laufenden Prozesse

Programm im Hintergrund/Vordergrund

bg <Nr. unter Jobs>

fg <Nr. unter Jobs>

Prioritäten der Prozesse

Linux Prozesse

Prioritäten beim Start mitgeben mit nice

Prozess-Prioritäten:

Priorität von (+19 bis -20)

Ein normaler Benutzer kann die Priorität nur niedriger setzen. Der Root hingegen kann Prioritäten auch erhöhen.

Prozess-Prioritäten mit nice reduzieren:

nice -n 10 vi oder nice -10 vi (niedrigere Prio)

Wichtig der Strich bei -10 bedeutet nur die Option nicht minus!!

Prozess-Prioritäten mit nice erhöhen (als root):

nice -n -10 vi oder nice --10 vi (höhere Prio)

Systemadministration

Kernel-Module

Der Linux-Kernel ist ein monolither Kernel, was bedeutet, dass gewisse Module/Treiber bereits mitgeladen werden, andere wiederum müssen noch nachgeladen werden.

Die Module haben den Dateityp `.ko` und die nachgeladenen Module sind im Verzeichnis `/lib/modules` zu finden.

Systemzustände

Um Informationen über das System zu erhalten, gibt es folgende Quellen:

`/proc/sys/kernel`

`more ostype`

`more osrelease`

`more hostname`

Um einen Überblick über die geladenen Module zu erhalten kann man mit `«lsmod»` oder `«more /proc/modules»` diese auflisten

Der command `«modinfo»` liefert Informationen über die Module und muss als root gestartet werden.

Um ein Modul zu stoppen kann der Befehl `«rmmod -f <module>»` verwendet werden.

Ein Module kann mit `«insmod <absolut module-path>` gestartet werden. Wichtig dabei der absolute Pfad(!)

Ein Module hinzuzufügen/entfernen kann man mit folgenden Befehlen:

`modprobe -v bluetooth` bzw. `modprobe -r -v bluetooth`

Um eine Abhängigkeit zwischen Modulen zu erstellen benötigt man den Befehl `«depmod»`.

Die vorhandenen Abhängigkeiten kann man mit `«more /lib/modules/<kernel-version>/modules.dep | grep «blue»:»` abrufen

Systemumgebung

Sprache

Aktive Sprache

`localectl`

verfügbare Sprachen

`localectl list-locales`

Sprachen installieren

`dpkg-reconfigure locales`

Konsolen-Setup/Kodierung

`dpkg-reconfigure console-setup`

Shared Libraries

Es gibt sogenannte Shared Libraries welche Programmfunktionen zur Verfügung stellen, welche von mehreren Programmen verwendet werden kann. So kann man beim Programmieren Zeit sparen indem man bereitsbestehendes verwendet.

Welche Shares Libraries verwendet werden kann man mit `«which less»` oder `«which programm»` herausfinden.

X-Window (X11)

Was ist ein X11?

X11 ist ein Netzwerkprotokoll sowie eine Software welche Bitmap-Displays unter Unixoiden-Systemen erstellt. Grundsätzlich ist es ein Baukasten für den Bau einer grafischen Benutzeroberfläche. Es steuert Ein- sowie Ausgabegeräte wie Maus, Tastatur und Grafikkarte.

Was ist ein Window-Manager?

Ein Window Manager verwaltet die Funktionen wie Minimieren, Vergrössern und Schliessen von Fenstern. Oft wird durch den Window-Manager Titelleiste & Rahmen der Fenster bestimmt.

Was ist ein Display-Manager?

Er zeigt den grafischen Anmeldebildschirm. Ausserdem werden damit Einstellungen getätigt über zu verwendende Desktop-Umgebung oder Fenster-Manager

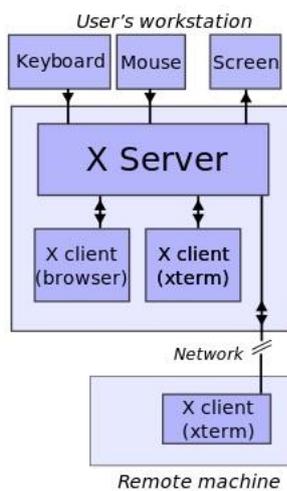


Abbildung 8 Display-Manager

Desktop-Umgebungen

Die Präsentationen zu den Desktop-Umgebungen findet man [hier](#).

Laufwerk-Management

Laufwerksnamen

Laufwerke haben unter Linux eine spezielle Syntax je nach Typ und Reihenfolge. IDE-Geräte heissen hda, hdb, hdc usw. wobei a,b,c für die Nummerierung steht. SATA-Geräte werde sda,sdb,sdc usw. benannt.

Zu finden sind diese im Directory /dev/xxx.

Partitionen

Die technische Limitierung vom MBR erlaubt nur 4 Partitionen. Durch erweiterte Partitionen kann diese Limitierung umgangen werden. Es können neue logische Partitionen erstellt werden:

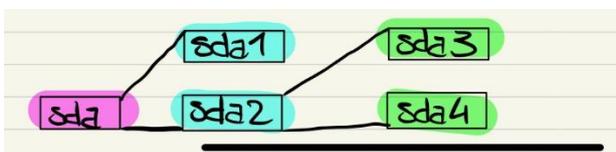


Abbildung 9 Partitionierung

Eine extended Partition ist nur der Container für weitere Partitionen. Dieser muss mind. So grosse Sein wie die Partition selbst darin.

Um alle Laufwerke anzuzeigen braucht man den Befehl «lsblk».
Die Partitionen können mit «more /proc/partitions» angezeigt werden.
Eine weitere Möglichkeit ist, mit «ls /dev/sd*».

Eine erweiterte Übersicht bekommt man mit «parted -l» sowie «fdisk -l» wobei bei fdisk nicht ersichtlich ist, ob es sich um eine primäre oder logische Partition handelt.

Partitionierung

fdisk /dev/sdb -> Programm für sdb aktivieren

m -> Hilfe

l = Übersicht der Dateisysteme

GPT-Limits

Max. 128 Partitionen

Max. 4 Exabytes

Dateisysteme

Die Präsentationen zu Dateisystemen findet man [hier](#).

Partitionen formatieren

Mkfs.ext4 /dev/sdb

Gewisse Dateisysteme sind standardmässig verfügbar, andere müssen bei Gebrauch noch installiert werden.

Partitionen mounten

mount <device> <mount-ort>

Um den Mount persistent zu machen, muss dieser in /etc/fstab eingepflegt werden

Um die Mountings zu prüfen, einfach «mount» eingeben.

Speicherplatz

Um den freien Speicher zu prüfen «df -h» wobei -h für humanreadable steht.

Um den verwendeten Speicher zu prüfen «du -sch <pfad>» wobei s für den Pfad, c für Zusammen rechnen und h für humanreadable steht.

Quotas

Quotas sind Limitierungen, welche den Speicher auf Benutzerebene limitiert.

Es gibt ein Hardlimit, welches das absolute Limit ist und auch nicht überschritten werden kann. Das Softlimit hingegen kann für eine gewisse Zeit überschritten werden und z.B. nach 5 Tagen wird dann das weitere Schreiben blockiert.

Quotas erstellen

```
sudo edquota -u user
```

Quotas starten

```
sudo quota on
```

Quota Stats

```
repquota /mnt/daten1
```

Weitere Informationen und Tutorials [hier](#).

Wichtige Konfig-Files und Pfade

Sources.list -> /etc/apt/sources.list

Benutzerhome -> /home/user, /root bzw. ~

DNS-Konfiguration -> /etc/resolv.conf

Target-Konfiguration -> /lib/systemd/system

Übersicht der Befehle

su -<user> -> Benutzerwechseln, wird kein Benutzerparameter angegeben wechselt das System zum root

pwd -> present working directory z.T: wird bei diesem Befehl nicht der absolute Pfad angezeigt

apt install <program> -y -> Programme können so von den Quellen heruntergeladen und installiert werden, der -Y Parameter steht dafür, dass keine zusätzliche Bestätigung während des Installationsprozesses mehr benötigt wird. Weitere apt-Befehle im Abschnitt «apt Programm»

ip a -> Netzwerkkonfiguration abrufen

ifconfig -> Netzwerkkonfiguration abrufen (nettools Installation nötig)

dpkg-reconfigure -> Systemweite Einstellungen können konfiguriert werden wie z.B. Zeitzone, verwendete Shell usw.

systemctl status <service> -> Der Status eines Services kann erfragt werden. Alternativ auch /etc/init.d/<service> status

systemctl start <service> -> Ein Service kann gestartet werden. Alternativ auch /etc/init.d/<service> start

systemctl stop <service> -> Ein Service kann gestoppt werden. Alternativ auch /etc/init.d/<service> stop

systemctl enable <service> -> Startet den Autostart. Alternativ auch /etc/init.d/<service> enable

systemctl disable <service> -> Stoppt den Autostart. Alternativ auch /etc/init.d/<service> disable

systemctl get-default -> Funktionszustand Standard

systemctl set-default multiuser.target -> Funktionszustand Standard definieren

sudo <befehl> -> Befehl als root ausführen

usermod -aG <group> <user> -> Benutzer zu einer Gruppe hinzufügen

adduser <user> <group> -> Ein neuer Benutzer wird angelegt und einer Gruppe zugeordnet

more <file> -> Der Inhalt einer Datei wird ausgegeben

less <file> -> Der Inhalt der Datei wird ausgegeben, es kann im Gegensatz zu more mit den Pfeiltasten navigiert werden

list --installed |less -> Zeigt eine seitenweise Liste der installierten Software

man <program> -> Zeigt Informationen über ein Programm an (detailliert)

apt --help <program> -> Zeigt Informationen über ein Programm an (weniger detailliert)

passwd -> Das Passwort wird für den aktuellen Benutzer geändert

passwd user -> Das Passwort wird für einen definierten user geändert

whoami -> aktueller Benutzer

who -> Angemeldete Benutzer

id -> Welche ID hat der aktuelle Benutzer

id user -> Welche ID hat ein bestimmter Benutzer

uname -a -> Informationen über das Linux-System

clear -> Terminal leeren

history -> Zeigt den Verlauf der Befehle an (history 8; Letzte 8 Befehle / history !59; Führt den 59. Befehl aus)

dd if=/dev/sda of=/dev/sdb/mbr.backup -> Backup des MBRs erstellen

ps tree (-p) -> Übersicht der Prozesse (inkl. Relationen)

ls -la/-al -> Zeigt alle versteckten Dateien an

file <file> -> Zeigt das Format von einem File an.

nl <file> -> number line (Leerzeilen werden nicht nummeriert)

wc <file> -> word count (1. Ausgabe Zeilen, 2. Ausgabe Wörter, 3. Ausgabe Zeichen)

pr <file> -> printable (Bereitet ein File für den Druck vor)

echo «text» -> gibt Text aus. Wobei mit dem pipe und «tr» die Ausgabe übersetzt werden kann («| tr -d [2]» -> Filtert z.B. die 2 heraus, «| tr [:upper:] [:lower:]» -> Übersetzt Grossbuchstaben in Kleinbuchstaben.

mkdir -> Erstellt ein neuer Ordner

mv <src> <dst> -> Files & Ordner verschieben. Kann mit «mv dat2 dat3» auch Files umbenennen.

rm <file> -> Löscht eine Datei, mit dem Parameter -r können auch Ordner gelöscht werden

rmdir -> Löscht leere Ordner

rm -rf -> Löscht Ordner rekursiv und enforced

find -> Sucht Dateien direkt auf der Festplatte; Parameter: -name «Text»-> Explizit und Casesensitive, -iname «Text» -> Nicht Casesensitive, -type d -> Alle Ordner, -type f -> Alle Files. Kann auch mit Platzhalter(*) ergänzt werden.

locate -> Sucht Dateien in einer DB (schneller), DB muss mit sudo updatedb aktualisiert werden.

whereis -> Programme suchen, Zeigt den Speicherort von Programmen

mail -s «Error» <Mailaddress> </var/log/syslog -> Sendet eine Mail mit dem Betreff Error mit dem Inhalt des Syslogs

Bash Befehle

Befehl-Informationen

ls ist ein Befehl, welcher ein Alias von «ls –color=auto» ist. Wobei die Auto-Color Funktion verschiedene File-Type verschieden darstellt.

kill ist ein Befehl welcher Standardmässig builtin ist und somit IMMER dabei ist.

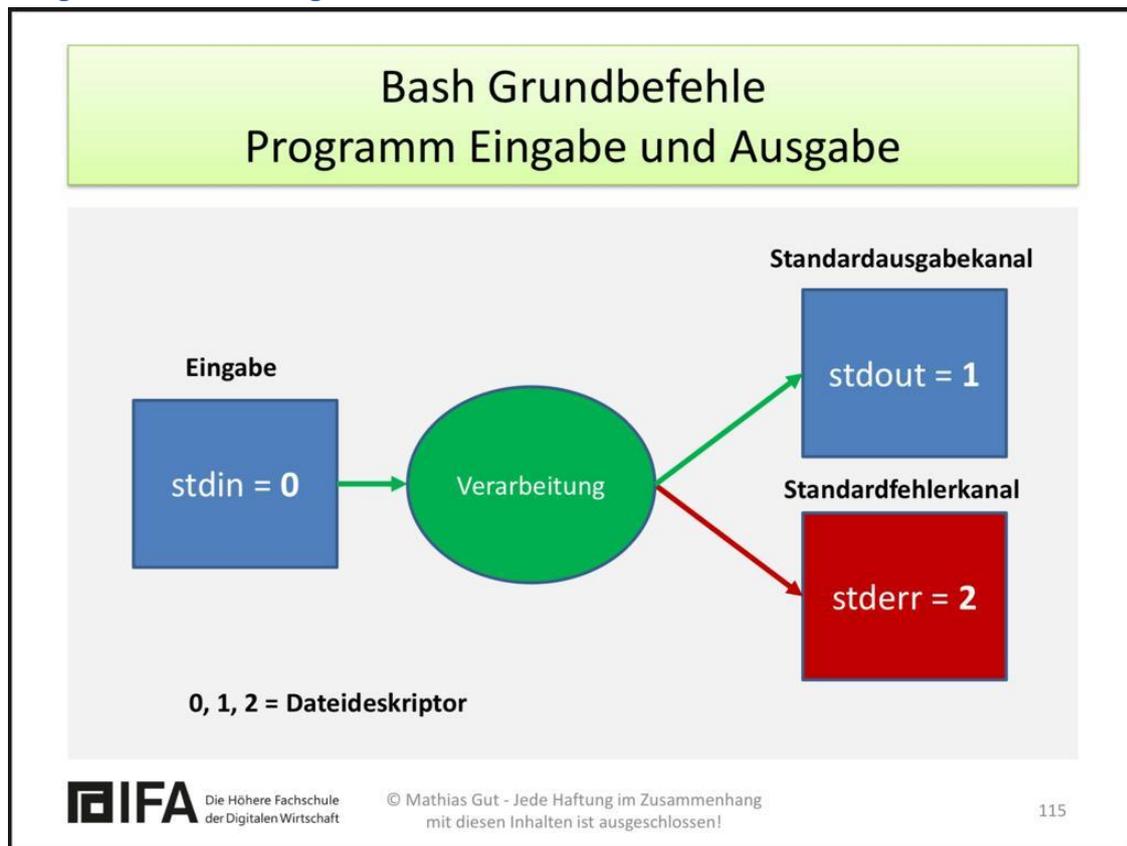
more ist ein Programm, welches zusätzlich installiert werden muss.

Mittels **type <command>** kann angezeigt werden, wo der Ursprung eines Befehles ist.

Gruppenarbeiten Bash Grundbefehle

Die Präsentationen zu den Bash Grundbefehlen findet man [hier](#) als Zip-Download.

Programm Ein- & Ausgabe



echo «abcd» 1>stdout.txt -> gibt den Standardausgabekanal aus

echo «abcd» 2>stdout.txt -> gibt den Standardfehlerkanal aus

echo «abcd» 2>>error.log 1>ok.log -> Trennt die Kanäle auf

echo «abcd» >>abcd.log 2>&1 -> Leitet den zweiten Kanal in den ersten

Aliase

Alias hinzufügen -> **alias <alias>«command»**

Alias entfernen -> **unalias <alias>**

Aliase werden so jedoch nicht persistent. Für persistente Aliase kann man diese im Userhome im «.bashrc»-Konfigfile persistent machen. Wobei eine Neuanmeldung nötig ist.

Es bietet sich auch an die Aliase in der Datei «.bash_aliases» im Userhome zu erstellen. Dies führt zu einer besseren Struktur und ist weniger Fehleranfällig.

Variablen

Mittels «env | less» werden die Umgebungsvariablen angezeigt.

Mittels «echo \$VARIABLE > echo \$USER» werden die Sessionvariablen angezeigt.

Um Variablen zu definieren muss man lediglich die Variable mit einem Gleichzeichen und dem Wert ergänzen:

```
«a=20»
```

Um sie dann aufzurufen ruft man die Variable mit einem vorhergehenden \$-Zeichen auf:

```
$a
```

```
> 20
```

Um alle Variablen anzuzeigen muss man den Befehl set | less ausführen.

Variablen können, müssen jedoch nicht definiert werden. Man kann mit dem Befehl «declare -i c=7» die Variable C mit dem Wert 7 versehen und mit -i die Variable als Integer definieren.

Auch kann man mit -r eine read-only Variable definieren.

Rechnen mit Variablen

Man kann in der Shell ebenfalls mit Variablen rechnen:

```
- echo $((a+d))
```

```
- let summe = a+d -> Schreibt das Ergebnis von a+d in die Variable «summe»
```

Sequenzielle Commands

Um Commands sequenziell nacheinander auszuführen können diese einfach mit «;» verbunden werden:

```
Bsp: ls ; ls -l oder apt update ; apt upgrade
```

Dabei werden die Befehle ausgeführt egal ob bei einem ein Fehler vorliegt. Würde also z.B. bei apt update ; apt upgrade wenig Sinn machen. Daher gibt es «apt update && apt upgrade». Da müssen beide/alle Commands ausführbar sein, dass diese auch ausgeführt werden.

Um eine Auswahl an Commands zu haben kann man «ls | | ls -l» brauchen. Dabei wird der erst mögliche Befehl ausgeführt. So kann man z.B. in einem Skript mehrere commands definieren für mehrere Distributionen mit anderer Befehl-Auswahl.

Bash skripting

Aufbau

Ein Skript unter Linux hat die Endung .sh und startet immer mit dem sogenannten shebang.

Der shebang definiert mit welcher Shell das Skript ausgeführt wird und sieht folgendermassen aus:

```
#!/bin/bash
```

Variable definieren

```
a=«Hallo»
```

```
b=Hallo
```

```
c=1
```

Skript ausführbar machen

Um ein Skript ausführbar zu machen, braucht es eine zusätzliche Berechtigung, welche nicht standardmässig vergeben wird um unabsichtliches Ausführen zu verhindern bzw. nur Skripts ausführbar zu machen. Mit dem Befehl «chmod +x skript.sh» wird die korrekte Berechtigung gesetzt.

Skript ausführen

Um ein Skript auszuführen muss immer ein Punkt vor dem Skript gesetzt werden. Dadurch wird verhindert, dass es zu einer versehentlichen Ausführung kommt:

```
./skript.sh
```

Variabelintegration

Um eine Variable in einen Text zu integrieren, wird die Variable mit `{a}` angesprochen. Wobei `{}` den Start und das Ende einer Variable definiert.

Array

Um ein Array zu definieren, gibt man den Befehl `ort = ('Bern' 'Zürich' 'Basel')` ein.

Um ein Wert zu einem Array hinzuzufügen `ort+=('Genf')`.

Um den ersten Wert des Arrays anzuzeigen `echo $ort`

Um alle Werte einer Arrays anzuzeigen `echo ${ort[*]}` oder `echo ${ort[@]}`

Um einen expliziten Wert auszulesen `echo ${ort[0]}`

Um eine Usereingabe in eine Variable einzulesen, kann man im Skript folgendes eintragen:
`read -p «Erste Zahl: » zahl1` wobei `zahl1` die Variable beschreibt, welche befüllt werden soll.

Schleifen

IF

```
If (($zahl1 > $zahl2); then
    <command>
elif <test-expression> ; then
    command
else
    command
fi
```

While

```
While <condition>
do
    Zeit = $(date +%H:%M:%S)
    echo [+] es ist: ${zeit}
    sleep 2
done
}
```

For

```
For i in $(seq 1 10)
do
    <commands>
done
```

Funktionen

Eine Funktion ist ein Programmteil welcher Operationen durchführt und an einem gewünschten Ort und Zeitpunkt aufgerufen werden kann.

Definition

Eine Funktion wird folgendermassen erstellt:

```
func_GroesserAls ()  
{  
<command>  
}
```

Wobei in den normalen Klammern Variablen in die Funktion mitgegeben werden können.

Aufruf

Um eine Funktion aufzurufen, kann man am gewünschten Programm/Skript-Ort den Funktionsnamen definieren:

```
funk_GroesserAls
```

Quellenverzeichnis

[https://de.wikipedia.org/wiki/GNU_General_Public_License#:~:text=Die%20GNU%20General%20Public%20License,und%20zu%20verbreiten%20\(kopieren\).](https://de.wikipedia.org/wiki/GNU_General_Public_License#:~:text=Die%20GNU%20General%20Public%20License,und%20zu%20verbreiten%20(kopieren).) (BASH – Die Shell unter Linux)

https://www.google.com/url?sa=i&url=https%3A%2F%2Fcommons.wikimedia.org%2Fwiki%2Ffile%3Alinux_einfache_architektur_500.png&psig=AOvVaw1sMr108N5HVUgmhM3Zva3r&ust=1641283387566000&source=images&cd=vfe&ved=0CAwQjhxqFwoTCLj_w7KPIfUCFQAAAAAdAAAAABAJ
(GNU/Linux Architektur)

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fde.wikipedia.org%2Fwiki%2FDatei%3ALinux-Distribution.svg&psig=AOvVaw0WhqDxi8PUfLU9ddQcix4W&ust=1641283786075000&source=images&cd=vfe&ved=0CAsQjRxqFwoTCKifwtuQlfUCFQAAAAAdAAAAABAJ> (Linux Distribution – Der Aufbau)

<https://de.wikipedia.org/wiki/Linux-Distribution> (Linux Distribution Tree)