

Linux-Kurzreferenz

Terminalprogramme und Shellprogrammierung



zusammengestellt von



2. Auflage - 2014

Linux-Kurzreferenz



*Terminalprogramme und
Shellprogrammierung*

Vorwort

Diese Kurzreferenz möchte ein kleiner Helfer für den alltäglichen Umgang mit **Linux** sein. Sie ist keine allumfassende vollständige Referenz.

Die Kurzreferenz bezieht sich im wesentlichen auf die Linux-Distributionen Linux Mint und Ubuntu. Bei den anderen Linux-Distributionen wie Xubuntu, Debian, Fedora, CentOS, Lubuntu, Mageia, OpenSuse, Symphony, Ark Linux, Manjaro, PCLinuxOS, Siduction, Bhodi Linux etc. sollte vieles sehr ähnlich funktionieren.

Diese Linux-Kurzreferenz dokumentiert auch die Entwicklung des Autors. Die ersten Gehversuche mit Linux wurden unter der Linux-Distribution SuSE unternommen. Danach erfolgte ein Umstieg auf Ubuntu und im Jahre 2014 auf die Linux-Distribution Linux Mint.

Einige Texte in dieser Kurzreferenz, werden im Laufe der Zeit sicher überholt sein. Trotz alledem sollten Sie sich dieser Texte annehmen. Letztendlich erhöhen sie das Verständnis für die Linux-Philosophie.

Die meisten Terminalprogramme bleiben von der verwendeten Linux-Distribution und Version i.d.R. unbeeindruckt - dies betrifft mehr als 90% der vorliegenden Linux-Kurzreferenz.

Auf die grafische Oberfläche wird in der vorliegenden Linux-Kurzreferenz nur am Rande eingegangen.

Die hier aufgeführten Befehle und Funktionen sind nicht vollständig in dieser Kurzreferenz dokumentiert, es sind nur die am häufigsten benutzten oder gesuchten Optionen zu den einzelnen Befehlen aufgelistet.

Wer hier einige Optionen oder einzelne Befehle vermisst, kommt nicht umhin die Kurzhilfe - über **<Befehlsname> --help** - die Infodateien - über **info <Befehlsname>** - oder die Manualseiten - über **man <Befehlsname>** zu konsultieren.

Bei den Linux-Distributionen Ubuntu, Linux Mint und den anderen Linux-Distributionen die auf Ubuntu basieren, ist der Benutzer root standardmäßig deaktiviert. Alle systemnahen Aktionen werden hier vom Hauptbenutzer mit einem vorangestellten **sudo** ausgeführt.

Nach der Installation von Linux befinden sich in der Datei /etc/passwd und /etc/group eine ganze Reihe von «Pseudousern». Die «Pseudousern» werden z.B. für das Funktionieren einer ganzen Reihe von Serverprozessen

benötigt.

Für die anderen Linux-Distributionen mit einem aktivierten Benutzer root (Wurzel), ist folgendes zu beachten:

Der Benutzer root (Gruppe: root) übernimmt die wichtigste Rolle in einem auf Linux basierenden System, die des Administrators. User-ID und Group-ID von root sind 0 (Null). Die Rolle von root wird auch als Superuser bezeichnet. root zeichnet sich dadurch aus, dass er mit den Rechten des Betriebssystems arbeitet. Alle Sicherheitsmaßnahmen, die das Betriebssystem zum Schutz der eigenen Integrität und zum Schutz der Benutzerprozesse und -daten eingebaut hat, sind für root außer Kraft gesetzt.

Benutzer können sich mit dem Systemprogramm **id** anzeigen lassen, zu welcher Hauptgruppe sie gehören - einschließlich aller Nebengruppen.

Hinweis: Falls auf der grafischen Oberfläche nichts mehr geht - z.B. Absturz der grafischen Oberfläche - so sollte folgendes versucht werden:

- drücken der Tastenkombination **[Strg] + [Alt] + [F2]**
- Login als **Hauptbenutzer** mit Root-Rechten oder **root** durchführen
- auf der Kommandozeile **sudo halt** oder **halt** eingeben

Der Rechner sollte jetzt ordnungsgemäß alle noch offenen Programme schließen und herunterfahren.

Inhaltsverzeichnis

Linux-Kurzreferenz.....	4
Vorwort.....	5
Einleitung: Linux-Kurzreferenz.....	9
A.....	12
B.....	56
C.....	58
D.....	78
E.....	114
F.....	120
G.....	132
H.....	151
I.....	160
J.....	182
K.....	189
L.....	193
M.....	208
N.....	231
O.....	247
P.....	248
Q.....	272
R.....	274
S.....	281
T.....	342
U.....	355
V.....	374
W.....	397
X.....	404
Y.....	408
Z.....	409
Anhang.....	416
Software-Empfehlungen.....	416
Suchmaschinen.....	421
Links.....	423
Literaturverzeichnis.....	427
Skript-Listings.....	429
Nützliche und hilfreiche Bash-Befehle.....	459
Tabellen.....	468
Einführung in die Shellprogrammierung.....	473
Hinweise zu einigen Programmen und speziellen Vorgehensweisen.....	529
Tipps und Tricks.....	553
Notizen.....	560
Stichwortverzeichnis.....	564

Einleitung: Linux-Kurzreferenz

Hinweis: Arbeitet man unter Linux auf der Kommandozeile (bash ... Bourne-Again Shell), so sollte folgende Regel beachtet werden. **KEINE MELDUNG, IST EINE GUTE MELDUNG.** Das heißt, wenn nach Eingabe eines Befehls eine Meldung auftaucht, dann ist meistens irgendetwas schief gelaufen. Diese Regel gilt bei sehr vielen Befehlen.

Die bash ist die Standardshell der meisten Linux-Distributionen und in dieser kann mit den Cursortasten [Pfeil hoch] und [Pfeil runter] in der eigenen Computergeschichte nach Wiederverwendbarem gesucht werden. Auch die gezielte Suche nach einer bestimmten Kommandozeile ist möglich. Mit der Tastenkombination [**Strg**] + [**r**] erscheint anstelle des bekannten Prompts die Aufforderung, den Suchbegriff einzugeben. Mit jedem eingegebenen Zeichen wird weiter rückwärts im Kommandozeilenspeicher nach einem passenden Begriff gesucht und die erste passende Zeile sofort angezeigt. Wenn auf diese Weise die gewünschte Zeile gefunden wurde - [**Esc**] drücken und das Kommando gegebenenfalls noch bearbeiten.

Anmerkung: Die History-Datei `.bash_history` wird im Home-Verzeichnis des Benutzers gespeichert. Der vorangestellte Punkt kennzeichnet dabei eine versteckte Datei. Die versteckten Dateien können über das Ansichtsmenü des Dateimangers angezeigt werden bzw. über die bash mittels des Befehls **ls -a**.

Mit [**Alt**] + [**.**] wird am aktuellen Cursor-Standpunkt ein Teil der letzten eingegeben Zeichenkette eingefügt, dadurch kann man sich Tipparbeit sparen. Durch mehrmaliges drücken des Punktes - bei festgehaltener [**Alt**]-Taste - geht man in der History zurück. Mit [**Shift**] + [**Bild hoch**] kann im Fenster des Kommandozeileninterpreter seitenweise gescrollt werden. Laufende Programme in der Befehlskonsole können über die Tastenkombination [**Strg**] + [**C**] vorzeitig beendet werden.

Linux unterscheidet zwischen Groß- und Kleinschreibung - dies sollte man sich unbedingt merken.

Mit **<Befehlsname> --help** wird die Syntaxhilfe für den eingegebenen Befehl aufgerufen.

Bei ausführbaren Programmen - z.B. Shellskripte, muss man dem Programm den gesamten Pfad mitteilen bzw. ihm ausdrücklich sagen, dass man den aktuellen Standort (`./`) als Ausgangssituation für die Abarbeitung des Befehls wünscht.

Beispiel: ./<Name des Shellskripts>

Wildcards und Pattern Matching

Folgende Zeichen haben im Terminalfenster (Konsole) eine spezielle Bedeutung:

- * ... beliebige Zeichenkette
- ? ... genau ein beliebiges Zeichen
- [] ... genau eines der genannten Zeichen
- [^] ... genau ein nicht genanntes Zeichen oder [!]
- {, } ... genau eine der Zeichenketten

siehe auch: ls, Shell, Anhang: Einführung in die Shellprogrammierung

Anders als unter Windows kann auf der grafischen Oberfläche mehr als ein Kommandozeileninterpret (Konsole) gestartet werden. Klicken Sie im einfachsten Fall einfach mehrmals auf das entsprechende Icon in der Kontrollleiste.

Nun stellt sich für Sie vielleicht die Frage - Warum sollte ich das tun? Bei zeitaufwendigen Programmabläufen ist die Konsole für die weitere Eingabe blockiert, so dass man in der Zwischenzeit einfach eine neue Konsole startet.

Ein häufiges Beispiel ist die Überwachung von Protokoll-Dateien:

tail -f /var/log/messages ... Befehl als Benutzer root oder als Hauptbenutzer mit root-Rechten über sudo aufrufen

sudo tail -f /var/log/ufw.log ... Log-Einträge der aktivierten Firewall fast in Echtzeit betrachten ; Befehl ist durch den Hauptbenutzer aufzurufen

Hinweis: Die Dateien /var/log/messages bzw. /var/log/ufw.log existieren nicht in allen Linux-Distributionen (**siehe:** Verzeichnis /var/log).

Regeln für Dateinamen

Die Dateinamen sollten auf keinen Fall Sonderzeichen (!\$%&{}[]#|<>*; etc.) enthalten. Weiterhin sollte man auf Leerzeichen in Dateinamen verzichten.

Bei einem grenzüberschreitenden Datenaustausch sollten für Dateinamen nur das englische Alphabet, Zahlen (0-9) und die Sonderzeichen Unterstrich (_), Minuszeichen (-) und der Punkt (.) verwendet werden.

Beim Einhalten dieser Regeln, werden die Dateinamen auch auf Rechner in

China, Thailand, Russland, Griechenland, den arabischen Staaten etc. korrekt angezeigt.
Abweichungen von dieser Regel sollten vorher ausgetestet werden.

Anmerkung: Dateinamen mit Leerzeichen innerhalb eines Terminalbefehls, werden vom Befehlsinterpreter als 2 oder mehrere Ausdrücke behandelt (Befehlsabarbeitung wird fehlerhaft). Um Dateinamen mit Leerzeichen innerhalb eines Terminalbefehls zu verwenden, sind die Dateinamen in einfache oder doppelte Hochkommas ('Regeln fuer Dateinamen.pdf' bzw. "Regeln fuer Dateinamen.pdf") einzuschließen oder dem Leerzeichen ist ein Backslash (Regeln\ fuer\ Dateinamen.pdf) voranzustellen.

Wichtig: Bei Experimenten die die Systemintegrität gefährden, sollte immer ein Test-System, ein zweiter Rechner genutzt werden.

Allgemeiner Hinweis: Einige der hier in der Linux-Kurzreferenz aufgeführten Programme sind nach einer Standardinstallation von Linux noch nicht verfügbar (**siehe auch:** `whereis`, `whatis`). Diese Programme sind mit den entsprechenden Werkzeugen (`apt-get`, `yum`, `zypper` etc.) der verwendeten Linux-Distribution nachträglich zu installieren.

A

ab

Das Standardwerkzeug unter Linux, um Antwortzeiten von Servern zu messen, ist der Apache Benchmark (`ab` ... Apache Benchmark; Bestandteil des Paketes `apache2-utils` bzw. `httpd-tools`). Das Kommandozeilen-Tool für Clients, das trotz seines Namens nicht nur Ergebnisse für den Apache Webserver liefert, sondern auch für andere Server Zugriffe über HTTP bzw. HTTPS auswertet.

`ab -c 10 -n 100 http://www.pcwelt.de/` ... das Kommando startet eine Messung über 10 konkurrierende Verbindungen (`-c 10`), mit insgesamt 100 Anfragen (`-n 100`)

Der abschließende Slash (/) ist notwendig, wenn keine benannte Datei, sondern ein Verzeichnis über die URL abgerufen wird.

Nach Abschluss der Anfrage präsentiert der Apache Benchmark eine Statistik.

Time taken for tests ... diese Zeile gibt an, wie viele Sekunden der Server für die Abarbeitung aller Anfragen benötigte

Request per second ... gibt an, wie viele Anfragen der Server pro Sekunde bedienen kann

Time per request ... gibt die durchschnittliche Ladezeit pro Anfrage an

siehe auch: `man ab`, `siege`, `htop`

antiword

`antiword` konvertiert Dateien vom `.doc`-Format ins reine Textformat.

`antiword <Dateiname.doc>` ... Konvertiert die angegebene Datei ins Textformat, die Ausgabe erfolgt am Bildschirm

`antiword <Dateiname.doc> | less` ... Konvertiert die angegebene Datei ins Textformat, die Ausgabe erfolgt mittels des Pagers `less` am Bildschirm. Über die Leertaste, den Pfeil- und Bildlauf Tasten kann im Dokument zeilen- oder seitenweise geblättert werden und über die Taste `[q]` wird `less` wieder

beendet.

antiword <Dateiname.doc> > <Dateiname.txt> ... konvertiert die angegebene Datei ins Textformat, die Ausgabe erfolgt in der angegebenen Textdatei

antiword -w 50 <Dateiname.doc> > <Dateiname.txt> ... konvertiert die angegebene Datei ins Textformat, die Ausgabe erfolgt in der angegebenen Textdatei, wobei die Spaltenbreite hier auf 50 Zeichen festgelegt wird

Beachte: antiword kann in der Version: 0.37 (Oktober 2009) nur Dateien im .doc-Format bis zur Word-Version 2000 öffnen und lesen.

siehe auch: lynx, w3m, Konvertierung von Textkodierungen, grep

apt-get

Das zentrale Programm, das man unter Ubuntu, Linux Mint und allen anderen Linux-Distributionen die auf Debian oder Ubuntu basieren zum Aktualisieren und Installieren von Paketen benutzt, ist apt-get. APT (Advanced Packaging Tool) ist eine fortschrittliche Schnittstelle zum Debian-Paketsystem dpkg.

Das Paketsystem dpkg verwendet eine Datenbank, die Informationen über installierte, nicht installierte und verfügbare Pakete enthält. Das Programm apt-get nutzt wiederum diese Datenbank, um herauszufinden, wie es die vom Benutzer angeforderten Pakete installieren soll und welche zusätzlichen Pakete benötigt werden, damit die ausgewählten Pakete ordnungsgemäß funktionieren.

apt-get [Optionen] befehl

apt-get [Optionen] install | remove paket1 [paket2 ...]

apt-get [Optionen] source paket1 [paket2 ...]

sudo apt-get install <Paketname> [<Paketname>] ... ein oder mehrere Programm-Pakete installieren

sudo apt-get -d install <Paketname> ... ein Programm-Paket nur herunterladen und nicht installieren (**siehe:** /var/cache/apt/archives/)

sudo apt-get check ... Diagnose-Tool; check aktualisiert den Paketzwischenspeicher und prüft, ob beschädigte Abhängigkeiten vorliegen

sudo apt-get clean ... clean entfernt nicht mehr benötigte Archive im Cache von apt-get (/var/cache/apt/archives/ und /var/cache/apt/archives/partial/)

sudo apt-get autoclean ... autoclean entfernt alle Pakete die nicht mehr in den Quellen vorhanden sind; entfernt Pakete die nur teilweise installiert wurden – bedingt durch ein Problem bei einer früheren Software-Installation

dpkg -s <Programmname bzw. Paketname> ... mit dpkg überprüfen, ob ein Programm vollständig installiert wurde

sudo add-apt-repository ppa:<Repository> ... externes APT-Repository hinzufügen; ist das Programm in der verwendeten Linux-Distribution nicht verfügbar, so ist die Datei /etc/apt/sources.list oder /etc/apt/sources.list.d/ direkt zu bearbeiten

sudo apt-get remove <Paketname> ... Programm-Paket entfernen, deinstallieren

sudo apt-get --purge remove <Paketname> ... entferne, deinstalliere Pakete restlos (inkl. Konfigurationsdateien)

sudo apt-get autoremove ... entferne, deinstalliere Pakete die während einer früheren Installation als Abhängigkeiten installiert wurden und jetzt nicht mehr benötigt werden

sudo apt-get --reinstall install <Paketname> ... ein bereits installiertes Pakets erneut installieren, z.B. ein installiertes Paket ist beschädigt oder funktioniert nicht wie erwartet

sudo apt-get update ... neue Paketinformationen einlesen

sudo apt-get -u upgrade ... nach dem Einlesen der neuen Paketinformationen (apt-get update), werden von allen Paketen die neusten verfügbaren Version installiert; die Option -u bzw. --show-upgraded lässt apt-get die komplette Liste der Pakete anzeigen, die aktualisiert werden sollen

sudo apt-get dist-upgrade ... spezielles Upgrade; Paketaktualisierung für die gesamte Distribution durchführen

Abgebrochenes Upgrade wieder aufnehmen:

Wenn aus irgendwelchen Gründen ein Upgrade der gesamten Distribution abbricht, so ist es leider nicht ausreichend, das Upgrade erneut aufzurufen. Stattdessen verwenden Sie den folgenden Befehl:

sudo dpkg --configure -a

siehe auch: man apt-get, apt-cache, dpkg

apt-cache

Das Programm apt-cache benutzt man zum suchen nach Paketinformationen. Alle Informationen bezieht apt-cache aus der Packages-Datei des Repositories. Das Paket muss also weder installiert sein noch heruntergeladen werden, um diese Suche durchführen zu können. Die wichtigsten Parameter von apt-cache sind **search** und **show**.

apt-cache search kwrite ... sucht nach dem Programmpaket kwrite

apt-cache show kwrite ... zeigt Details zum Paket kwrite an

```
karl@tux:~$ apt-cache show kwrite
Package: kwrite
Priority: optional
Section: editors
Installed-Size: 396
Maintainer: Kubuntu Developers <kubuntu-devel@lists.ubuntu.com>
Original-Maintainer: Debian Qt/KDE Maintainers <debian-qt-kde@lists.debian.org>
Architecture: i386
Source: kdebase
Version: 4:4.3.2-0ubuntu3
Replaces: kate (< 4:4.0.0-1), kwrite-kde4
Depends: kdebase-runtime (>= 4:4.3.2), kdelibs5 (>= 4:4.3.2), libc6 (>= 2.1.3), libqtcore4 (>= 4.5.1), libqtgui4 (>= 4.5.1), libstdc++6 (>= 4.1.1)
Conflicts: kate (< 4:4.0.0-1), kwrite-kde4
Filename: pool/main/k/kdebase/kwrite_4.3.2-0ubuntu3_i386.deb
Size: 127946
MD5sum: c5054a3b03d74cfc781e19515694b51d
SHA1: e190fba4a0ab94f6f000417d53322059406c9582
SHA256: 136f83348568f489f6ccebc10810b6da09f131b2fa9f8028cb942478441d4dd2
Description-de: text editor for KDE 4
  KWrite is the KDE 4 simple text editor. It uses the Kate editor component,
  so it supports powerful features such as flexible syntax highlighting,
  automatic indentation, and numerous other text tools.
.
```

Dieses Paket ist Teil des KDE-4-Basis-Anwendungen-Moduls.
Homepage: <http://www.kde.org/>
Bugs: <https://bugs.launchpad.net/ubuntu/+filebug>
Origin: Ubuntu

- **Version**
Die Version ist vor allem für den Update-Mechanismus wichtig: Gibt es nämlich eine neuere Version auf dem Repository, als gerade installiert ist, so kann man das System upgraden, indem man die neue Version installiert. Außerdem ist die Version wichtig, um die Abhängigkeiten zu managen: Manche Pakete setzen andere Pakete in einer bestimmten Version voraus.
- **Provides**
Dieses Schlüsselwort gibt an, welches virtuelle Paket vom betrachteten Paket bereitgestellt wird. Ein virtuelles Paket hat den Zweck, dass andere Pakete von ihm abhängen können, aber der Benutzer noch entscheiden kann, welche Software er zur Implementierung dieses speziellen Dienstes einsetzen möchte.
- **Depends**
Dieses Feld gibt die Abhängigkeiten durch Paketnamen an. Falls notwendig, wird hinter dem Paketnamen angegeben, welche Version genau (=) oder mindestens (>=) vorausgesetzt wird.
- **Suggests**
Diese Pakete werden vom betrachteten Paket vorgeschlagen. Im Regelfall werden diese Pakete jedoch nicht automatisch mit dem betrachteten Paket installiert.
- **Conflicts**
Diese Pakete können nicht gleichzeitig mit dem betrachteten Paket installiert werden.
- **Filename**
Hier kann das Paket auf dem Repository gefunden werden.
- **MD5sum**
Hier steht die Prüfsumme des Pakets. Anhand dieser können eventuelle Veränderungen festgestellt werden.
- **Deskriptiv**
Der letzte Punkt beinhaltet schließlich eine kurze Beschreibung des Pakets.

Anmerkung:

In den APT-Versionen einiger Linux-Distributionen (ab Linux Mint 17 bzw. Ubuntu 14.04 LTS), kann für die Installation von Programmpaketen eine Fortschrittsanzeige aktiviert werden. Der folgende Befehl schaltet die Fortschrittsanzeige ein: `echo 'Dpkg::Progress-Fancy "1";' | sudo tee -a /etc/apt/apt.conf.d/99progressbar`

Das Kommando ergänzt die Konfiguration von APT mit der Datei `/etc/apt/apt.conf.d/99progressbar`, die den Parameter `'Dpkg::Progress-Fancy "1";'` enthält. Die Einstellung gilt ab dem nächsten Aufruf von `apt-get`.

siehe auch: `apt-get`, `dpkg`, `apt-cache --help`

at

Prozesse oder Programme zeitgesteuert starten und beenden.

`at` ist nur sinnvoll für zeitgesteuerte Einzelaufträge. Für regelmäßig ablaufende Prozesse oder Routinen sollte `cron` verwendet werden (**siehe auch:** `crontab`).

Mit `at` können nur Programme gestartet werden, für deren Ausführung man auch die notwendigen Rechte besitzt. Für das Beispiel sollte man als **root** angemeldet sein, da normale Benutzer mit **find** nicht in allen Verzeichnissen suchen dürfen.

Beispiel: Da die Ausführung des Befehls **find** / **-nouser** (Dateien finden die keinem Benutzer gehören, also Datenmüll finden) sehr zeitaufwendig ist, sollte man die Suche in die späte Nacht verlegen.

Dazu gibt man ein:

at 23:00

Am veränderten Prompt (`at>`) gibt man nun den eigentlichen Befehl ein

find / -nouser

und schließt die Eingabe mit der Tastenkombination **[Strg] + [d]** ab. Die Zeitpunkte für die Ausführung kann man auf verschiedene Arten angeben, wie hier im Beispiel über **HH:MM**, aber auch mit **now +2 hours**. Damit würde das Programm in zwei Stunden starten. Statt **hours** sind auch die Angaben **minutes**, **days** und **weeks** und absolute Zeitangaben wie **teatime** (16:00 Uhr) und **midnight** möglich. Möchten Sie einen Befehl in drei Tagen um 14:00 Uhr starten, teilen Sie das **at** mit dem Kommando **at 14:00 + 3 days** mit. Um eine Aufgabe für 9:00 Uhr am nächsten Tag zu planen, lautet der Befehl **at 9:00 tomorrow**. Beachten Sie das diese Befehle nur

funktionieren, wenn der Rechner zur eingestellten Zeit auch eingeschaltet ist.

Unerledigte Aufträge zeigt **atq** an:

atq

Anhand der angezeigten Jobnummer (1. Zahl) kann man diesen auch wieder löschen:

atrm <Jobnummer>

Der at-Dämon gibt die Daten statt auf dem Bildschirm in eine Mail an den Auftraggeber aus.

Der Dämon muss evtl. erst gestartet werden: **rcatd start** (bzw. stop, status)

siehe auch: mail, /etc/init.d/atd

apropos <Suchbegriff>

Mit apropos kann man die Manualseiten nach einem Begriff durchsuchen.

apropos -w <*begriff*> ... der Suchbegriff enthält Wildcards

siehe auch: whatis

arp

ARP (Address Resolution Protocol) stellt das Bindeglied zwischen IP- und MAC-Adressen dar. Bevor ein IP-Paket verschickt werden kann, muss der Absender die MAC-Adresse (Hardware-Adresse, Media Access Control; z.B. 00:0D:88:21:4D:99; kann mittels ifconfig ermittelt werden) des Zielrechners ermitteln. Dazu versendet ARP einen Broadcast (Rundruf) mit der Frage »Who has <IP-Adresse>« ins LAN. Ist der Ziel-Host online, antwortet dieser mit einem an den Absender gerichteten ARP-Reply »<IP-Adresse> is at <MAC-Adresse>«. Diese Antwort speichert der Rechner temporär im ARP-Cache, um weitere Anfragen zu vermeiden.

arp -n ... zeigt die Rechner und die MAC-Adressen ihrer Netzwerkkarten in einem lokalen Netzwerk an, evtl. vorher einige Rechner anpingen (z.B. ping 192.168.1.1)

siehe auch: man arp, ifconfig, ip, Netzwerkkarte, Internet-Suchmaschine: ARP-Spoofing

awk

Die Programmiersprache dient, ähnlich wie sed, zum Auseinandernehmen und Verarbeiten von Streams. Jedoch bietet awk weitaus umfangreichere Möglichkeiten. Es handelt sich dabei schließlich um eine echte Programmiersprache. Solch eine Skriptsprache benötigt einen Interpreter, der die Anweisungen im Code einliest und ausführt. Dieser Interpreter heißt wie die Sprache selbst awk.

Die Syntax von awk ist sehr stark an die Programmiersprache C angelehnt, zudem ist die Sprache äußerst einfach zu erlernen -- bereits mit wenigen Zeilen Code kann ein komplexes Problem gelöst werden, für das man in Sprachen wie C einige Hundert Zeilen Quellcode benötigen würde.

Ein awk-Aufruf setzt sich aus mehreren Parametern zusammen, die teilweise optional sind.

awk [Ausdruck] [{ Anweisungen }] [Datei]

Ausdruck

Der erste Parameter ist ein regulärer Ausdruck. Dieser Ausdruck muss nicht immer übergeben werden. Da awk, wie auch sed, den Input-Stream zeilenweise durcharbeitet, kann durch den Ausdruck-Parameter jedoch eine Filterung realisiert werden. In englischen Büchern nennt man diesen Ausdruck oft Pattern, in einigen deutschsprachigen Büchern aber auch Muster – gemeint ist immer das Gleiche: der reguläre Ausdruck.

Anweisungen

Den zweiten Parameter stellen die awk-Anweisungen – der eigentliche Skriptcode – dar. Diese Anweisungen legen fest, welche Manipulationen am Input-Stream durchgeführt werden sollen. Wichtig ist dabei, dass diese Anweisungen in geschweifte Klammern eingebettet werden. Auch der Anweisungen-Parameter muss nicht immer übergeben werden.

Datei

Der Parameter Datei legt die Datei fest, aus der der Input-Stream gelesen werden soll. Sie müssen auch diesen Parameter nicht angeben – awk liest in diesem Fall von der Standardeingabe oder aus einer Pipe.

awk '/(.)*' /etc/group ... gesamten Inhalt der Datei /etc/group ausgeben

awk '/n/' /etc/group ... Zeilen ausgeben in denen der Buchstabe **n** vorkommt

awk '/x/ {print}' /etc/group ... Zeilen ausgeben in denen der Buchstabe **x** vorkommt; etwas andere Schreibweise als im vorherigen Beispiel

awk -F : '\$3 >= 1000 && \$3 < 2000 {print \$1}' /etc/passwd ... Ausgabe

der Benutzernamen deren User-ID ≥ 1000 und < 2000 ist

awk -F : '{print \$1, \$6}' /etc/passwd ... Ausgabe der Benutzernamen und ihres Home-Verzeichnisses (1. und 6 Feld der /etc/passwd)

awk '/nobody/ {print}' /etc/passwd ... falls ein Eintrag nobody in der Datei /etc/passwd existiert, so wird die entsprechende Zeile der Datei passwd zurückgegeben

ping -c 1 192.168.1.53 2> /dev/null | grep packet | awk '{print \$4}' | grep -c ^[1]\$... überprüft ob ein Rechner online ist; gibt 1 (Online) oder 0 (Offline) zurück

siehe auch: man awk und Internet

Autostart mit ROOT-Rechten: Start-/Stop-Skript erstellen

Man kann natürlich eigene Start/Stop-Skripte erstellen. Dazu sollte man sich die bereits in /etc/init.d/ liegenden Skripte als Vorbild nehmen oder die offizielle Vorlage /etc/init.d/skeleton als Ausgangsbasis nutzen.

Wem die Start-/Stop-Skripte und die offizielle Vorlage (/etc/init.d/skeleton) zu komplex sind, der kann sich ein relativ einfaches Start-/Stop-Skript für fast jeden Zweck selbst schreiben. Man benötigt lediglich eine Datei, die auf folgendem Beispiel basiert:

```
#!/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 Beschreibung
### END INIT INFO
# Author: Name <email@domain.tld>

# Aktionen
case "$1" in
    start)
        /opt/beispiel start
        ;;
    stop)
        /opt/beispiel stop
        ;;
    restart)
```

```
/opt/beispiel restart
;;
esac

exit 0
```

Der Kommentar-Text im Kopfteil der Datei ist dabei sehr wichtig und wird vom Befehl `update-rc.d` verwendet. Er sollte angepasst, aber nicht gelöscht werden! Die Datei speichert man z. B. in `/etc/init.d/beispiel` und macht die Datei per

`sudo chmod 755 /etc/init.d/beispiel`

ausführbar. Anschließend fügt man das Skript mit dem Befehl

`sudo update-rc.d`

in die entsprechenden Runlevel ein. Von nun an, wird bei jedem Systemstart der Dienst **beispiel** mit dem Parameter `start` aufgerufen und damit gestartet. Beim Herunterfahren des Systems wird der Dienst **beispiel** automatisch gestoppt.

Manuell kann der Dienst über das Start-/Stop-Skript `beispiel` gesteuert werden.

Start-/Stop-Skript `beispiel`:

`sudo /etc/init.d/beispiel start` ... startet den Dienst

`sudo /etc/init.d/beispiel stop` ... stoppt den Dienst

`sudo /etc/init.d/beispiel restart` ... stoppt und startet den Dienst

Anmerkung: Die Datei **skeleton** ist eine etwas umfangreichere Vorlage und liegt auf jedem Linux-System im Verzeichnis `/etc/init.d/`. Folgende Parameter werden von Start-/Stop-Skripte unterstützt: **start**, **stop**, **status**, **restart**, **reload**, **force-reload**

Bearbeiten von Diensten per Hand

Debian und **Ubuntu** besitzt mit dem Befehl `update-rc.d` ein mächtiges Werkzeug, um Dienste in den einzelnen Runleveln zu aktivieren bzw. zu deaktivieren. Am Beispiel von `pcmciautils` wird nun erklärt, wie man einen Dienst aus dem aktuellen Runlevel entfernen und wieder hinzufügen kann.

Das Entfernen ist nicht sonderlich schwer. Man öffnet ein Terminalfenster

und gibt den Befehl:

sudo update-rc.d -f pcmciautils remove

ein. Das Kommando liefert hierbei beispielsweise folgende Ausgabe:

```
Removing any system startup links for /etc/init.d/pcmciautils ...  
/etc/rc0.d/K88pcmciautils  
/etc/rc6.d/K88pcmciautils  
/etc/rcS.d/S13pcmciautils
```

Hier erkennt man die oben beschriebenen Hintergründe. Die Links zum Start-/Stop-Skript von pcmciautils - /etc/init.d/pcmciautils - werden aus den Runlevels gelöscht. Möchte man den Dienste wieder in die dafür vorgesehenen Runlevel einfügen, so macht man das mit dem Befehl

sudo update-rc.d pcmciautils defaults

Dabei werden die Links wieder angelegt. Nachdem man einen Dienst aus den Runleveln entfernt hat, sollte man ihn abschließend von Hand stoppen

sudo invoke-rc.d pcmciautils stop

Sonst würde er beim nächsten Herunterfahren des Systems nicht sauber beendet werden, und evtl. drohen Datenverluste.

Experten-Info:

Dieser Schritt allein entfernt den Dienst leider nicht für immer. Bei einer Aktualisierung ("update") des entsprechenden Pakets (hier also pcmciautils bzw. pcmcia-cs) merkt die Paketverwaltung, dass diese Links nicht mehr existieren, und legt sie wieder an, da das System nun davon ausgeht, dass das Paket zum ersten Mal installiert werden würde. Der Weg, um einen Dienst ein für alle Mal zu deaktivieren, besteht daher darin, die Verknüpfungen nur zu entschärfen.

Verknüpfungen entschärfen:

Das vorangegangene Löschen des Dienstes liefert die entscheidenden Informationen, wo die Start- und Stop-Skripte verlinkt werden müssen. Daraufhin legt man selbständig passende Links mit den soeben erworbenen Informationen an.

sudo update-rc.d pcmciautils stop 13 S . stop 88 0 6 .

Um pcmciautils wieder als Dienst einzufügen, entsprechend wieder

```
sudo update-rc.d -f pcmciautils remove  
sudo update-rc.d pcmciautils start 13 S . stop 88 0 6 .
```

eingeben.

Siehe auch: Skript-Listings: Start- und Stop-Skript für XAMPP, initctl, service

Autostart mit KDE

Für den Autostart eines Programms, muss ein Link in dem versteckten Verzeichnis `.kde/Autostart` im Home-Verzeichnis angelegt werden.

```
ln -s /Pfad/zum/Programm/Programmname $HOME/.kde/Autostart/  
z.B. ln -s /opt/kde3/bin/kcalc $HOME/.kde/Autostart/
```

Dasselbe kann auch über das Kontrollzentrum erreicht werden, indem mit den Systemnachrichten beim Start von KDE gleichzeitig ein Programm aufgerufen wird.

Anmerkung: Ein einzelner Kommandozeilenaufruf der mit root-Rechten laufen soll, kann auch in der `/etc/init.d/boot.local` eingetragen werden.

avconv

avconv ist ein Kommandozeilenprogramm, zur Konvertierung von Video-, Audio- oder Bildformate. Das Programm avconv ist Bestandteil des Paketes libav-tools (Fork von ffmpeg; Januar 2011). Das Paket libav-tools enthält die Programme avconv, avprobe, avserver, avplay und qt-faststart.

Hinweis: In der nachfolgenden Version (Linux Mint 19 und 21) wurden die Programmpakete von avconv wieder durch die Programmpakete von ffmpeg ersetzt. Die Syntax von ffmpeg ist mit der Syntax von avconv fast identisch (**Beispiel:** `avconv -i VTS_01_4.VOB -vcodec mpeg4 -b:v 2048k output.mp4` → `ffmpeg -i VTS_01_4.VOB -vcodec mpeg4 -b:v 2048k output.mp4`).

avconv ... Universeller Konverter für Audio-, Video- und Bilddateien

avprobe ... Programm zur Streamanalyse von Audio- und Videodateien

avserver ... Streamingserver für Audio- und Video-Streams; zur Zeit wird von der Benutzung des Streamingserver abgeraten (Stand: 2014)

avplay ... Sehr einfacher, aber universeller Multimedia-Player

qt-faststart ... Dienstprogramm, das Quicktime-Dateien neu anordnet, so

dass Netzwerk-Streaming möglich ist

Die Bibliotheken von Libav enthalten weitere Kommandozeilenwerkzeuge.

libavcodec ... enthält alle Encoder und Decoder von Libav

libavformat ... enthält Muxer und Demuxer für Audio- und Videoformate

libavutil ... Hilfsbibliothek, die Routinen enthält, die von mehreren Libav-Anwendungen benutzt werden

libavfilter ... Ermöglicht die Modifizierung von Audio- und Videodateien zwischen Encoder und Decoder

libavresample ... enthält Routinen, um Audio neu einzulesen (resampling)

libswscale ... enthält Routinen, um Video zu skalieren und Farbraum/Pixelformat umzuwandeln

Hinweis: Die Video- und Audioqualität des neu erstellten Videos kann maximal nur so gut sein, wie die Qualität des Ausgangs-Videos (INPUTFILE). Die Qualität der Videos wird im wesentlichen vom verwendeten Video- und Audio-Codec, sowie von der Bitrate bestimmt.

siehe auch: man libav-tools

```
avconv [ OPTION GLOBAL ] [ OPTION INPUTFILE ] -i INPUTFILE  
[ OPTION VIDEO OUTPUTFILE ] [ OPTION AUDIO OUTPUTFILE ]  
OUTPUTFILE
```

Allgemeine Optionen:

-L ... Lizenz anzeigen

-h ... Hilfe anzeigen

-version ... Version anzeigen

-formats ... verfügbare Formate anzeigen

-codecs ... verfügbare Codecs anzeigen

-f FORMAT ... FORMAT für Ein-/Ausgabe nutzen

-threads ANZAHL ... ANZAHL an Threads verwenden (erhöht die Geschwindigkeit bei Mehrkernprozessoren)

-i INPUTFILE ... Name des Input-Files

-shortest ... hört auf, wenn das kürzeste Inputfile (Audio oder Video) endet

-itsoffset 00:00:03.5 ... Video- und Audiospur werden um 3 Sekunden und 5 Millisekunden verschoben; die Audiospur wird mehr zum Anfang des Videos verschoben

-y ... vorhandene gleichnamige Datei überschreiben

Video-Optionen (siehe: Beispiele):

-codec copy ... Video- und Audio-Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

-vcodec copy ... Video-Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)
-vcodec CODEC ... CODEC zum Dekodieren/Enkodieren nutzen, falls als Eingabe-/Ausgabeoption genutzt. copy angeben, um den Stream zu kopieren; c:v CODEC
-f FORMAT ... Format des Videos (mp4, mov, flv, avi ...)
-aspect 16:9 ... Seitenverhältnis 16:9 (sinnvolle Werte: 4:3, 16:9, 1.3333, 1.7777)
-b:v BITRATE ... Video-Bitrate setzen (in Bit/s)
-r FRAMERATE ... Framerate setzen (in Frames/Sekunde)
-s GRÖSSE ... Größe des Videos setzen (Breite x Höhe, 640x480, vga ...)
-aspect VERHÄLTNIS ... Seitenverhältnis setzen (z.B. 4:3 oder 16:9)
-vn ... keine Video-Information übertragen, Video deaktivieren
-ss 0:0:12.200 ... Start des Videos bei 12 Sekunden und 200 Millisekunden
-t 0:0:5 ... Länge des Videos 5 Sekunden (kann auch als -t 5 angegeben werden)
-qmin 6 ... qmin (quantiser scale parameter) ist ein Faktor, der die Stärke der Kompression regelt; bei MPEG-4 liegt der Faktor zwischen 1 und 31, wobei nur ganze Zahlen erlaubt sind; 1 .. sanfte Kompression und gute Qualität; 31 .. starke Kompression; optimale Werte liegen etwa bei 2 - 8
-loop ... Endlosschleife für eine bestimmte Zeit (-t 0:0:5)
-vf crop=660:300:70:80 ... Zielformat des Videos ist 660x300; oberer Balken von 80 Pixel wird entfernt; linker Balken von 70 Pixel wird entfernt; Nullpunkt des neuen Bildes ist 70 Pixel nach rechts und 80 Pixel nach unten verschoben
-vf pad=660:495:0:97:0xff0000 ... Videoformat verändern; Farbbalken-Höhe 97 Pixel; Farbwert des Farbbalken über und unter dem Bild 0xff0000 (rot, hexadezimale Schreibweise)
-setpts=0.5*PTS ... Videogeschwindigkeit verdoppeln (fast motion)
-setpts=2.0*PTS ... Videogeschwindigkeit halbieren (slow motion)
-map 0:v ... speichere alle Bildspuren (meist gibt es nur eine) ; siehe Abschnitt: I: Eine oder mehrere Audiospuren mit der Option -map extrahieren (VOB, MPG)

Audio-Optionen (siehe: Beispiele):

-acodec CODEC ... CODEC zum Dekodieren/Enkodieren nutzen (libmp3lame, libvorbis ...); copy angeben, um den Stream zu kopieren; c:a CODEC
-b:a BITRATE ... Audio-Bitrate setzen (Bit/s); sinnvolle Werte: 32k, 40k, 48k, 56k, 64k, 80k, 96k, 112k, 128k, 160k, 192k, 224k, 256k, 320k, 448k;
Hinweis: Die alte Option -ab für die Festlegung der Audio-Bitrate sollte nicht mehr verwendet werden, sie gilt als deprecated, veraltet.
-ar RATE ... Abtastrate, Samplingfrequenz setzen (in Hz; 32000, 44100,

48000)

-ac KANÄLE ... Anzahl der Audiokanäle setzen (1 .. Mono, 2 .. Stereo, ohne Angabe – default: Mono)

-aq 5 ... Audioqualität Stufe 5 (Vorgabe); die Option ist nicht für alle Audio-Codecs zulässig; zulässige Werte 0 - 10 (0 ... hohe Qualität, 10 ... niedrige Qualität)

-dialnorm WERT ... Dialnorm gibt an, wie weit der durchschnittliche Lautstärkepegel des Programms unter den digitalen 100% des Endwerts (0 dBFS) liegt. -31dB führt zu keiner Änderung der Lautstärke und ist der Standardwert. Gültige Werte sind ganze Zahlen im Bereich -31 bis -1.

-vol 20 ... Mit -vol wird die Lautstärke gesteuert; Grundwert, der keine Änderung bewirkt ist 256; ein höherer Wert hebt und ein tieferer Wert senkt die Lautstärke

-an ... keine Audio-Information übertragen, Audio deaktivieren

-map 0:a:(Zahl) ... speichere Audiospur (Zahl), **Hinweis:** Zählung fängt bei 0 an! ; siehe Abschnitt: I: Eine oder mehrere Audiospuren mit der Option -map extrahieren (VOB, MPG)

Hinweis: Die Anweisungen -vcodec und -acodec werden in Zukunft durch -c:v und -c:a abgelöst.

Inhaltsverzeichnis

A: In den Beispielen verwendete Codecs und Formate

B: Video- und Audio-Codecs

C: Video- und Audio-Formate

D: Videonormen

E: Informationen zu einem Video und anderen Dateien anzeigen

F: Player – Videos mit avplay abspielen

G: Videodateien umwandeln, konvertieren

H: Audiodaten extrahieren, entfernen, hinzufügen

I: Eine oder mehrere Audiospuren mit der Option -map extrahieren (VOB, MPG)

J: Audiodateien umwandeln, konvertieren

K: Bilddateien umwandeln, konvertieren

L: Bilder aus ein Video extrahieren

M: Video aus Einzelbilder erstellen

N: Video aus einem einzelnen Bild erstellen

O: HTML 5 -konformes Video erstellen

P: Farbbalken entfernen, hinzufügen

Q: Bildausschnitt aus einem Film herausschneiden

R: Videos zusammenfügen, verbinden

S: Bitrate, Framerate und Bildgröße eines HD-Films reduzieren

T: Bildschirm-Aufzeichnung

U: Video-Untertitel einfügen, fest »einbrennen« in die Videodatei

V: Webcam - Aufzeichnen eines Videos

A: In den Beispielen verwendete Codecs und Formate

Video-Codecs (-vcodec) ... mpeg4 (.mp4, .avi, .mov), libx264 (.mp4, .avi, .mov - H.264), libxvid (.mp4, .avi, .mov) libtheora (.ogg oder .ogv), libvpx (.webm), mpeg2video (.mpg)

Audio-Codecs (-acodec) ... libmp3lame (.mp3), flac (.flac), libvorbis (.ogg, .webm), pcm_s16le (.wav)

Formate (-f) ... mjpeg, mp4, flv, mp3, avi, mov, image2, x11grab, video4linux2

siehe auch: avconv -codecs, avconv -formats

B: Video- und Audio-Codecs

Die Ausgabe der verfügbaren Codecs mit der Option -codecs könnte wie folgt aussehen:

avconv -codecs

D ... Dekodierung möglich

E ... Enkodierung möglich

V/A/S ... Video/Audio/Untertitel-Codec

S ... Codec unterstützt Slices

D ... Codec unterstützt direktes Rendering

T ... Codec kann Eingaben verarbeiten, die an einer beliebigen Stelle abgeschnitten sind.

DEV D	bmp	BMP image
DES	dvbsub	DVB subtitles
DES	dvdsup	DVD subtitles
DEV D	dvvideo	DV (Digital Video)
DEA D	flac	FLAC (Free Lossless Audio Codec)
DEV D	flashsv	Flash Screen Video
D V D	flashsv2	Flash Screen Video v2
DEVSD	flv	Flash Video (FLV) / Sorenson Spark / Sorenson H.263
D V D	h264	H.264 / AVC / MPEG-4 AVC / MPEG-4 part 10
DEV D	jpegl	JPEG-LS
EA	libmp3lame	libmp3lame MP3 (MPEG audio layer 3)

D V D	libopenjpeg	OpenJPEG based JPEG 2000 decoder
EV	libtheora	libtheora Theora
EA	libvorbis	libvorbis Vorbis
DEV	libvpx	libvpx VP8
EV	libx264	libx264 H.264 / AVC / MPEG-4 AVC /
MPEG-4	part 10	
EV	libxvid	libxvidcore MPEG-4 part 2
EV	ljpeg	Lossless JPEG
DEV D	mjpeg	MJPEG (Motion JPEG)
D V D	mjpegb	Apple MJPEG-B
D A D	mp3	MP3 (MPEG audio layer 3)
DEVSDT	mpeg2video	MPEG-2 video
DEVSDT	mpeg4	MPEG-4 part 2
DEVSD	msmpeg4	MPEG-4 part 2 Microsoft variant
	version 3	
DEV D	png	PNG image
DEV D	ppm	PPM (Portable PixelMap) image
DEV	rawvideo	raw video
D S	srt	SubRip subtitle
DEV D	targa	Truevision Targa image
D VSD	theora	Theora
DEV D	tiff	TIFF image
DEA D	vorbis	Vorbis
D A D	wavpack	WavPack
	[...]	

C: Video- und Audio-Formate

Die Ausgabe der verfügbaren Formate mit der Option `-formats` könnte wie folgt aussehen:

avconv -formats

D = unterstützt Demuxing

E = unterstützt Muxing

E	3g2	3GP2 format
E	3gp	3GP format
DE	au	SUN AU format
DE	avi	AVI format
E	avm2	Flash 9 (AVM2) format
E	dvd	MPEG-2 PS format (DVD VOB)
DE	ffmetadata	FFmpeg metadata in text format
D	film_cpk	Sega FILM/CPK format
DE	filmstrip	Adobe Filmstrip
DE	flac	raw FLAC
D	flic	FLI/FLC/FLX animation format

DE flv FLV format
 E gif GIF Animation
 DE h261 raw H.261
 DE h263 raw H.263
 DE h264 raw H.264 video format
 DE image2 image2 sequence
 E ipod iPod H.264 MP4 format
 DE m4v raw MPEG-4 video format
 E matroska Matroska file format
 D matroska,webm Matroska/WebM file format
 E md5 MD5 testing format
 DE mjpeg raw MJPEG video
 E mov MOV format
 D mov,mp4,m4a,3gp,3g2,mj2 QuickTime/MPEG-4/Motion JPEG
 2000 format
 E mp2 MPEG audio layer 2
 DE mp3 MPEG audio layer 3
 E mp4 MP4 format
 DE mpeg MPEG-1 System format
 E mpeg1video raw MPEG-1 video
 E mpeg2video raw MPEG-2 video
 DE mpegts MPEG-2 transport stream format
 D mpegtsraw MPEG-2 raw transport stream format
 D mpegvideo raw MPEG video
 E mpjpeg MIME multipart JPEG format
 D msnwctcp MSN TCP Webcam stream
 DE ogg Ogg
 DE rm RealMedia format
 DE srt SubRip subtitle format
 E svcd MPEG-2 PS format (VOB)
 DE swf Flash format
 E vcd MPEG-1 System format (VCD)
 D video4linux2 Video4Linux2 device grab
 E vob MPEG-2 PS format (VOB)
 DE wav WAV format
 E webm WebM file format
 D x11grab X11grab
 DE yuv4mpegpipe YUV4MPEG pipe format
 [...]

D: Videonormen

Folgende Formate haben sich mehr oder weniger etabliert:

Jahr	Standard	Auflösung	Video	Audio	Datenrate	Endung
1993	VCD	352×288	MPEG-1		1.5 MBit/s	.vcd
1994	DV	720×576	MPEG-1		25 MBit/s	.dv

1994	DVB-T	720×576	H.262	MP2	3 MBit/s	.ts
1995	DVD	720×576	H.262	AC3	5 MBit/s	.vob
	HDV1	1280×720	H.262		19 MBit/s	
	HDV2	1440×1080	H.262		25 MBit/s	
	HDTV	1920×1080	H.264			
2002	Blu-ray	1920×1080	H.264	DTS	36 MBit/s	.mts
2003	DVB-T2	1920×1080	H.26x	AAC	20 MBit/s	.t2s
2005	HD-DVD	1920×1080	H.264	AAC	29 MBit/s	.evo
2011	AVCHD	1920×1080	H.264	AAC	24 MBit/s	

E: Informationen zu einem Video und anderen Dateien anzeigen

avconv -i QUELLDATEI ... Informationen über eine Datei anzeigen

Bildschirmausgabe: avconv -i filename.vob

avconv version 0.8.6-6:0.8.6-1, Copyright (c) 2000-2013 the Libav developers

Input #0, mpeg, from 'filename.vob':

|
Container

Duration: 00:43:02.79, start: 0.053678, bitrate: 3558 kb/s

	Codec	Farbraum	Auflösung
Stream#0.0[0x1e0]:Video: mpeg2video (Main), yuv420p, 720x576 [PAR 16:15 DAR 4:3], 7500 kb/s, 25 fps, 25 tbr, 90k tbn, 50 tbc			
	Bitrate	Bilder/s	

Stream #0.1[0x80]: Audio: ac3, 48000 Hz, stereo, s16, 192 kb/s	Codec	Frequenz	Bitrate

F: Player – Videos mit avplay abspielen

avplay output.mp4 ... Video mit dem avconv-Player abspielen

mplayer output.mp4 ... Video mit dem alternativen Player MPlayer abspielen

avplay -vf scale=WIDTH:HEIGHT INFILE ... Film bei Wiedergabe skalieren

avplay -aspect 4:3 output.mp4 ... Seitenverhältnis (4:3, 16:9) festlegen;

mitunter wird das Seitenverhältnis des Videos von avplay nicht automatisch erkannt

mplayer -monitoraspect 4:3 output.mp4 ... Seitenverhältnis (4:3, 16:9) festlegen; mitunter wird das Seitenverhältnis des Videos von mplayer nicht automatisch erkannt

G: Videodateien umwandeln, konvertieren

1. AVI-Video → MPEG4

avconv -i input.avi -vcodec mpeg4 -b:v 1024k -s 649x486 output.mp4

-i ... Name des Input-Files

-vcodec mpeg4 ... Video-Codec für output.mp4

-b:v 1024k ... Video-Bitrate 1024 kBit/s

-s 649x486 ... Bildgröße 649x486 Pixel

Hinweis: Die Angabe für die Videogröße (-s 649x486) kann normalerweise entfallen, nur bei ungewöhnlichen Videoformaten von output.mp4, sind die Bildschirmausgaben (avplay output.mp4) auf sinnvolle Pixelangaben für die Videogröße zu durchsuchen (Stream #0.0(und): Video: mpeg4 (Simple Profile), yuv420p, 649x486 [PAR 1:1 DAR 649:486], 886 kb/s, 25 fps, 25 tbr, 25 tbn, 25 tbc) und im avconv-Befehl zur Erstellung des Videos einzutragen.

avconv -i input.avi -codec copy output.mp4 ... AVI-Video in ein MP4-Video konvertieren

-codec copy ... Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

avconv -i input.avi -b:v 9000k -b:a 256k output.mpg ... MPEG-Video das von vielen Mediaplayern problemlos erkannt und abgespielt wird

-b:v 9000k ... Video-Bitrate 9000 kBit/s

-b:a 256k ... Audio-Bitrate 256 kBit/s

2. AVI-Video → Flash_Video (FLV)

avconv -y -i input.avi -b:v 1024k -s 640x480 -ar 44100 output.flv

-y ... vorhandene gleichnamige Datei überschreiben

-i ... Name des Input-Files

-b:v 2048k ... Video-Bitrate 2048 kBit/s

-s 640x480 ... Größe des Videos in Pixel; mögliche Angaben: vga, svga, hd720 bzw. die Angabe erfolgt in Pixel (640x480, 800x600 ...)
-ar 44100 ... Audio Samplingfrequenz 44100 Hz

3. AVI → OGV, OGG

avconv -i input.avi -b:v 2048k output.ogg

Audio-Codecs ... flac (Vorgabe)

Video-Codec ... libtheora (Vorgabe)

avconv -y -i input.avi -b:v 2048k -acodec libvorbis -b:a 192k -ac 2 output.ogg

-y ... vorhandene gleichnamige Datei überschreiben

-i ... Name des Input-Files

-b:v 2048k ... Video-Bitrate 2048 kBit/s

-acodec libvorbis ... Audio-Codec libvorbis (siehe: avconv -codecs)

-b:a 192k ... Audio-Bitrate 192 kBit/s

-ac 2 ... Audio-Channel 2 (Stereo); 1 (Mono); ohne Angabe-default (Mono)

ffmpeg2theora -v 8 input.avi -o output.ogv

-v ... Videoqualität; 0 .. geringe Qualität, 10 .. höchste Qualität, 5 .. default

-o ... Name der Ausgabedatei

-h ... Hilfe ausgeben

Anmerkung: ffmpeg2theora wird im Laufe der Zeit durch das Programm avconv2theora ersetzt.

4. AVI → WEBM

avconv -i input.avi -s 781x585 output.webm

Audio-Codecs ... libvorbis (Vorgabe)

Video-Codec ... libvpx (Vorgabe)

-i ... Name des Input-Files

-s 781x585 ... Größe des Bildes in Pixel

Hinweis: Die Angabe für die Videogröße (-s 781x585) kann normalerweise entfallen, nur bei ungewöhnlichen Videoformaten von output.webm, sind die Bildschirmausgaben (avplay output.webm) auf sinnvolle Pixelangaben für die Videogröße zu durchsuchen Stream #0.0: Video: vp8, yuv420p,

781x585, PAR 1:1 DAR 781:585, 25 fps, 25 tbr, 1k tbn, 1k tbc) und im avconv-Befehl zur Erstellung des Videos einzutragen.

avconv -i input.avi -b:v 1024k -s 781x585 -acodec libvorbis -b:a 128k -ac 2 -ar 44100 output.webm

-i ... Name des Input-Files
-b:v 1024k ... Video-Bitrate 1024 kBit/s
-s 781x585 ... Größe des Bildes in Pixel
-acodec libvorbis ... Audio-Codec libvorbis (siehe: avconv -codecs)
-b:a 128k ... Audio-Bitrate 128 kBit/s
-ac 2 ... Audio-Channel 2 (Stereo); 1 (Mono); ohne Angabe - default (Mono)
-ar 44100 ... Audio Samplingfrequenz 44100 Hz

5. AVI → QuickTime-Video (MOV)

avconv -i input.avi -f mov -b:v 1024k -s 649x486 output.mov

-i ... Name des Input-Files
-f mov ... Format des Videos: QuickTime (siehe: avconv -formats)
-b:v 1024k ... Video-Bitrate 1024 kBit/s

Hinweis: Die Angabe für die Videogröße (-s 649x486) kann normalerweise entfallen, nur bei ungewöhnlichen Videoformaten von output.mov, sind die Bildschirmausgaben (avplay output.mov) auf sinnvolle Pixelangaben für die Videogröße zu durchsuchen (Stream #0.0(eng): Video: mpeg4 (Simple Profile), yuv420p, 352x576 [PAR 59:27 DAR 649:486], 683 kb/s, 25 fps, 25 tbr, 25 tbn, 25 tbc) und im avconv-Befehl zur Erstellung des Videos einzutragen.

avconv -i input.avi -codec copy output.mov ... AVI-Video in ein QuickTime-Video konvertieren

-codec copy ... Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

6. QuickTime-Video (MOV) → AVI

avconv -i input.mov -vcodec mjpeg -b:v 8000k -acodec libmp3lame -b:a 128k -ar 44100 output.avi

-i ... Name des Input-Files
-vcodec mpeg4 ... Video-Codec für output.avi
-b:v 8000k ... Video-Bitrate 8000 kBit/s

-acodec libmp3lame ... Audio-Codec libmp3lame (siehe: avconv -codecs)
-b:a 128k ... Audio-Bitrate 128 kBit/s
-ar 44100 ... Audio Samplingfrequenz 44100 Hz

7. QuickTime-Video (MOV) → MPEG4

avconv -i output.mov -vcodec mpeg4 -b:v 1024k output.mp4

-i ... Name des Input-Files
-vcodec mpeg4 ... Video-Codec für output.mp4
-b:v 1024k ... Video-Bitrate 1024 kBit/s

8. Flash-Video (FLV) → MPEG4

avconv -i output.flv -vcodec mpeg4 -b:v 1024k output.mp4

-i ... Name des Input-Files
-vcodec mpeg4 ... Video-Codec für output.mp4
-b:v 1024k ... Video-Bitrate 1024 kBit/s

9. VOB → MPEG4

avconv -i VTS_01_4.VOB -vcodec mpeg4 -b:v 2048k output.mp4

-i ... Name des Input-Files
-vcodec mpeg4 ... Video-Codec für output.mp4
-b:v 2048k ... Video-Bitrate 2048 kBit/s

Hinweis: Der Umweg über ein AVI-Format bringt mitunter bessere Resultate. Da durch die Konvertierung manchmal die Synchronisation von Video- und Audiosignal ein wenig durcheinander gerät (lippensynchrones Audiosignal).

avconv -i VTS_01_4.VOB -vcodec libx264 -ar 48000 -b:a 448k -threads 0 output.mp4

-i ... Name des Input-Files
-ar 48000 -b:a 448k ... bevorzugte Kombinationen von Samplingrate und Bitrate (-ar 48000 -b:a 192k bzw. -ar 48000 -b:a 448k)
-vcodec libx264 ... Video-Codec (H.264) für output.mp4

10. VOB → AVI → MPEG4

avconv -i VTS_01_4.VOB -codec copy output.avi

-i ... Name des Input-Files
-codec copy ... Codecs kopieren, übernehmen, ohne Reencoding

(decoding/encoding)

avconv -i output.avi -vcodec mpeg4 -b:v 2048k output.mp4

-i ... Name des Input-Files

-vcodec mpeg4 ... Video-Codec für output.mp4

-b:v 2048k ... Video-Bitrate 2048 kBit/s

11. MP4 → AVI

avconv -y -i input.mp4 -vcodec libxvid -b:v 2048k -s 960x540 -aspect 16:9 -f avi -acodec libmp3lame -ac 2 output.avi

-y ... vorhandene gleichnamige Datei überschreiben

-i ... Name des Input-Files

-vcodec libxvid ... Video-Codec für output.avi

-b:v 2048k ... Video-Bitrate 2048 kBit/s

-s 960x540 ... Bildgröße in Pixel

-aspect 16:9 ... Seitenverhältnis 16:9

-f avi ... Video-Format avi

-acodec libmp3lame ... Audio-Codec libmp3lame (siehe: avconv -codecs)

-ac 2 ... Audio-Channel 2 (Stereo); 1 (Mono); ohne Angabe - default (Mono)

12. MP4 → PAL-DVD

avconv -i input.mp4 -target pal-dvd output.mpg

-targeted pal-dvd ... MP4-Datei für die Speicherung auf einer DVD bearbeiten; die Option -targeted pal-dvd bzw. -targeted ntsc-dvd fasst einige Optionen und Parameter zusammen (siehe: nachfolgenden Befehlszeile)

avconv -i input.mp4 -c:v mpeg2video -s 720x576 -aspect 16:9 -r 25 -b:v 8000k -c:a mp2 -ac 2 -ar 48000 -b:a 192k output.mpg

-i ... Name des Input-Files

-c:v mpeg2video ... Video-Codec für output.mpg

-s 720x576 ... Bildgröße in Pixel

-aspect 16:9 ... Seitenverhältnis 16:9

-r 25 ... Framerate des Videos setzen (Frames/Sekunde)

-b:v 8000k ... Video-Bitrate 8000 kBit/s

-c:a mp2 ... Audio-Codec mp2

-ac 2 ... Audio-Channel 2 (Stereo); 1 (Mono); ohne Angabe - default (Mono)

-ar 48000 ... Sampling-Rate 48000 (sinnvolle Werte in Hz: 44100, 48000,

32000)

-b:a 192k ... Audio-Bitrate 192 kBit/s

H: Audiodaten extrahieren, entfernen, hinzufügen

1. MPEG4 → MP3 (Sound extrahieren)

avconv -i input.mp4 -vn -ar 44100 -ac 2 -b:a 128k -f mp3 output_audio.mp3

-i ... Name des Input-Files

-vn ... keine Video-Information übertragen; es wird hier nur der Sound benötigt

-ar 44100 ... Sampling-Rate 44100 (sinnvolle Werte in Hz: 44100, 48000, 32000)

-ac 2 ... Audio-Channel 2 (Stereo); 1 (Mono); ohne Angabe - default (Mono)

-b:a 128k ... Audio-Bitrate 128 kBit/s (sinnvolle Werte: 32k, 40k, 48k, 56k, 64k, 80k, 96k, 112k, 128k, 160k, 192k, 224k, 256k, 320k)

-f mp3 ... Audio-Format mp3 (Beispiele für die Option -f: flv, gif, image2, ipod, mov, mp3, mp4, mpeg, ogg, swf, vob, wav, webm)

2. AVI → MP3 (Sound extrahieren)

avconv -i input.avi -vn -ar 44100 -ac 2 -b:a 192k -f mp3 output.mp3

-i ... Name des Input-Files

-vn ... kein Videosignal

-ar 44100 ... Sampling-Rate 44100 (sinnvolle Werte in Hz: 44100, 48000, 32000)

-ac 2 ... Zwei Kanäle (stereo) ; 1 (Mono)

-b:a 192k ... Audio-Bitrate 192 kBit/s

-f mp3 ... Format mp3

3. AVI → FLAC (Sound extrahieren)

avconv -i input.avi -acodec flac output.flac

-i ... Name des Input-Files

-acodec flac ... Audio-Codec flac (siehe: avconv -codecs)

4. Audiodaten entfernen

avconv -i input.mp4 -vcodec copy -an output_stumm.mp4

-i ... Name des Input-Files

-vcodec copy ... Video-Codec kopieren, übernehmen, ohne Reencoding

(decoding/encoding)
-an ... kein Audiosignal

5. Audiodaten, Tonspur hinzufügen

avconv -i input_stumm.mp4 -i sound.mp3 -b:v 7000k -vol 20 -acodec libmp3lame -ac 2 -b:a 128k output_sound.mp4

-i ... Name der 2 Input-Files: -i video.mpg -i sound.mp3
-b:v 7000k ... Bitrate 7000 kBit/s
-vol 20 ... Mit -vol wird die Lautstärke bei zu leisen Audioquellen angehoben, wobei der Grundwert, der keine Änderung bewirkt 256 ist; ein höherer Wert hebt und ein tieferer Wert senkt die Lautstärke (-vol 20).
-acodec libmp3lame ... Audio-Codec libmp3lame (siehe: avconv -codecs)
-ac 2 ... Audio-Channel 2 (Stereo); 1 (Mono)
-b:a 128k ... Audio-Bitrate in kb/s

Hinweis: Der Codec von input_stumm.mp4 und output_sound.mp4 müssen übereinstimmen.

6. Tonspur hinzufügen, aber die Sounddatei ist länger als das Video und zu laut

avconv -shortest -i input_stumm.avi -i sound.mp3 -vcodec copy -vol 20 -acodec libmp3lame -ac 2 -b:a 128k output_sound.avi

-shortest ... hört auf, wenn das kürzeste Inputfile (Audio oder Video) endet
-i ... Name des Input-Files
-vcodec copy .. Video-Codec kopieren, übernehmen, ohne Reencoding (decoding/encoding)
-vol 20 ... Lautstärke (20 ist ziemlich leise)
-acodec libmp3lame ... Audiocodec libmp3lame
-ac 2 ... zwei Kanäle - Stereo (default Mono)
-b:a 128k ... Audio-Bitrate setzen - 128 kBit/s (default 64kBit/s)

7. Videospur und Audiospur synchronisieren (delay)

Videospur ohne Ton:

avconv -i input.mp4 -vcodec copy -an output_video.mp4

-i ... Name des Input-Files
-vcodec copy ... Video-Codec kopieren, übernehmen, ohne Reencoding (decoding/encoding)
-an ... keine Audio-Information übertragen

Audiospur ohne Bild:

```
avconv -i input.mp4 -vn -ar 44100 -ac 2 -b:a 192k -f mp3  
output_audio.mp3
```

-i ... Name des Input-Files

-vn ... keine Video-Information übertragen

-ar 44100 ... Sampling-Rate 44100 (sinnvolle Werte in Hz: 44100, 48000, 32000)

-ac 2 ... zwei Kanäle - Stereo (default Mono)

-b:a 192k ... Audio-Bitrate setzen - 192 kBit/s

-f mp3 ... Format MP3

Videospur und Audiospur zusammenfügen:

```
avconv -itsoffset 00:00:03.5 -i output_video.mp4 -i output_audio.mp3 -  
b:v 7000k -acodec libmp3lame -ac 2 -b:a 192k output.mp4
```

-itsoffset 00:00:03.5 ... Video- und Audiospur werden um 3 Sekunden und 5 Millisekunden verschoben; die Audiospur wird mehr zum Anfang des Videos verschoben

-i ... Name des Input-Files

-b:v 7000k ... Video-Bitrate setzen - 7000 kBit/s

-acodec libmp3lame ... Audio-Codec libmp3lame

-ac 2 ... zwei Kanäle - Stereo (default Mono)

-b:a 192k ... Audio-Bitrate setzen - 192 kBit/s

I: Eine oder mehrere Audiospuren mit der Option -map extrahieren (VOB, MPG)

Optionen -map:

-map 0 ... übernimmt alle Datenströme - Video- und Audiospuren

-map 0:v ... speichere alle Bildspuren (meist gibt es nur eine)

-map 0:a ... speichere alle Audiospuren

-map 0:a:(Zahl) ... speichere Audiospur (Zahl), **Hinweis:** Zählung fängt bei 0 an!

-map 0:s ... speichere alle Untertitelspuren

-map 0:s:(Zahl) ... speichere die Untertitelspur (Zahl)

-map 0 -map -0:a:(Zahl) ... speichere alle Spuren außer der Tonspur (Zahl), **Hinweis:** das Minuszeichen vor der zweiten Null nicht vergessen

-map 0 -map -0:s ... speichere alles außer den Untertitelspuren, **Hinweis:** das Minuszeichen vor der zweiten Null nicht vergessen

-map 0 -map -0:s:(Zahl) ... speichere alles außer der Untertitelspur (Zahl), **Hinweis:** das Minuszeichen vor der zweiten Null nicht vergessen

Bildschirmausgabe: avconv -i input.vob

```
avconv version 0.8.10-6:0.8.10-0ubuntu0.12.10.1, Copyright
(c) 2000-2013 the Libav developers
  built on Feb  6 2014 20:56:10 with gcc 4.7.2
[mpeg @ 0x7ccb80] max_analyze_duration reached
Input #0, mpeg, from 'input.vob':
Duration: 01:25:45.95, start: 0.049756, bitrate: 6473 kb/s
Stream #0.0[0x1e0]: Video: mpeg2video (Main), yuv420p,
720x576 [PAR 64:45 DAR 16:9], 9800 kb/s, 25 fps, 25 tbr,
90k tbn, 50 tbc
Stream #0.1[0x82]: Audio: ac3, 48000 Hz, 5.1, s16, 448 kb/s
Stream #0.2[0x81]: Audio: ac3, 48000 Hz, 5.1, s16, 448 kb/s
Stream #0.3[0x80]: Audio: ac3, 48000 Hz, 5.1, s16, 448 kb/s
At least one output file must be specified
```

Das Video input.vob enthält eine Videospur und 3 Audiospuren. Im nachfolgenden Beispiel wird die 1. Audiospur (-map 0:a:0) extrahiert und in eine MP3-Datei gespeichert.

avconv -i input.vob -map 0:a:0 -c:a libmp3lame -b:a 256k -ar 44100 -ac 2 output.mp3

-i ... Name des Input-Files
-map 0:a:0 ... 1. Audiospur
-c:a libmp3lame ... Audio-Codec libmp3lame
-b:a 256k ... Audio-Bitrate setzen - 256 kBit/s
-ar 44100 ... Audio Samplingfrequenz 44100 Hz
-ac 2 ... Zwei Kanäle (stereo); -ac 1 .. Mono

J: Audiodateien umwandeln, konvertieren

avconv -i INFILE -b:a 128k -acodec libmp3lame -ar 44100 -ac 2 ...
OUTFILE

-b:a 128k ... Audio-Bitrate in kb/s
-vn ... kein Videosignal
-acodec libmp3lame ... Audio-Codec libmp3lame
-acodec pcm_s16le ... PCM-Codec (WAV)
-f mp3 ... Format MP3
-ar 44100 ... Sampling-Rate 44100 (sinnvolle Werte in Hz: 44100, 48000, 32000)
-ac 2 ... Zwei Kanäle (stereo); -ac 1 .. Mono
... weitere Optionen

avconv -i input.mp3 output.wav ... WAV → MP3

avconv -i input.wav -b:a 160k output.mp3 ... WAV → MP3; Bitrate 160 kBit/s

avconv -i input.mp3 output.ogg ... MP3 → OGG

avconv -i input.mp3 -acodec pcm_s16le -ar 44100 -ac 2 output.wav ... MP3 → WAV

avconv -i input.flac -id3v2_version 3 output.mp3 ... FLAC → MP3; ID3-Tags - Version 2 – übernehmen

avconv -i input.flac -acodec libmp3lame -aq 5 output.mp3 ... FLAC → MP3; -aq 5 (0 ... hohe Qualität, 10 ... niedrige Qualität)

avconv -i input.avi -vn -ar 44100 -ac 2 -b:a 192k -f mp3 output.mp3 ... AVI → MP3

avconv -i input.3gp -acodec libmp3lame -ar 32000 -ac 2 -f mp3 output.mp3 ... iPhone-Audio-Format ins MP3-Format konvertieren

K: Bilddateien umwandeln, konvertieren

avconv -i input.jpg output.png ... ein JPEG-Bild in ein PNG-Bild umwandeln

avconv -i input.jpg output.tif ... ein JPEG-Bild in ein TIFF-Bild umwandeln

avconv -i input.jpg -pix_fmt rgb24 output.gif ... ein JPEG-Bild in ein GIF-Bild umwandeln; Pixelformat rgb24

avconv -i input.png output.jpg ... ein PNG-Bild in ein JPEG-Bild umwandeln

avconv -i input.png output.gif ... ein PNG-Bild in ein GIF-Bild umwandeln

avconv -i input.png output.tif ... ein PNG-Bild in ein TIFF-Bild umwandeln

L: Bilder aus ein Video extrahieren

avconv -i INFILE -ss 0:0:12.200 -t 0:0:5 ... OUTFILE

-ss 0:0:12.200 ... Start bei 12 Sekunden und 200 Millisekunden
-t 0:0:5 ... Länge 5 Sekunden (kann auch als **-t 5** angegeben werden)

Es ist wichtig, dass der Zeit-Offset **-ss 0:0:12.200** als Parameter der Ausgabedatei angegeben wird und nicht als Parameter der Eingabedatei, da mitunter Keyframe-Probleme auftreten können.

1. ein JPEG-Bild aus einem Video extrahieren – Bildgröße festlegen

avconv -an -i input.avi -ss 0:0:25 -t 0:0:0.100 -f image2 -s 800x450 thumb%d.jpg

-an ... kein Audiosignal übertragen
-i ... Name des Input-Files
-ss 0:0:25 ... Aufnahmezeitpunkt bei 25 Sekunden
-t 0:0:0.100 ... das Video läuft nur 100 Millisekunden
-f image2 ... Format image2 (JPEG)
-s 800x450 ... Größe des Bildes in Pixel (Bildformat 16:9)

2. ein JPEG-Bild aus einem Video extrahieren

avconv -y -an -i input.mp4 -vframes 1 -ss 00:00:25 -qmin 6 -f mjpeg -s 1600x900 output.jpg

-y ... vorhandene gleichnamige Datei überschreiben
-an ... kein Audiosignal übertragen
-i ... Name des Input-Files
-vframes 1 ... nur ein Video-Frame anzeigen
-ss 0:0:25 ... Aufnahmezeitpunkt bei 25 Sekunden
-qmin 6 ... qmin (quantiser scale parameter) ist ein Faktor, der die Stärke der Kompression regelt; bei MPEG-4 liegt der Faktor zwischen 1 und 31, wobei nur ganze Zahlen erlaubt sind; 1 .. sanfte Kompression und gute Qualität; 31 .. starke Kompression; optimale Werte liegen etwa bei 2 - 8
-f mjpeg ... Format MJPEG
-s 1600x900 ... Größe des Bildes in Pixel (Bildformat 16:9)

avconv -i input.avi -ss 25 -vframes 1 output.jpg ... ein JPEG-Bild aus einem Video extrahieren

3. Bilderserie aus einem Video extrahieren

avconv -i video.avi -f image2 -s 800x450 bild%06d.jpg

-i ... Name des Input-Files
-f image2 ... Format image2 (JPEG)
-s 800x450 ... Größe des Bildes in Pixel (Bildformat 16:9)
bild%06d.jpg ... Namen der Bilder - bild000001.jpeg, bild000002.jpeg, bild000003.jpeg etc.

Hinweis: Abbruch der Erstellung einer Bilderserie - Tastenkombination [Strg] + [C].

M: Video aus Einzelbilder erstellen

Zunächst müssen die Bilder von 1 an aufsteigend nummeriert werden, damit avconv auch die richtigen Bilder verwendet.

1. In einem Terminal in das Verzeichnis mit den Bildern wechseln und folgende kurze Zeile eingeben:

```
x=1; for i in *.jpg; do counter=$(printf %04d $x); mv "$i"
picture_"$counter".jpg; x=$((x+1)); done
```

Der Befehl benennt alle JPEG-Bilder (Endung .jpg) in picture_0001.jpg, picture_0002.jpg, picture_0003.jpg ... um.

Achtung: Mitunter muss das 1. Bild und das letzte Bild verdoppelt werden, das es sonst zu kurz oder gar nicht im Video angezeigt wird (Name des 1. Bildes picture_0000.jpg).

2. Video aus den Einzelbildern erstellen

```
avconv -r 0.07 -f image2 -i picture_%04d.jpg -vcodec libx264 -r 20
video.mp4
```

-r 0.1 ... jedes Bild wird ca. 10 Sekunden angezeigt

-r 0.3 ... jedes Bild wird ca. 3 Sekunden angezeigt

-r 0.01 ... jedes Bild wird ca. 100 Sekunden angezeigt

Der Parameter -r beschreibt hier die Rate mit der die Bilder im Video angezeigt werden, je größer die Zahl, desto schneller ist das Video vorbei und demzufolge je kleiner -r ist, umso langsamer läuft die Dia-Show ab.

-r 0.07 ... jedes Bild wird ca. 15 Sekunden angezeigt

-f image2 ... Format image2 (JPEG)

-i ... Name des Input-Files
-vcodec libx264 ... Videocodec libx264
-r 20 ... Framerate des Videos setzen (Frames/Sekunde)

Hinweis: Mit der Option **-s** Bildbreite x Bildhöhe (Angaben in Pixel) kann eine von der Originalgröße der Einzelbilder abweichende Bildgröße angegeben werden (z.B. **-s 640x480**).

3. Anschließend kann das Video von den verdoppelten Bildern befreit werden.

Falls das 1. und das letzte Bild im Video zu lange angezeigt wird, kann die entsprechende Videosequenz um einige Sekunden gekürzt werden.

avconv -i video.mp4 -ss 0:0:10 -codec copy video_short.mp4 ... vom Video die ersten 10 Sekunden entfernen

avconv -y -i video.mp4 -t 0:5:34 -codec copy video_short.mp4 ... vom Video (Länge 5 Minuten und 44 Sekunden) die letzten 10 Sekunden entfernen

N: Video aus einem einzelnen Bild erstellen

avconv -loop 1 -i image.png -vcodec libx264 -r 25 -t 30 video.mp4

-loop ... Endlosschleife für die Dauer von 30 Sekunden (**-t 30**)
-i ... Name des Input-Files
-vcodec libx264 ... Videocodec libx264
-r 25 ... Framerate des Videos setzen (Frames/Sekunde)
-t 30 ... Länge des Videos in Sekunden

Hinweis: Nun kann dem Video eine Tonspur hinzugefügt werden.

avconv -shortest -i video.mp4 -i sound.mp3 -vcodec copy -vol 256 -acodec libmp3lame -ac 2 -b:a 128k video_sound.mp4

O: HTML 5 -konformes Video erstellen

Mit dem nachfolgenden **avcov**-Kommando wird ein HTML 5 -konformes Video erstellt.

avconv -i INPUT-FILE -pix_fmt yuv420p -c:v libx264 -crf 30 -preset slower -profile:v Main -level 31 -s 568x320 -c:a aac -ar 22050 -ac 1 -b:a

64k -strict experimental out.mp4 ; qt-faststart out.mp4 OUTPUT-FILE.mp4; rm out.mp4

-i ... Name des Input-Files (MP4, AVI, FLV, WEBM, VOB, MOV)
-s 568x320 ... Bildgröße des Videos
-ar 22050 ... Sampling-Rate 22050 Hz
-ac 1 ... Mono
-b:a 64k ... Audio-Bitrate 64 kBit/s

Durch qt-faststart werden die Metadaten an den Anfang des Videos eingefügt. Dadurch können nach dem Download der ersten Datenblöcke der Videostream bereits angezeigt werden.

P: Farbbalken entfernen, hinzufügen

1. Crop - schwarze Balken von dem Videos entfernen

avconv -i video_720x480.mp4 -vf crop=660:300:70:80 -vcodec mpeg4 -b:v 2048k output.mp4

-vf crop=660:300:70:80 ... Zielformat des Videos ist 660x300; oberer Balken von 80 Pixel wird entfernt; linker Balken von 70 Pixel wird entfernt; Nullpunkt des neuen Bildes ist 70 Pixel nach rechts und 80 Pixel nach unten verschoben

2. Pad – farbigen Balken zum Video hinzufügen

Das Video wird durch das Einfügen von schwarze Balken in ein Seitenverhältnis von 4:3 gebracht.

Originalbreite des Videos: 660 Pixel

Originalhöhe des Videos: 300 Pixel

Zielverhältnis: Breite x Höhe: 4:3

Zielbreite: $660:4=165$... Breite des Videos bleibt erhalten

Zielhöhe: $165*3=495$

Balkenhöhe: $495-300/2=97$.. der obere und untere Balken hat eine Höhe von jeweils 97 Pixel

660:495 ... Zielgröße des Videos (mit den Balken oben und unten)

0 ... kein Versatz in der x-Achse (horizontal)

97 ... 97 Pixel Versatz in der y-Achse (oberer Balken)
0xff0000 ... Farbwert des Balkens (rot) in der hexadezimalen Schreibweise
(0x000000 ... schwarz)

avconv -i video.mp4 -vf pad=640:480:0:104:0xff0000 -b:v 2048k -r 25 -acodec libmp3lame -ac 2 -b:a 192k video_new.mp4

-i ... Name des Input-Files
-vf pad=640:480:0:104:0xff0000 ... Videoformat verändern
-b:v 2048k ... Video-Bitrate 1024 kBit/s
-r 25 ... Framerate des Videos setzen (Frames/Sekunde)
-acodec libmp3lame ... Audio-Codec libmp3lame
-ac 2 ... Zwei Kanäle (stereo); -ac 1 .. Mono
-b:a 192k ... Audio-Bitrate setzen - 192 kBit/s

Q: Bildausschnitt aus einem Film herauschneiden

1. Videoausschnitt: Startpunkt bei xx Minuten bis Ende

avconv -i VTS_01_4.VOB -ss 0:13:00 -codec copy output.vob

-i ... Name des Input-Files
-ss 0:13:00 ... Video-Ausschnitt: Start bei 13 Minuten bis Ende
-codec copy ... Codex kopieren, übernehmen, ohne Reencoding
(decoding/encoding)

2. Videoausschnitt: Startpunkt und Endpunkt festlegen

avconv -i VTS_01_4.VOB -ss 0:13:00 -t 0:00:05 -codec copy output.vob

bzw.

avconv -i VTS_01_4.VOB -ss 0:13:00 -t 0:00:05 -vcodec mpeg4 -b:v 2048k output.mp4

bzw.

avconv -i VTS_01_4.VOB -ss 0:13:00 -t 0:00:05 -vcodec copy -acodec copy output.avi

-ss 0:13:00 ... Video-Ausschnitt: Start bei 13 Minuten
-t 0:00:05 ... Länge des Video-Ausschnitts: 5 Sekunden

-ss 0:0:12.200 ... Video-Ausschnitt: Start bei 12 Sekunden und 200 Millisekunden

-t 0:0:5 ... Länge des Video-Ausschnitts: 5 Sekunden (kann auch als **-t 5** angegeben werden)

-vcodec copy ... Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

-acodec copy ... Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

-codec copy ... Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

R: Videos zusammenfügen, verbinden

avconv -i concat:file1.mp4|file2.mp4 -codec copy output.mp4

avconv -i "concat:file1.avi|file2.avi|file3.avi" -codec copy output.avi

-i concat:file1|file2 ... die Videos miteinander verbinden; der Backslash (\) vor dem geraden Strich (|) ist erforderlich oder der concat-Parameter ist in Hochkammata einzuschließen

-codec copy ... Codecs kopieren, übernehmen, ohne Reencoding (decoding/encoding)

Hinweis: Videos mit H.264/AAC-Codec, müssen erst in einen TS-Container überführt werden (siehe: man avconv). Alternativ können die Videos mit H.264/AAC-Codec auch mit dem Videoschnittprogramm OpenShot zusammengefügt werden.

S: Bitrate, Framerate und Bildgröße eines HD-Films reduzieren

avconv -i INFILE -b:v 5000k -r 25 -s 1280x720 ... OUTFILE

avconv -i input.mp4 -b:v 5000k -r 25 -s 1280x720 output.mp4

-b:v 5000k ... Video-Bitrate auf 5000 kb/s reduzieren

-r 25 ... Frame-Rate auf 25 Bilder pro Sekunde reduzieren

-s 1280x720 ... Bildgröße auf 1280x720 Pixel reduzieren; 640x360, 960x540, hd720 (1280x720), hd1080 (1920x1080)

... weitere Optionen

Hinweis: Falls es zu Fehlermeldungen kommt, so liegt dies meistens an einer falschen Angabe für die Bildgröße.

T: Bildschirm-Aufzeichnung

Mit avconv ist es möglich, den Inhalt eines X11-Bildschirms aufzunehmen. Für die Bildschirmaufzeichnung gilt eine veränderte Syntax:

```
avconv -f x11grab -i HOSTNAME:DISPLAY.BILDSCHIRM-  
Nummer[+X-b:astand,Y-b:astand] VIDEO-OPTIONEN AUSGABEDATEI
```

**avconv -f x11grab -r 25 -s 1920x1080 -i :0.0 -vcodec libx264 -
screencast.flv** ... die Aufnahme erfolgt mit Display-Nummer.Bildschirm-
Nummer 0.0 und dem Videocodec libx264; die Aufnahme wird mit der
Tastenkombination [Strg] + [C] beendet

**avconv -f x11grab -r 25 -s 640x300 -i :0.0+10,200 -b:v 2048k
output.mp4**

-f x11grab ... Format x11grab

-r 25 ... Frame-Rate auf 25 Bilder pro Sekunde reduzieren

-s 640x300 ... Größe des Bildes in Pixel

-i :0.0+10,200 ... Nullpunkt des Videos - 10 Pixel von links und 200 Pixel
von oben

-b:v 2048k ... Video-Bitrate 2048k/s

Hinweis: Mit dem Terminalprogramm xwininfo kann die Bildgröße des
aufzunehmenden Programmfensters ermittelt werden.

U: Video-Untertitel einfügen, fest »einbrennen« in die Videodatei

Für die Untertitel muss eine Textdatei mit der Endung .srt erstellt werden.
Erhält die Datei mit den Untertiteln denselben Namen wie das Video, so
erkennen einige Player (Mplayer) die Untertitel-Datei automatisch (Datei
muss sich in denselben Verzeichnis wie das Video befinden).

Bei den anderen Playern (VLC) muss die Datei mit den Untertiteln über das
Menü erst ausgewählt werden.

Untertitelformat SubRip (.srt)

Einfaches Textformat:

```
1  
00:00:10,500 --> 00:00:13,000  
Elephant's Dream
```

2
00:00:15,000 --> 00:00:18,000
At the left we can see...

Textformat mit Angabe der Position und Formatierung des Untertitels im Video:

1
00:00:10,500 --> 00:00:13,000 X1:63 X2:223 Y1:43 Y2:58
<i>Elephant's Dream</i>

2
00:00:15,000 --> 00:00:18,000 X1:53 X2:303 Y1:438 Y2:453
At the left we can see...

Voraussetzung: Eine Video-Datei mit Audiospur (input.mp4) und eine Datei mit den Untertiteln (subtitle.srt).

1. Videodatei und Audosignal trennen

```
avconv -i input.mp4 -vcodec copy -an output_video.mp4
```

```
avconv -i input.mp4 -vn -ar 44100 -ac 2 -b:a 192k -f mp3  
output_audio.mp3
```

Hinweis: Die Audio-Bitrate (-b:a 192k) sollte in etwa der Größe des Ursprungssignals entsprechen oder kleiner sein (avconv -i input.mp4).

2. Untertitel in den Videostream fest »einbrennen«

```
mplayer -benchmark -nolirc -nojoystick -nosound -noframedrop -  
really-quiet -vo yuv4mpeg:file=/dev/stdout output_video.mp4 | avconv -  
i - -vcodec libx264 -b:v 8000k -an output_subtitle.mp4
```

Hinweis: Unterscheiden sich der Basisname (Dateiname ohne Endung) der Videodatei und der Dateiname der Datei mit den einzufügenden Untertiteln, so ist der Terminalbefehl zu ergänzen (-sub subtitle.srt; **siehe auch:** man mplayer, mplayer -vo help).

```
mplayer -benchmark -nolirc -nojoystick -nosound -noframedrop -sub  
subtitle.srt -really-quiet -vo yuv4mpeg:file=/dev/stdout  
output_video.mp4 | avconv -i - -vcodec libx264 -b:v 8000k -f mp4 -an  
output_subtitle.mp4
```


Die Videobitrate (-b:v 8000k) sollte in etwa der Größe des Ursprungssignals entsprechen oder kleiner sein (avconv -i input.mp4, -vcodec libx264 → MP4, -vcodec libxvid → AVI, MP4).

3. Video und Audiosignal zusammenfügen

avconv -i output_subtitle.mp4 -i output_audio.mp3 -vcodec copy -acodec libmp3lame -ac 2 -b:a 192k output.mp4

Beispiel: AVI-Video mit Audiosignal

I: Ausgangs-Video

avconv -i urlaubs_video.avi ... Informationen über das Video anzeigen

Name: urlaubs_video.avi

Größe: 7,2 GByte

Bildgröße: 960x720 Pixel

Seitenverhältnis: 4:3

Video-Länge: 01:33:38.41 (Stunden:Minuten:Sekunden.Millisekunden)

Video-Bitrate: 10202 KBit/s

Audio-Bitrate: 192 Kbit/s

Datei mit den Untertiteln: urlaubs_video_untertitel.srt

II: Video ohne Audosignal erstellen

avconv -i urlaubs_video.avi -vcodec copy -an output_video.avi

Name: output_video.avi

Größe: 7 GByte

Bildgröße: 960x720 Pixel

Seitenverhältnis: 4:3

Video-Länge: 01:33:38.37 (Stunden:Minuten:Sekunden.Millisekunden)

Video-Bitrate: 10004 KBit/s

III: Audosignal von der Videodatei trennen

avconv -i urlaubs_video.avi --vn -acodec libmp3lame -b:a 192k -ar 44100 -ac 2 output_audio.mp3

Name: output_audio.mp3

Größe: 134,9 MByte

Audio-Länge: 01:33:38.41 (Stunden:Minuten:Sekunden.Millisekunden)

Audio-Bitrate: 192 KBit/s

IV: Untertitel in den Videostream fest »einbrennen«

```
mplayer -benchmark -nolirc -nojoystick -nosound -noframedrop -sub  
urlaubs_video_untertitel.srt -really-quiet -vo yuv4mpeg:file=/dev/stdout  
output_video.avi | avconv -i - -vcodec libxvid -b:v 9000k -f avi -an  
output_subtitle.avi
```

Die Video-Bitrate des bearbeiteten Videos mit den fest »eingebrennten«
Untertiteln wird auf 9000 KBit/s reduziert.

Name: output_subtitle.avi

Größe: 5,4 GByte

Bildgröße: 960x720 Pixel

Seitenverhältnis: 4:3

Video-Länge: 01:18:06.75 (Stunden:Minuten:Sekunden.Millisekunden)

Video-Bitrate: 9000 KBit/s

MPlayer in Kombination mit avconv haben in diesem Beispiel die
Videolänge etwas verkürzt. Um die Videolänge an das Audiosignal wieder
anzupassen, ist der avconv-Befehl um die Option -setpts zu erweitern.

Video-Länge des ursprünglichen Videos: 01:33:38.37

(Stunden:Minuten:Sekunden.Millisekunden)

Video-Länge des bearbeiteten Videos: 01:18:06.75

(Stunden:Minuten:Sekunden.Millisekunden)

Videolänge in Sekunden:

Duration: 01:33:38.37 -> 5618 Sekunden

Duration: 01:18:06.74 -> 4686 Sekunden

Parameter für die Option -setpts ermitteln: $5618 / 4686 = 1.19889$

```
avconv -i output_subtitle_2.avi -vf setpts=1.19889*PTS -vcodec libxvid -  
b:v 9000k -f avi -an output_subtitle_1.avi
```

bzw.

```
mplayer -benchmark -nolirc -nojoystick -nosound -noframedrop -sub  
subtitle.srt -really-quiet -vo yuv4mpeg:file=/dev/stdout
```

output_subtitle_2.avi | avconv -i - -vf setpts=1.19889*PTS -vcodec libxvid -b:v 9000k -f avi -an output_subtitle_1.avi

Name: output_subtitle_1.avi

Größe: 5,4 GByte

Bildgröße: 960x720 Pixel

Seitenverhältnis: 4:3

Video-Länge: 01:33:38.87 (Stunden:Minuten:Sekunden.Millisekunden)

Video-Bitrate: 9000 KBit/s

Anmerkung: Die 2. Variante mit mplayer sollte bevorzugt werden, da ein zusätzliches Reencoding vermieden wird. Jedes zusätzliche Reencoding des Videos verbessert nicht unbedingt die Qualität des Ergebnisses.

V: Video und Audiosignal zusammenfügen

avconv -shortest -i output_subtitle_1.avi -i output_audio.mp3 -vcodec copy -acodec libmp3lame -ac 2 -b:a 192k urlaubs_video_sub_de.avi

Name: urlaubs_video_sub_de.avi

Größe: 5,4 GByte

Bildgröße: 960x720 Pixel

Seitenverhältnis: 4:3

Video-Länge: 01:33:38.75 (Stunden:Minuten:Sekunden.Millisekunden)

Video-Bitrate: 9000 KBit/s

Audio-Bitrate: 192 Kbit/s

OpenShot - eine Alternative:

Alternativ kann die Videolänge auch über das Videoschnitt-Programm OpenShot (grafische Oberfläche) angepasst werden.

1. Video output_subtitle.avi importieren
2. Video mit der Maus auf eine Videospur ziehen

aktuelle Videolänge: 4686 Sekunden (01:18:06)

gewünschte Videolänge: 5618,00 Sekunden (01:33:38)

Geschwindigkeit (Einstellwert für OpenShot): $4686 / 5618 = 0,834104$
→ 0.84

Clip-Eigenschaften (rechte Maustaste): output_subtitle.avi (ohne Audiosignal)

Audio: deaktivieren

Position/Dauer: 0,00 bis 5618,00 Sekunden (Videolänge: 01:33:38)
Geschwindigkeit: 0.84 (**Hinweis:** Genauigkeit der Geschwindigkeitseinstellung nur 2 Stellen nach dem Komma)

Export des Videos:

Dateiname: output_subtitle_1.avi
Profil: Alle Formate
Ziel: AVI (h.264)
Video-Profil: 768x576 4:3 PAL
Qualität: Mittel

Video output_subtitle_1.avi mit avconv kürzen:

```
avconv -i output_subtitle_1.avi -ss 0:00:03 -t 1:32:55 -vcodec libxvid -b:v 5000k -an output_subtitle_2.avi
```

Das Video das von OpenShot erstellt wurde, muss in diesem Beispiel noch gekürzt werden. Am Anfang des Videos werden 3 Sekunden und am Ende des Videos werden etwa 40 Sekunden entfernt. Die Kürzung war in diesem Beispiel notwendig, da die Geschwindigkeitseinstellung in OpenShot nur eine Genauigkeit von 2 Stellen nach dem Komma erlaubt.

Video und Audiosignal mit avconv zusammenfügen:

Videolänge von output_subtitle_2.avi: 01:32:55.00
(Stunden:Minuten: Sekunden.Millisekunden)

Der MP3-Datei output_audio.mp3 wurde mit Audacity (Audacity: Programm mit grafischer Oberfläche) gekürzt. Der Schluss des Audiosignals wurde sanft ausgeblendet und das Audiosignal wurde wieder als MP3-Datei exportiert (output_audio_1.mp3). Da die bearbeitete MP3-Datei eine etwa 2 Sekunden längere Spielzeit als das Videosignal hat, wird das letzte Bild im Video etwa 2 Sekunden länger angezeigt.

Länge des Audiosignals (output_audio_1.mp3): 01:32:57.35
(Stunden:Minuten: Sekunden.Millisekunden)

```
avconv -i output_subtitle_2.avi -i output_audio_1.mp3 -vcodec copy -acodec libmp3lame -ac 2 -b:a 128k urlaubs_video_sub_de_1.avi
```

V: Webcam - Aufzeichnen eines Videos

Mit dem Parameter `-t` wird die Aufnahmedauer definiert und `avconv` beendet die Aufnahme selbstständig nach Ablauf der festgelegten Zeit. Wird keine Aufnahmedauer übergeben oder soll die Aufnahme vorzeitig beendet werden, muss man nur die Tastenkombination `[Strg] + [C]` betätigen. Die Gerätedatei der Webcam wird durch `-i` angegeben und ist in der Regel `/dev/video0`.

```
avconv -y -f video4linux2 -i /dev/video0 -vcodec mpeg4 -b:v 4000k -r 15  
-s 640x480 -t 0:0:30 output.mp4
```

`-y ...` vorhandene gleichnamige Datei überschreiben
`-f video4linux2 ...` Format des Videos
`-i ...` Name des Input-Gerätes
`-vcodec mpeg4 ...` Video-Codec für output.mp4
`-b:v 4000k ...` Video-Bitrate setzen (in Bit/s)
`-r 15 ...` Framerate des Videos setzen (Frames/Sekunde)
`-s 640x480 ...` Bildgröße 640x480 Pixel (320x240, 640x480, vga, svga)
`-t 0:0:30 ...` Länge des Video-Ausschnitts: 30 Sekunden (kann auch als `-t 30` angegeben werden)

Ergänzung: Dasselbe kann man auch mit MPlayer in Kombination mit `avconv` erreichen. Mitunter liefert diese Variante die besseren Bilder.

mplayer tv:// ... Webcam mit MPlayer starten

```
mplayer -benchmark -nolirc -nojoystick -nosound -noframedrop -  
nosub -really-quiet -vo yuv4mpeg:file=/dev/stdout tv:// | avconv -y -i - -  
vcodec mpeg4 -b:v 4000k -f mp4 -t 30 -an output.mp4
```

Anmerkung:

Alternative Programme für die Videobearbeitung mit grafischer Oberfläche sind OpenShot, Kino, Kdenlive, Open Movie Editor, Cinelerra, MainActor (kostenpflichtig). Open Movie Editor, Cinelerra und MainActor sind i.d.R. nicht Bestandteil der offiziellen Paketquellen.

siehe auch: man `avconv`, man `libav-tools`, `mplayer`, `lsdvd`



Mar 9 12:06:08 tux1 kernel: [5291.441161] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=222.122.213.35 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11304 DF PROTO=TCP SPT=80
DPT=35163 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.448091] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=91.203.96.253 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11305 DF PROTO=TCP SPT=80
DPT=40330 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.448514] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=202.131.28.25 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11306 DF PROTO=TCP SPT=80
DPT=36272 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.449475] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=116.126.142.18 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11307 DF PROTO=TCP SPT=80
DPT=56965 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.478222] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=222.122.213.35 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11308 DF PROTO=TCP SPT=80
DPT=35162 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.487507] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=116.126.142.18 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11310 DF PROTO=TCP SPT=80
DPT=56964 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.488245] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=116.126.142.18 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11311 DF PROTO=TCP SPT=80
DPT=56966 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 12:06:08 tux1 kernel: [5291.488954] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=125.209.226.147 DST=10.249.26.170 LEN=52
TOS=0x00 PREC=0x00 TTL=61 ID=11312 DF PROTO=TCP SPT=80
DPT=47574 WINDOW=32851 RES=0x00 ACK URGP=0

Mar 9 13:55:07 tux1 kernel: [11823.505981] [UFW BLOCK] IN=ppp0
OUT= MAC= SRC=217.160.153.118 DST=10.249.26.170 LEN=40
TOS=0x00 PREC=0x00 TTL=253 ID=41515 DF PROTO=TCP SPT=443
DPT=55725 WINDOW=0 RES=0x00 ACK RST URGP=0

B

basename

Den <Dateipfad> ohne führende Verzeichnisse ausgeben. Wenn angegeben, auch den SUFFIX entfernen. Der Befehl basename ist besonders in Shellskripte hilfreich.

```
basename <Dateipfad> [SUFFIX]
```

Beispiele:

```
for i in $(ls -1 *.JPEG);do mv "$i" $(basename "$i" .JPEG).jpg;done
```

Ersetzung der Endung .JPEG aller Bild-Dateien des aktuellen Verzeichnisses durch die üblichere Endung .jpg.

```
for j in $(find $HOME -iname "*.mp3");do echo "$(basename $  
{}))";done
```

Zeigt alle MP3-Dateien an, ohne die Angabe des Dateipfades. Die globale Variable \$HOME enthält das Home-Verzeichnis des aktuellen Benutzers.

Anmerkung: Mit `${i%.JPEG}.jpg` statt `$(basename "$i" .JPEG).jpg` kann man dasselbe erreichen.

siehe auch: Anhang: Einführung in die Shellprogrammierung

bc

Mit bc können an der Kommandozeile Berechnungen ausgeführt werden. Nach Eingabe von bc gibt das Programm eine kurze Versionsinformation aus und wartet auf ihre Eingaben. Durch Eingabe von **quit** wird das Programm beendet.

Das Programm bc ist besonders innerhalb von Shellskripten nützlich.

+ Addition

- Subtraktion

/ Division

* Multiplikation

Dezimalstellen werden als Punkt eingeben, also 5.245

Klammerberechnungen: z.B. $5*((2+1)*(7-3))$

Mit **scale** wird die Genauigkeit festgelegt. Mit **scale=3** wird die Genauigkeit auf 3 Nachkommastellen begrenzt. Ohne Angabe von **scale** wird die Vorgabeeinstellung benutzt, **scale=0** (Null Nachkommastellen, z.B. $3/5=0$ bei **scale=0**; $3/5=0.600$ bei **scale=3**).

echo "scale=2; (500-480)/2" | bc ... die Ausgabe erfolgt direkt im Kommandozeilenfenster ohne den Umweg über den bc-Prompt; diese Schreibweise wird vorwiegend in Shellskripte verwendet z.B.:
BREITEN_QUOTIENT_A=\$(echo "scale=0; (\${ZIELBREITE}*100/\${BILDBREITE_A})+1" | bc)

siehe auch: Anhang: Einführung in die Shellprogrammierung

BIOS

siehe auch: EFI

blkid

Mit blkid können alle dem System bekannten UUID's (Universally Unique Identifiers) ermittelt werden.

sudo blkid /dev/sda1 ... UUID der Partition /dev/sda1 anzeigen
sudo blkid ... alle UUID's die dem System bekannt sind anzeigen oder
ls -l /dev/disk/by-uuid ... alle UUID's die dem System bekannt sind anzeigen

```
karl@tux1 ~ $ sudo blkid
[sudo] password for karl:
/dev/sda1: UUID="02ddc64d-9bc3-4cc7-825c-67148796f413"
TYPE="ext4"
/dev/sda5: UUID="1bbb91d9-1ef1-49ab-ad49-9adb1fc9635b"
TYPE="ext4"
/dev/sda6: UUID="bd461eb0-b31f-4125-b8e3-97a0fe0bfab6"
TYPE="swap"
```

siehe auch: man blkid, mount

cal bzw. ncal

Erstellt einen aktuellen Monatskalender, per Default wird der Sonntag als erster Wochentag ausgegeben.

ncal -M ... erstellt einen aktuellen Monatskalender, mit Montag (M steht für monday) als erster Wochentag

ncal -M -3 ... erstellt einen aktuellen Monatskalender, mit Montag als erster Wochentag, zusätzlich wird der Vor- und Folgemonat ausgegeben

cal 6 2005 ... zeigt den Juni des Jahres 2005; **Hinweis:** cal 4 04 zeigt **nicht** den April des Jahres 2004, sondern den April des Jahres 04 n.Chr.

cal -y ... zeigt die 12 Monate des aktuellen Jahres

cal -j 09 2003 ... offenbart das z.B. der 15. September der 258. Tag des Jahres 2003 ist

siehe auch: man cal

cat

Mit cat kann der Inhalt von Textdateien am Bildschirm angezeigt werden oder auch Dateistücke die vorher mit split erstellt wurden, wieder miteinander verkettet werden.

cat <Dateiname> ... der Inhalt einer Textdatei wird am Bildschirm ausgegeben

cat -n <Dateiname> ... Zeilennummerierung - die Zeilen der Datei werden nummeriert ausgegeben

Hinweis: Mit der Tastenkombination [Shift] + [Bild hoch] oder [Shift] + [Bild runter] kann innerhalb der Bildschirmausgabe Seitenweise geblättert werden.

cat -v <Dateiname> ... zeigt auch die nicht ausdrückbaren Zeichen an, wie z.B. die Zeilenumbrüche von unter Windows erstellten Dateien (dargestellt in der folgenden Notation: ^M); die verschiedenen Betriebssysteme benutzen für das Zeilenende jeweils ein anderes **nicht** sichtbares Steuerzeichen (Linux \n; MAC \r; Windows \r\n); mit den Terminalprogrammen fromdos (Windows/DOS → Linux) und todos (Linux → Windows/DOS) können die Zeilenumbrüche von Textdateien konvertiert werden

cat kleinedatei* > grossedatei ... die Dateistücke (kleinedatei_a, kleinedatei_b, ...) die vorher mit **split** zerteilt wurden, werden wieder

zusammengesetzt (* ist ein so genanntes Wildcard-Zeichen, es steht für beliebige Zeichen)

cat <Dateiname1> > <Dateiname2> ... der Inhalt von Dateiname1 wird nach Dateiname2 umgeleitet. Existiert <Dateiname2> nicht, so wird die Datei angelegt. Existiert <Dateiname2> so wird der gesamte Inhalt überschrieben (**Hinweis:** mit >> wird der Inhalt von <Dateiname1> an den Inhalt von <Dateiname2> angefügt).

cat 1.dat 2.dat 3.dat > file.dat ... fügt die 3 Dateiteile zu einer Datei zusammen

cat {1..3}.dat > file.dat ... fügt die 3 Dateiteile zu einer Datei zusammen; Kurzschreibweise

cat filename{1..25}.dat > file.dat ... fügt die 25 Dateiteile zu einer großen Datei zusammen; Kurzschreibweise

siehe auch: split, tac, mp3splt, mp3wrap, man zcat

cd <MeinVerzeichnis>

Directory, Verzeichnis wechseln (change directory) (z.B. cd /etc).

cd ohne Parameter wechselt ins Home-Verzeichnis.

cd wechselt eine Verzeichnisebene höher bzw.

cd ../ ... wechselt eine Verzeichnisebene höher

cd ../../ ... wechselt zwei Verzeichnisebenen höher

cd ~ ... wechselt ins Home-Verzeichnis

cd / ... wechselt ins Wurzelverzeichnis

CDs und DVDs erstellen - Brennen von der Kommandozeile

Sämtliche Programme mit grafischen Frontends basieren auf einer Handvoll von Kommandozeilenprogrammen, die man auf die Schnelle auch einmal in einer Konsole testen kann. Die Basis ist zunächst das Programm cdrecord. Es muss innerhalb einer Shell mit Root-Rechten gestartet werden.

Beispiel: Um eine Sicherung von Ihrem Heimatverzeichnis zu starten, verwenden Sie zuvor das Programm mkisofs:

**mkisofs -o /tmp/sicherung.iso -r -T -V Sicherung /home/<Benutzer>
sudo cdrecord dev=/dev/sr0 speed=48 /tmp/sicherung.iso**

Geschwindigkeit und Device-Bezeichnung (**hier:** /dev/sr0; **siehe auch:** mount bzw. blkid) ist entsprechend der aktuell verwendeten Hardware (DVD-Laufwerk) anzupassen.

Auf diese Weise wird zunächst eine Abbilddatei erstellt (ein sogenanntes ISO-Image), das anschließend mit `cdrecord` auf einen Rohling geschrieben wird.

Eine komplette CD kann mit folgenden Befehlen 1:1 kopiert werden:

```
dd if=/dev/sr0 of=/tmp/copyiso.img; sudo cdrecord dev=/dev/sr0  
speed=48 /tmp/copyiso.iso
```

An die Stelle von `cdrecord` treten beim Brennen von DVDs die `dvd+rw-tools`. Der Name `+RW` ist ein wenig irreführend, da die Tools mittlerweile sämtliche Spielarten von Rohlingen (`+/-R(W)`) beschreiben können.

Ein größeres Verzeichnis wird mit folgendem Befehl auf DVD gesichert:

```
sudo growisofs -R -J -Z /dev/sr0 /home/<Benutzername>
```

Haben Sie ein DVD-ISO aus dem Internet auf Ihren Rechner heruntergeladen, so können Sie dieses mittels

```
sudo growisofs -dvd-compat -Z /dev/sr0=<ISO-Name>
```

auf den Rohling bringen. Der Parameter `dvd-compat` sorgt für eine maximale Kompatibilität mit handelsüblichen DVD-ROM-Laufwerken.

siehe auch: `dd`, `mkisofs`, `man cdrecord`, `man growisofs`

chpasswd

`chpasswd` dient zum Erstellen und Ändern der Passwörter von Benutzerkonten im **Batchbetrieb**. Der Befehl liest eine Liste der Benutzer:Passwort-Paare von der **Standard-Eingabe** (interaktiv am Terminal oder Umleitung einer Datei zur Standard-Eingabe) und verwendet diese Informationen um die Passwörter der Benutzer-Konten zu ändern. Das Eingabeformat besteht aus je einer Zeile mit diesen Angaben

Benutzername:Passwort

Es wird der Hash-Algorithmus der in `/etc/pam.d/chpasswd` bzw. `/etc/pam.d/common-password` festgelegt ist (Ubuntu: `sha512`) verwendet. Wenn dort kein Algorithmus konfiguriert ist, wird der Standard-Algorithmus `DES` verwendet. Mögliche Hash-Algorithmen sind `DES`, `MD5`, `Blowfish`, `SHA-256` oder `SHA-512`.

chpasswd [Optionen]

sudo chpasswd -S

ben:eifelturm

Optionen:

-S ... das verschlüsselte Passwort wird in der Standardausgabe zusätzlich angezeigt

[Strg] + [C] ... Befehlsabbruch

siehe auch: man chpasswd, passwd, Umleitung an ein Programm

cclive

Mit cclive können Videos von Youtube und einigen anderen Anbieter (siehe: man cclive) heruntergeladen werden. Um mehrere Videos in einem Zuge herunterzuladen, trägt man besten in eine Textdatei (url.lst) je Zeile eine URL ein und übergibt mit **cat** den Inhalt der Datei an **cclive**. Ohne Angabe des Formats lädt cclive die Datei im FLV-Format herunter. Die Bezeichnungen für die Formate weichen bei den einzelnen Anbieter voneinander ab (**siehe:** man cclive). Im Zweifelsfall übergibt man an cclive das Stream-Argument **-s best**.

Über die Option **exec** kann der Download auch direkt an ein anderes Programm übergeben werden (z.B. an **avconv** zur Konvertierung des Dateiformates).

Neben dem reinen Download bietet cclive auch die Möglichkeit, Videos als Stream an den lokalen Player wie Mplayer oder VLC zu übergeben.

cclive [options] ... [URL] [URL] [URL] ...

Beispiele:

cclive "http://www.youtube.com/watch?v=4OdrJWtAKeo" ... Datei wird im FLV-Format heruntergeladen

cclive -s best "http://www.youtube.com/watch?v=4OdrJWtAKeo" ... Datei wird im bestmöglichen Format (Media Stream) heruntergeladen

cclive -S "http://www.youtube.com/watch?v=4OdrJWtAKeo" ... mögliche Media-Stream-Formate für die angegebene URL ermitteln; die ermittelten Formate können dann für die Stream-Angabe (**-s**) des eigentlichen Downloads verwendet werden

cclive --no-download "http://www.youtube.com/watch?v=4OdrJWtAKeo" ... kein Download; Ausgabe von Details zur Media-Datei

cat url.lst | cclive ... Dateien die in der Textdatei url.lst (je Zeile eine Adresse; http://www.youtube.com/watch?v=4OdrJWtAKeo) aufgelistet sind, werden im FLV-Format heruntergeladen.

Beispiel:

1. Unterstützte Media-Stream-Formate ermitteln

cclive -S http://www.youtube.com/watch?v=KGle6_3S-KM

fnt05_240p ... FLV (H263) 240x400 Pixel
fnt17_144p ... 3gp - 176x144 Pixel
fnt18_360p ... mp4 (H264) - 480x360 Pixel
fnt22_720p ... mp4 (H264) - 1280x720 Pixel
fnt43_360p ... WebM-Video (VP8) 480x360 Pixel
fnt45_720p ... WebM-Video (VP8) 1280x720 Pixel

2. Video im Format fnt22_720p (MP4) herunterladen

cclive -s fnt22_720p http://www.youtube.com/watch?v=KGle6_3S-KM

siehe auch: man cclive

chmod

chmod vergibt Rechte (change mode) - z.B. macht 755 eine Datei ausführbar - (-v ... Ausgabe einer Diagnose für jede verarbeitete Datei).

chmod -v 0755 <MeineDatei> ... eine Datei für alle Benutzer lesbar und ausführbar machen; der Eigentümer der Datei hat zusätzlich auch das Schreibrecht

chmod 0644 <Datei_1> <Datei_2> <Datei_3> ... <Datei_n> ... die angegebenen Dateien sind für alle Benutzer lesbar; der Eigentümer der Datei hat zusätzlich auch das Schreibrecht

chmod -R 777 /srv/www/htdocs/ ... ändert die Rechte aller Dateien in diesem Verzeichnis, einschließlich aller Unterverzeichnisse (-R steht für Rekursiv)

chmod +x datei.sh ... die angegebene Datei ausführbar machen

chmod -x datei.sh ... die Rechte zum Ausführen für die angegebene Datei entziehen

chmod =x datei.sh ... die Rechte zum Ausführen als einziges Recht vergeben

x ... Recht zum Ausführen einer Datei (x ... eXecute)

r ... Leserecht (r ... read)

w ... Schreibrecht (w ... write)

chmod u+x datei.sh ... die angegebene Datei ist nur für den Eigentümer ausführbar (vorher: die Datei war für alle Benutzer nur lesbar; der Eigentümer hatte zusätzlich ein Schreibrecht; Zugriffsrechte - 0644)

chmod g-x datei.sh ... die Rechte zum Ausführen für die angegebene Datei werden der Gruppe entzogen (vorher: Zugriffsrecht – 0754)

chmod u=rwx,g=rx,o=x datei.sh ... der Eigentümer der Datei ist der Benutzer root und er ist auch Mitglied der gleichnamigen Gruppe root; die Rechte zum Ausführen für die angegebene Datei erhalten alle Benutzer; Leserecht erhalten der Eigentümer und die Mitglieder der Gruppe root; alleiniges Schreibrecht erhält nur der Eigentümer der Datei; in Zahlen ausgedrückt entspricht dies dem Zugriffsrecht – 0751

u ... Benutzer, Eigentümer (u ... user)

g ... Gruppe (g ... group)

o ... Rest der Welt (o ... other)

1 ... Recht zum Ausführen einer Datei

2 ... Schreibrecht

4 ... Leserecht

3 ... 1 + 2 → Ausführ- und Schreibrecht

5 ... 1 + 4 → Ausführ- und Leserecht

6 ... 2 + 4 → Lese- und Schreibrecht

7 ... 1 + 2 + 4 → Ausführ-, Lese- und Schreibrecht

Zahlen mit weniger als 4 Stellen erhalten automatisch vorangestellte Nullen. Daher bekommt eine Datei mit **chmod 7 datei.txt** die Rechte 0007.

Standardrechte: Verzeichnisse 0755, Dateien 0644

siehe auch: Zugriffsrechte, umask, man chmod

chown, chgrp, groups

chown -R peter /srv/www/htdocs/

chown peter /srv/www/htdocs/MeineDatei.txt

chown -R peter:users /srv/www/htdocs/

chgrp -R users /srv/www/htdocs/

chgrp users /srv/www/htdocs/MeineDatei.txt

chown ändert den Besitzer einer oder mehrerer Dateien. chgrp ändert die Gruppenzugehörigkeit einer oder mehrerer Dateien. chgrp -R users /srv/www/htdocs/ ändert die Gruppenzugehörigkeit - Gruppe users - aller Dateien in diesem Verzeichnis, einschließlich aller Unterverzeichnisse (-R steht für Rekursiv). chown -R peter /srv/www/htdocs/ ändert den Besitzer - neuer Besitzer: peter - aller Dateien in diesem Verzeichnis, einschließlich aller Unterverzeichnisse.

groups ... groups zeigt die Gruppenzugehörigkeit des aktuellen Benutzers an

chown -R --from=<OLDUSER> <NEWUSER> /srv/www/htdocs/

Es wird der Besitzer nur von Verzeichnissen und Dateien geändert, deren Besitzer aktuell <OLDUSER> ist (**Achtung:** Sie sollten wissen was Sie mit diesem Befehl tun. Im Zweifelsfall - **Hände weg** - von diesem Befehl.).

find / -uid <UID> -exec chown <NEWUSER> {} \;

Es wird der Besitzer nur von Verzeichnissen und Dateien geändert, deren Besitzer aktuell die UID <UID> hat - z.B.:

find / -uid 501 -exec chown andi {} \;

Die UID eines Benutzers erfährt man aus der Benutzerverwaltung oder aus der Datei /etc/passwd (**Achtung:** Sie sollten wissen was Sie mit diesem Befehl tun. Im Zweifelsfall - **Hände weg** - von diesem Befehl. Zwischen {} und \ muss sich ein Leerzeichen befinden.).

find <Verzeichnis> -type f -exec chmod a-x "{}" ";"

Dieser Befehl findet alle Dateien (keine Verzeichnisse - ohne Ausführrechte kann man nicht in die Verzeichnisse hineinwechseln) und entzieht ihnen die Ausführrechte (a ... gilt für alle Benutzer: user, group, other).

siehe auch: man chown, Network Filesystem einrichten

chroot

chroot (chroot steht für »change root«) führt ein Kommando oder eine interaktive Shell in einem angegebenen Wurzelverzeichnis (/) aus. Nach dem Aufruf kann das Programm in einer chroot-Umgebung nicht mehr auf Dateien außerhalb des neuen Baumes (Wurzelverzeichnis der chroot-Umgebung) zugreifen, man spricht hier auch von einer chroot-jail Umgebung oder einer Sandbox.

chroot VERZEICHNIS BEFEHL

Hinweis: chroot kann nur vom Benutzer root bzw. von Benutzern mit root-Rechten aufgerufen werden.

Beispiele:

sudo mount /dev/sda6 /mnt/debian ... ein parallel installiertes Debian-System einhängen

sudo chroot /mnt/debian /bin/bash -i ... in die Bash des parallel installierten und im eigenen System eingehängten Debian-Systems zu wechseln

Wird kein Befehl (wie hier /bin/bash) angegeben, wird `${SHELL}` -i verwendet. Die Variable `${SHELL}` ist eine Umgebungsvariable und der Parameter -i steht dabei für eine interaktive Shell.

chroot wird gerne verwendet, um kritische Programme vom übrigen Dateisystem zu isolieren. Wenn man z.B. einen Server (Fileserver etc.) in einer chroot-Umgebung laufen lässt, so befindet sich ein eventueller Eindringling im Gefängnis (jail), aus dem er kaum ausbrechen kann - natürlich installiert man in diesem Gefängnis nur Programme, die der Server selbst braucht; andere Programme haben dort nichts verloren. In der Gefängniszelle sind ja auch keine Schlosserwerkzeuge vorhanden ...

Die Einrichtung einer chroot-Umgebung ist immer dann sinnvoll, wenn einige wenige Programme in einer gekapselten Umgebung laufen sollen und die Installation eines VServer's (virtuellen Servers) zu aufwendig ist.

Anmerkung: Bei einem System-Update werden die Programme in der chroot-Umgebung nicht erneuert, d.h. ein Update der Programme in einer chroot-Umgebung erfolgt nicht automatisch. Unter Umständen muss die Installation und Einrichtung der chroot-Umgebung wiederholt werden.

A: chroot in einer Live-CD/DVD Umgebung

chroot kann auch benutzt werden, um von einer Live-CD/DVD in die Umgebung eines installierten Systems einzugreifen (z.B. vollständiger Zugriff auf das System für Reparaturen).

Im Prinzip ist jedes Linux-Live-System (Installations-CD, Knoppix etc.) zur Reparatur geeignet. Dabei ist wichtig, dass die Systemarchitektur des Live-Systems mit der des installierten Ubuntu-Systems übereinstimmt, da man sonst eine Fehlermeldung erhält.

chroot: Befehl `>>/bin/bash` konnte nicht ausgeführt werden: Fehler im Format der Programmdatei

Es ist also nicht ohne weiteres möglich, sich mit chroot und einer 32-Bit-Live-CD an einem 64-Bit-System anzumelden. Wird die gleiche CD wie zur Installation genutzt, so sollte es keine Probleme geben.

Für einen einfachen Zugriff, zum Beispiel um die Ausgabe eines einzelnen Befehls zu überprüfen, können ggf. einzelne Schritte, wie das Einbinden von /sys und /proc ausgelassen werden.

Hinweis: Alle folgenden Befehle beziehen sich auf den Einsatz von Ubuntu bzw. auf Systeme die auf Ubuntu basieren als Live-System. Für andere Systeme können bestimmte Befehle oder Pfade davon abweichen.

Ubuntu-LiveCDs erkennen das installierte Ubuntu-System selbständig und mounten es auf /media/EINE_LANGE_ZAHLENKOMBINATION, wobei EINE_LANGE_ZAHLENKOMBINATION die UUID des Dateisystems ist. Das ganze hat zur Folge, dass man die ersten drei Befehle im Kapitel Einrichtung auslassen und in allen folgenden Befehlen in diesem Artikel /mnt mit /media/EINE_LANGE_ZAHLENKOMBINATION ersetzen kann. Die unten beschriebene normale Vorgehensweise funktioniert aber natürlich weiterhin.

Für einige Linux-Systeme ist die Verwendung von dem Dateisystem Btrfs möglich, das sollte man im Terminal vorab abklären mit:

sudo blkid

Außerdem kann man damit gleich prüfen, auf welcher Partition sich Linux-System befindet.

1. Einrichtung

Nach dem Start des Live-Systems muss die Partition mit dem installierten Linux-System eingebunden werden. Wo das eigene Linux zu finden ist erfährt man mit dem Befehl

sudo fdisk -l

Es muss für ein ext-formatiertes System der Befehl

sudo mount /dev/sdxY /mnt

bzw. bei einem btrfs-formatierten System:

sudo mount -o subvol=@ /dev/sdxY /mnt

ausgeführt werden, dabei sind die Bezeichnungen xY (z.B. /dev/sda1) an die eigenen Gegebenheiten anzupassen.

Nutzt das System eine separate boot-Partition, so muss diese mit

sudo mount /dev/sdzY /mnt/boot

eingebunden werden. Auch hier sind die Bezeichnungen zY an die eigenen Gegebenheiten anzupassen. Bei einer normalen Installation ist diese Partition nicht vorhanden, und somit dieser letzte Befehl nicht erforderlich.

2. Zusätzliche Schritte

Vor dem Wechsel in das installierte System muss gegebenenfalls diesem der Zugriff auf wichtige Systeminformationen zugesichert werden. Man bindet dazu das Verzeichnis mit den Gerätedateien /dev innerhalb des installierten Systems ein:

sudo mount -t devtmpfs /dev /mnt/dev

sudo mount -t devpts /dev/pts /mnt/dev/pts

Ähnlich verfährt man mit dem Schnittstellendateisystem /proc und dem System-Verzeichnis /sys. Diese werden mit

sudo mount -t sysfs /sys /mnt/sys

sudo mount -t proc /proc /mnt/proc

sudo cp /proc/mounts /mnt/etc/mtab

eingebunden.

Der folgende Einzeiler hängt diese notwendigen Systemverzeichnisse in einem Aufruf ein:

for i in /dev /dev/pts /proc /sys /run; do sudo mount -B \$i /mnt\$i; done

Um die Internetverbindung sicherzustellen, werden unter Umständen die DNS-Server-Angaben benötigt. Diese kopiert man mit:

sudo mount -o bind /etc/resolv.conf /mnt/etc/resolv.conf

3. Auf das zu rettende System zugreifen

Um vom gebooteten Rettungssystem aus auf das installierte System zuzugreifen, mountet man die Root-Partition des installierten Systems z.B. nach dem Verzeichnis /mnt des Rettungssystems. Kapselt man es mit chroot /mnt ab, kann man darin genauso arbeiten, als befände man sich im eigentlichen System und muss nicht daran denken, allen Pfadangaben ein /mnt voranzustellen.

Es erfolgt der Wechsel in das installierte System:

sudo chroot /mnt /bin/bash

Nun kann die Reparatur des installierten System vorgenommen werden.

chroot erlaubt es auch, einzelne Befehle in einer gekapselten Umgebung auszuführen:

sudo chroot /mnt passwd

Für den Befehl passwd sieht das so aus, als ob das Verzeichnis /mnt als Root-Verzeichnis gemountet wäre.

4. Beenden

Abschließend verlässt man die chroot-Umgebung mit exit und kann, sofern man nichts anderes mehr mit dem Live-System machen möchte, das System mit **sudo reboot** neu starten.

Hinweis: Genauso kann man auch vorgehen, wenn man von einem USB-Medium booten möchte.

B: Sichere chroot-Umgebung für SSH-Datei-Übertragungen (SFTP)

SSH ist ein ausgesprochen vielseitiges und nützliches Werkzeug. Seit OpenSSH 4.8p1 existiert eine neue Option für den SSH-Server, die es ermöglicht, eine sogenannte chrooted-jail (»changed-root«-Umgebung) aufzubauen. Eine solche Umgebung bietet den Vorteil, dass Anwender, die sich mittels SFTP mit dem Server verbinden, direkt in ein vorgegebenes Unterverzeichnis »eingesperrt« werden, so dass sie von da aus nicht in höhere Verzeichnisebenen wechseln können. Damit verhindert man den Zugriff auf unerwünschte Bereiche des Dateisystems. Dies ist zwar kein Allzweckheilmittel für die Sicherheit eines Systems, erhöht jedoch die Hürde um ein gutes Stück.

Anwenden kann man diese Konfiguration z.B. bei Shared-Webhosting, dort wo unabhängige Benutzer ihren eigenen Webspace verwalten. Anstelle des

unsicheren FTP kann hier dann eine sichere SFTP-Kommunikation stattfinden.

Falls der OpenSSH-Server nicht installiert ist, so kann er mit der Paketverwaltung oder auf der Kommandozeile mit

sudo apt-get install openssh-server

installiert werden.

1. Benutzer-Konfiguration

Zunächst ist auf dem System eine Gruppe namens sftponly für die Benutzer, die Dateien über SFTP an den Server senden, anzulegen.

sudo groupadd sftponly ... eine Gruppe namens sftponly anlegen

sudo useradd -s /bin/false webuser_01 ... Benutzer ohne Login-Shell anlegen

sudo passwd webuser_01 ... Zugangspasswort für den Benutzer webuser_01 festlegen

sudo usermod -a -G sftponly webuser_01 ... Benutzer webuser_01 zur Gruppe sftponly hinzufügen

Mit dem usermod-Befehl können beliebigen Benutzer zur Gruppe sftponly hinzugefügt werden.

2. OpenSSH-Konfiguration

Als nächstes ist die SSH-Konfiguration anzupassen. Der standardmäßige Eintrag für den sftp-Dienst ist zu deaktivieren und durch den Eintrag internal-sftp zu ersetzen. Die einzufügende Match-Direktive sorgt dafür, dass nur Benutzer die der Gruppe sftponly angehören den Dienst sftp nutzen dürfen.

Datei: /etc/ssh/sshd_config

[...]

```
Subsystem      sftp    internal-sftp
#Subsystem     sftp    /usr/lib64/misc/sftp-server
```

```
Match Group sftponly
    ChrootDirectory /home/%u
    ForceCommand internal-sftp
    X11Forwarding no
    AllowTcpForwarding no
```

Match

3. Konfiguration des Home-Verzeichnisses

Eine Einschränkung bringt jedoch die Chroot-Konfiguration mit sich (**Hinweis:** Eigentümer des chroot-Ordners muss »root« (UID 0) sein.), die Benutzer der Gruppe sftponly besitzen in der obersten Verzeichnisebene ihres Home-Verzeichnis kein Schreibrecht.

Dieses Problem kann durch Erstellung eines Unterverzeichnisses (z.B. ~/public_html) im Home-Verzeichnis des Benutzers behoben werden. In diesem Unterverzeichnis hat der Benutzer uneingeschränktes Schreibrecht und kann dort seine Dateien speichern und auch wieder löschen.

```
sudo mkdir /home/webuser_01
sudo chmod 750 /home/webuser_01
sudo chown root:sftponly /home/webuser_01
sudo mkdir /home/webuser_01/public_html
sudo chmod 700 /home/webuser_01/public_html
sudo chown webuser_01:sftponly /home/webuser_01/public_html
```

4. Anpassungen in der Datei /etc/ssh/sshd_config

In den Datei /etc/ssh/sshd_config sind zur Erhöhung der Sicherheit noch einige Änderungen notwendig.

Datei: /etc/ssh/sshd_config

Port 42123

Der Standardport für SSH ist 22. Mit dieser Variablen lässt sich der Listen Port z.B. auf 42123 für den Server ändern. Das macht den Server zwar kein bisschen sicherer (security through obscurity), hilft aber unter Umständen die Existenz des ssh-Servers zu verschleiern. Die Ports größer 1024 bis 65536 können im Allgemeinen frei gewählt werden (**siehe auch:** Ports, man /etc/services).

PermitRootLogin no

Wenn ein Angreifer root werden möchte, so muss er mit dieser Einstellung mindestens noch das Passwort eines weiteren Benutzers auf dem Server kennen.

Hinweis: Unter Ubuntu und bei Linux-Distributionen die auf Ubuntu basieren ist der Benutzerzugang root standardmäßig deaktiviert, somit spielt es keine Rolle ob hier der Wert **yes** oder **no** eingetragen ist.

PermitEmptyPasswords no

Sollte dort in der Standardkonfiguration eine **yes** stehen, so ist dieser Eintrag in **no** zu ändern. Damit werden SSH-Nutzer ohne Passwort abgewiesen.

5. Testen des OpenSSH-Server

Jetzt ist es an der Zeit den OpenSSH-Server neuzustarten. Als SFTP-Klienten eignen sich später z.B. das Firefox Addon Fireftp, Filezilla, WinSCP für Windows oder ganz einfach das Kommandozeilenprogramm sftp bzw. die vielseitig einsetzbaren Dateimanager unter Linux.

/etc/init.d/ssh restart

Danach sollte der SFTP-Zugriff für den Benutzer webuser_01 am Beispielerchner 192.168.0.200 funktionieren. Hier im Beispiel wurde der Port 42123 in der Datei /etc/ssh/sshd_config eingetragen (Standardport ist der Port 22).

sftp -P 42123 webuser_01@192.168.0.200

und der ssh-Zugriff verwehrt werden

ssh -p 42123 webuser_01@192.168.0.200

Hinweis: Sollte etwas nicht funktionieren, so liegt dies möglicherweise an einer aktivierten Firewall. Der von SSH-Server benutzte Port (**siehe auch:** /etc/ssh/sshd_config) sollte in der Firewall geöffnet werden.

6. Last but not least – Denyhosts

Wenn man seinen Server eine Weile laufen lässt und spaßeshalber mal die Datei /var/log/auth.log öffnet, werden einem die regen Zugriffsversuche auffallen die allesamt von unbekannten Bneutzern stammen. Willkommen im Internet! John Doh und Nihao versuchen mit Brute-Force-Attacken in den ssh-Server einzubrechen. Eine wirksame Methode das zu unterbinden ist z.B. das Installations-Paket: denyhosts.

Die Datei /etc/denyhosts.conf ist gut kommentiert und lässt sich leicht an die eigenen Wünsche anpassen. Das Programm registriert fehlgeschlagene Logins und setzt die IP Adresse des Angreifers nach einer festgelegten Anzahl von Versuchen in die Datei /etc/hosts.deny, womit jeder weitere Loginversuch verwehrt wird.

Eine sehr restriktive Methode ist die direkte Verwaltung der Zugriffe auf den SSH-Server über die Dateien /etc/hosts.allow und /etc/hosts.deny. Mit

Hilfe dieser Dateien und den festen IP-Adresse (keine dynamischen IP-Adressen) von bekannten Benutzern, kann man alle Zugriffe mit Hilfe der Dateien /etc/hosts.allow und /etc/hosts.deny verbieten und nur ausgewählte IP-Adressen den Zugriff erlauben.

siehe auch: man hosts_options, man hosts_access

7. DynDNS.org

Es ist zwar schön, dass der Server im eigenen internen Netzwerk funktioniert, aber irgendwie müssen die Benutzer auch von außerhalb zugreifen können. Die meisten werden sicherlich eine dynamische IP vom Internet-Provider zugewiesen bekommen, weshalb es Sinn macht einen kostenlosen DNS-Dienst wie z.B. dyndns.org (kostenpflichtig) oder www.noip.com zu benutzen, um eine leicht zu merkende URL wie z.B. 4freunde.dyndns.org zu erhalten. Nachdem man sich bei dyndns.org angemeldet hat, muss man dem Dienst auf irgendeine Art noch mitteilen wie die aktuelle IP-Adresse des eigenen SSH-Servers lautet. Dazu bietet sich z.B. Installations-Paket **ddclient** an. Bei der Installation wird man nach Login und Passwort bei dyndns.org gefragt und kann ansonsten die Standardeinstellungen verwenden.

Sollte man eine Firewall einsetzen oder einen Router mit eingebauter Firewall nutzen, so muss nur noch der SSH-Dienst mit Hilfe von PortForwarding an den Zielservers und Port 42123 weitergeleitet werden. Danach können sich die Benutzer von überall auf der Welt mit

sftp -P 42123 webuser_01@4freunde.dyndns.org

auf dem SSH-Server einloggen.

8. Zusätzliche Hinweise

Die Verzeichnisse der Benutzer für die Speicherung der Dateien können auch im Verzeichnis /var/sftp oder /srv/sftp erstellt werden.

Anmerkung: Mit schroot ("securely enter a chroot environment") kann man Befehle oder eine Login-Shell als normaler Benutzer in einer chroot-Umgebung ausführen.

siehe auch: man chroot, man ssh, groupadd, useradd, usermod, SSH - secure shell, Uncomplicated Firewall (UFW), man sshd_config, man ssh_config, man /etc/services, hosts.allow, hosts.deny

clear

clear bereinigt den Bildschirm von überflüssig gewordenen Textzeilen. Mit

der Tastenkombination **[Strg] + [L]** wird der Bildschirm ebenfalls bereinigt.

cp

Datei kopieren.

`cp <QuellDatei> <ZielDatei>`

cp -a ... kopiert ein ganzes Verzeichnis, einschließlich seiner Unterverzeichnisse und die Dateieigenschaften bleiben bei den Kopien erhalten; besonders geeignet für Backups

cp -r ... kopiert ein ganzes Verzeichnis, samt seiner Unterverzeichnisse

cp -p ... kopiert die Zugriffsrechte und auch evtl. vorhandenen erweiterten Dateirechte von der Quelldatei zur Zieldatei

siehe auch: man cp

crontab -u <users> file

crontab -u <users> -e

crontab -u <users> -l

crontab -u <users> -r

Mit Cronjobs können Programme zeitgesteuert aufgerufen und abgearbeitet werden.

1. Shellskript erstellen

Hinweis: Shellskript und die Ergebnisdateien des Shellskripts sind in ein Verzeichnis zu speichern, indem der Benutzer über Schreibrechte verfügt (hier im Beispiel heißt der Benutzer karl).

Beispiel: Dateiname date

#!/bin/bash

#Dieses Shellskript schreibt regelmäßig das aktuelle Datum

#in die Datei /home/karl/zeit.txt

date >> ./zeit.txt

Mit **chmod 755 /home/karl/date** wird das Shellskript date zu einer ausführbaren Programmdatei.

2. Anschließend, sich als root anmelden.

3. `cd /var/spool/cron/tabs/`

4. `touch crontab` (die Datei kann einen beliebigen Namen erhalten - crontab hat sich eingebürgert)

5. `crontab -u karl -e`

6. im Editor (-e) die Taste [Einf]g oder [i] drücken und damit in den Einfügemodus wechseln und folgende Zeile eingeben:

1-59/5 * * * * /home/karl/date

Taste [Esc] drücken und anschließend folgendes Kommando eingeben

:wq!

Damit wird die Zeile in die Datei crontab geschrieben (w ... write) und der Editor beendet (q ... quit).

Alternativ: Midnight Commander öffnen mit **mc**, die Datei crontab öffnen bzw. **mcedit ./crontab** eingeben und z.B. die folgenden Zeilen schreiben

```
#Minute (1...59) Stunde (0...23) Tag (1...31) Monat (1..12)
#Tag der Woche (0...7) command
#* Jokerzeichen steht für alle Zeiten, 1-59/5 oder */10 bei
#Minute steht für jede 5 Minuten bzw. 10 Minuten
#(1=Montag, 0 oder 7=Sonntag, 1-5=Werktag)
#command z.B. /home/karl/date ruft das Shellskript date auf
1-59/5 * * * * /home/karl/date
*/10 * * * * /home/karl/date2
```

7. crontab -u karl ./crontab

8. crontab -u karl -l

Es sollte der Cronjob mit der vorher geschriebenen Zeile angezeigt werden.

Anmerkung: Nach Änderungen in der Datei crontab ist der Cronjob bei Cron erneut anzumelden (u ... user).

crontab -u karl ./crontab

bzw.

crontab -u karl /var/spool/cron/tabs/crontab

Kontrolle des Cronjobs (l ... list):

crontab -u karl -l

In der Datei crontab können beliebig viele Conjobs eingetragen werden - je Zeile ein Cronjob.

Der Cronjob kann komplett mit folgenden Befehl gelöscht werden (r ...

replace):

crontab -u karl -r

Hinweis: Wenn bei command statt eines Shellskripts ein Shellkommando eingegeben wird, so ist der Befehl so einzugeben wie auf der Kommandozeile. Wollen Sie mehrere Kommandos durch die crontab-Zeile ausführen, trennen Sie diese durch ein Semikolon [;].

Beispiele:

10 * * * * date

Das Kommando wird immer 10 Minuten nach einer vollen Stunde ausgeführt.

10,20,30 * * * * date

Das Kommando wird immer in der 10,20 und 30 Minute nach einer vollen Stunde ausgeführt.

0 0 * * 5 date

Das Kommando wird immer jeden Freitag um 0:00 Uhr ausgeführt.

0 2-4 * * * date

Das Kommando wird jede Nacht zwischen 2:00 und 4:00 Uhr immer zur vollen Stunde (also dreimal pro Nacht) ausgeführt.

0 10-18/2 * * * date

Das Kommando wird jeden Tag zwischen 10:00 und 18:00 Uhr alle 2 Stunden ausgeführt.

*/12 5 * * * date

Das Kommando wird in der fünften Stunde jede 12 Minuten ausgeführt.

Wenn im Verzeichnis /var/spool/cron die Datei allow existiert, kann crontab nur von den darin aufgeführten Systembenutzern ausgeführt werden. Wenn anstelle der allow-Datei eine Datei mit dem Namen deny existiert, steht das crontab-Kommando allen Systembenutzern zur Verfügung, die dort **NICHT** aufgeführt sind.

Wenn keine dieser Dateien existieren, hängt das Verhalten von crontab von Einstellungen bei der Compilierung ab. Entweder kann jeder das crontab-Kommando ausführen, oder nur der Superuser (root).

Wenn die crontab-Datei vom crontab-Kommando verändert wird, liest der Dämon crond automatisch die neuen Daten, d.h. crond muss **nicht** extra neu gestartet werden (wie z.B. der Apache-Server nach Änderungen an der Konfigurationsdatei httpd).

siehe auch: Anhang: Links -> fcron

cpuinfo

siehe auch: proc Dateisystem

crypt

siehe auch: useradd, Anhang: Skript-Listings -> Crypt

curlftpfs

CurlFtpFS ist ein auf FUSE und cURL basierendes Dateisystem für den Zugriff auf FTP-Server.

CurlFtpFS ist ein Programm zum Mounten von FTP-Servern als lokale Verzeichnisse. Es verbindet sich mit einem FTP-Server und bildet dessen Verzeichnisstruktur auf das lokale Dateisystem ab.

Auf dem lokal gemounteten FTP-Verzeichnis erscheinen die Dateien und Unterverzeichnisse so, als lägen sie auf der lokalen Festplatte. Zwar ist die Zugriffszeit deutlich länger, Befehle und Programme können aber trotzdem auf die Dateien zugreifen – auch zum Suchen und Ersetzen in Dateien.

`curlftpfs [options] <ftphost> <mountpoint>`

Beispiel:

FTP-Server: ftp.server.de

Benutzername: jemand

Passwort: geheim

lokales Verzeichnis: /home/<Benutzername>/ftp/

**`curlftpfs ftp://jemand:geheim@ftp.server.de
/home/<Benutzername>/ftp/`**

Hat alles geklappt, so kann man die FTP-Verbindung wie ein Verzeichnis ansprechen.

Im Fehlerfall kann man noch die Option -v einfügen, um eine ausführliche detaillierte Ausgabe aller FTP-Befehle zu erhalten.

Um die Verbindung zu trennen, schließen sie alle geöffneten Dateien im FTP-Verzeichnis und geben dann den folgenden Befehl ein.

`fusermount -u /home/<Benutzername>/ftp/`

siehe auch: man curlftpfs

cut

Ausgewählte Teile einer oder mehrerer DATEIEN auf Standardausgabe

ausgeben.

cut [OPTION] ... [DATEI] ...

Erforderliche Argumente für lange Optionen sind auch für kurze erforderlich.

-c, --characters=LISTE ... nur diese Zeichen ausgeben

-d, --delimiter=TRENN ... TRENN anstelle von Tabulator als Trenner benutzen; Wenn cut das Trennzeichen nicht finden kann, so hebt es schlichtweg die Hände und lässt die gesamte Ausgabe durch (siehe auch: **-s, --only-delimited**).

-f, --fields=LIST ... nur die angegebenen Felder auf Standardausgabe ausgeben

-s, --only-delimited ... keine Zeilen ausgeben, die keinen Trenner enthalten

--output-delimiter=ZKETTE ... ZKETTE als Ausgabetretnnzeichen benutzen; Voreinstellung ist das Eingabetrennzeichen

Ohne DATEI, oder wenn DATEI "-" ist, die Standardeingabe lesen.

Beispiele:

cut -d : -f 1,3 /etc/passwd ... Benutzernamen und die UID am Bildschirm ausgeben

In den nachfolgenden Beispielen wird die echo-Ausgabe an cut zur weiteren Bearbeitung umgeleitet. Diese Methode wird häufig in Shellskripten angewendet.

echo "12:hallo:buena:karl" | cut -d : -f 1,3 ... 12:buena

echo "12 hallo buena:karl" | cut -d " " -f 1,3 "-" ... 12 buena:karl

echo "12 hallo buena:karl" | cut -c 4 ... h

echo "12 hallo buena:karl" | cut -c 4-7 ... hall

echo "12 hallo buena:karl" | cut -c 7,11,13,14 ... luna

echo -e "12\thallo\tbuena\tkarl " | cut --output-delimiter=" " -f 2,4 ...
hallo karl (echo -e "\t" ... gibt am Bildschirm einen Tabulator aus)

echo -e "12\thallo\tbuena\tkarl " | cut --output-delimiter="+" -f 2,4 ...
hallo+karl (echo -e "\t" ... gibt am Bildschirm einen Tabulator aus)

df -T /dev/hda2 | grep "^[0-9]\{1,2\}%\+.*\$" | cut -d " " -f 13 ... 20%;
Auslastung der Partition /dev/hda2 ermitteln

siehe auch: Umleitung von Befehle

D

date

In Unix und Linux werden Datumsangaben intern immer als die Anzahl der Sekunden die seit dem 1. Januar 1970 um 00:00 Greenwich Mean Time (GTM, heute UTC) vergangen sind.

In manchen Situationen muss man - z.B. in Shellskripten - die Unix-Zeit in ein normales Datum umrechnen und umgekehrt.

date ... die aktuelles Datum und die Uhrzeit ausgeben

Ausgabe: Mi 12. Mär 13:45:45 CET 2014

date +%s ... aktuelle UNIX-Timestamp

Ausgabe: 1394626751

date -d @1394626751 ... UNIX-Timestamp in ein Datum umrechnen

Ausgabe: Mi 12. Mär 13:19:11 CET 2014

date -d '2008-12-18 12:34:00' +%s ... eine festgelegtes Datum und Uhrzeit als UNIX-Timestamp ausgeben

Ausgabe: 1229600040

date -d '12-oct-2013' +%s ... eine festgelegtes Datum und Uhrzeit als UNIX-Timestamp ausgeben

Ausgabe: 1381528800

date -d @1234567890 +%d.%m.%Y ... UNIX-Timestamp in ein formatiertes Datum umrechnen

Ausgabe: 14.02.2009

date -d '1970-01-01 00:00:00' +%s ... Zeitstempel wird immer in UTC gerechnet - im Gegensatz zum ausgeschriebenen Datum (Datumsausgabe), welches relativ zur Zeitzone des Systems zu verstehen ist

Ausgabe: -3600

Warum? Als es in Deutschland der 1.1.1970 um 00:00 war, war es in England noch 23:00 am Vortag.

Systemzeit ändern

date --set='15:25:00' ... setzt die Systemzeit auf den angegebenen Wert bzw.

date --set "13:30:00" +%H:%M:%S

date --set "04/16/2005" +%D ... setzt das Systemdatum auf den 16. April 2005 – Monat/Tag/Jahr

Hinweis: Die Systemzeit bzw. das Systemdatum kann nur von root oder mit sudo-Rechten geändert werden.

Zeitdrift: Linux-Systemzeit und Rechneruhr (CMOS-Uhr) synchronisieren

Die Rechneruhr geht in aller Regel nicht wirklich genau, doch sind für die meisten Anwender die Unterschiede nicht besonders wichtig. Da die Gangdifferenz aber meist nicht zufällig ist, sondern eine systematische Komponente aufweist, kann die Rekalibrierung automatisiert werden.

Zu diesem Zweck gibt es die Datei »/etc/adjtime«. Wird die Rechneruhr zweimal manuell umgestellt (über »hwclock --set ...«), so berechnet das Programm aus der vergangenen Zeit und der Größe der Korrektur die systematische Drift. Sind beispielsweise zwischen den beiden Zeitpunkten 30 Tage vergangen und bei der zweiten Korrektur ist die Zeit um eine Minute zurückgestellt worden, so ergibt sich eine Drift von zwei Sekunden pro Tag. Über »hwclock --adjust« (beim Systemstart aufgerufen oder mittels »cron« gestartet) wird dann automatisch die Korrektur eingearbeitet.

1. als root anmelden

2. Systemzeit setzen - 14.11.2005 20:30:00 Uhr

date --set '2005-11-14 20:30:00' +%Y-%M-%d %T'

3. die CMOS-Uhr mit der Systemzeit synchronisieren - 14.11.2005 20:30:00 Uhr

hwclock --set --utc --date "2005-11-14 20:30:00"

Die Option »--localtime« statt »--utc« übernimmt die lokale Zeit als Systemzeit.

4. Pkt. 1 bis 3 ist nach ca. 30 Tagen zu wiederholen.

5. Den Befehl hwclock --adjust mit cron täglich starten lassen. Dafür ist die Crontab wie folgt anzupassen:

0 12 * * * /sbin/hwclock --adjust

Im Beispiel wird der Befehl täglich um 12:00 Uhr als Benutzer root ausgeführt.

Hinweis: Der Befehl /sbin/hwclock --adjust kann auch in der Datei /etc/init.d/boot.local eingetragen werden. Der Befehl wird dann bei jeden Bootvorgang ausgeführt.

Dieses Verfahren führt zu einer insgesamt akzeptablen Genauigkeit.

Wirklichen Genauigkeitsfanatikern kann das aber nicht genug sein. Als Lösungsmöglichkeiten bieten sich hier der Anschluss einer Funkuhr (DCF77) oder eines GPS-Receivers an einen seriellen Port des Rechners an oder die Aktualisierung der Uhrzeit aus dem Netz.

Ein Problem, das allerdings erst am 19. Januar 2038 aktuell wird, sei hier aber noch erwähnt: Da die Zeit intern in einem 32-Bit-Wert gespeichert wird, läuft die Systemzeit an diesem Tag (um 03:14:08 Uhr) in das Jahr 1970 zurück. Wie bei der Jahr-2000-Umstellung wird hier also viel Arbeit auf die Programmierer zukommen. Nur: Wer wird im Jahr 2038 noch in C programmieren können?

Hinweis: Nach einem CMOS-Batteriewechsel (auf dem Motherboard) sollten alle Einträge in der Datei `/etc/adjtime` gelöscht werden (z.B. mit **echo -n > /etc/adjtime**). Anschließend ist die Zeitdrift, wie unter Pkt. 1 bis 4 beschrieben, erneut zu ermitteln.

siehe auch: `crontab`

Dateien: versehentlich gelöschte Dateien wiederherstellen

Falls Sie auf ein ext2 Dateisystem (ext3, ext4) versehentlich eine oder mehrere Dateien gelöscht haben und Sie wollen den Inhalt dieser Dateien wiederherstellen.

Hinweis: Die Wiederherstellung dieser Dateien aus einem Backup ist natürlich wesentlich einfacher, soweit ein aktuelles Backup zur Verfügung steht.

Hintergrund

Wie auch unter anderen Betriebssystemen üblich wird auch auf dem Linux ext2 Dateisystem eine Datei dadurch »gelöscht«, dass entsprechende Verweise in der Verzeichnisdatei gelöscht werden. Die eigentlichen Daten der Datei sind nach dem eigentlichen »löschen« zwar zum Überschreiben freigegeben, aber noch auf der Festplatte lesbar. Erst beim Anlegen weiterer Dateien werden diese nun freigegebenen Datenblöcke überschrieben.

Zudem existiert auch nach dem Löschen die sog. Inode in der die logische Verkettung der Blöcke und die Zugriffsrechte gespeichert sind. Zudem wird hier auch der Zeitpunkt des Löschens (»Deletion time«) festgehalten. Was wirklich durch das Löschen endgültig verloren gegangen ist, ist der Dateiname.

Durch Setzen eines Flags mit Hilfe des Kommandos **chattr** kann man auf dem ext2 Dateisystem auch ein »sicheres« Löschen beim Absetzen eines Löschkommandos bewirken. Hierdurch wird die Datei vom Kernel beim Löschen vollständig mit Nullen überschrieben. Solchermaßen gelöschte Daten sind nur u.U. mit speziellen Geräten, keinesfalls aber mit der hier beschriebenen Methode wiederherstellbar.

Eine Warnung vorneweg. Mit dem im folgenden beschriebenen Programm **debugfs** können Sie sehr systemnahe Zugriffe auf das Dateisystem durchführen. Ohne genaue Kenntnis der Interna des ext2 Dateisystems ist es sehr gefährlich für Ihre restlichen Daten mit den in der Hilfe bzw. in der Man-Page zu »debugfs« beschriebenen Kommandos zu »experimentieren«. Falls Sie experimentieren möchten, so legen Sie sich am besten eine Übungspartition an.

Vorgehen

Führen Sie keine Schreibzugriffe mehr auf das Dateisystem aus, auf dem Sie Dateien wiederherstellen möchten. Insbesondere das Neuanlegen von Dateien kann fatal für Ihre zu rettenden Daten sein. Falls Sie nur eine Partition »/« verwenden, so fahren Sie den Rechner am besten geordnet herunter und booten mit einem Rettungssystem. Ansonsten reicht es, wenn Sie auch die betroffene(n) Partition(en) mit dem Kommando **umount** aushängen.

Nehmen wir einfach an Sie haben folgende Datei versehentlich gelöscht und wollen Sie wiederherstellen.

```
erde:/mnt/Versuch # ls -la Protokoll_2009.txt
-rw-r--r-- 1 cg suse 1050 Dec 29 13:31 Protokoll_2009.txt
erde:/mnt/Versuch # rm Protokoll_2009.txt
```

Die betreffende Partition ist sofort aus dem Linux-Verzeichnisbaum auszuhängen (ummounten).

```
erde:~# umount /mnt
```

In meinem Beispiel handelt es sich um die Partition /dev/sdc1 die auf den Mountpoint /mnt gemounted ist. Sie müssen natürlich den Devicenamen in Ihrem System in den unten aufgeführten Beispielen einsetzen. Geben Sie bitte das Kommando **debugfs devicename** ein um den Filesystemdebugger aufzurufen. Dieser meldet sich nach dem Start mit dem Prompt debugfs: an dem Sie Kommandos zum direkten Zugriff auf das Dateisystem eingeben können.

```

erde:/# debugfs /dev/sdc1
debugfs 1.17, 26-Oct-1999 for EXT2 FS 0.5b, 95/08/09
debugfs: lsdel
1 deleted inodes found.
Inode Owner Mode Size Blocks Time deleted
25794 515 100644 1050 2/ 2 Wed Dec 29 13:32:32 1999

```

Mit dem Kommando **lsdel** kann man sich also die gelöschten Inodes auflisten lassen. An den Dateirechten (Mode), dem Eigentümer (hier als numerische User-ID), dem Zeitpunkt des Löschens und der ehemaligen Dateigröße (Size) kann man recht gut die wiederherzustellende Datei auffinden, falls mehrere Dateien gelöscht wurden (und beispielsweise nur eine wiederhergestellt werden soll). Wichtig ist hier für das folgende **dump** Kommando vor allem die Inode Nummer, im obigen Beispiel 25794.

Geben Sie folgendes Kommando ein um die oben gelöschte Datei mit den gleichen Dateirechten als neue Datei »gerettet« im Verzeichnis /tmp wiederbeleben. Beachten Sie bitte, dass Sie die spitzen Klammern < und > mit eingeben!

```

debugfs: dump -p <25794> /tmp/gerettet
debugfs: quit
erde:/# ls -l /tmp/gerettet
-rw-r--r-- 1 cg suse 1050 Dec 29 13:31 /tmp/gerettet

```

Nach dieser Aktion können Sie das Dateisystem wieder mounten und die Datei aus dem /tmp Verzeichnis zurückspielen.

Es gibt noch weitere Methoden diese Datei wiederherzustellen, doch diese Methode hat den Vorteil keine Schreibzugriffe über debugfs oder nach vollzogener Wiederherstellung einen Dateisystemcheck zu benötigen.

Hinweis: Neben dem Programm **debugfs** gibt es noch das Programm **recover**. Das Programm **recover** sollte bereits installiert sein, **bevor** der Schadensfall eintritt.

* * * * *

Dateien: versehentlich gelöschte Dateien mittels des Befehls grep wiederherstellen

Zuerst sollte die betroffene Partition sofort als READ-ONLY gemountet werden oder komplett ummounten, damit der Kernel nicht irgendwas

überschreibt.

mount -r -t ext2 /dev/hdaX /mnt/ -o remount=ro

Danach sollte man verhindern, dass durch verringerte Systemaktivitäten Daten der gelöschten Datei nicht überschrieben werden. Dazu fährt man das System am besten in Runlevel 1 herunter:

init 1

Wer das Risiko liebt, kann auch darauf verzichten.
Danach kann der eigentliche grep-Befehl aufgerufen werden.

Ausgabe der gefundenen Zeilen am Bildschirm, mit Angabe des Dateinamen:

grep -a -B zeilendavor -A zeilendanach "einwenigtextderdatei"
/dev/hdx

Ausgabe der gefundenen Zeilen am Bildschirm, ohne Angabe des Dateinamen:

grep -ah -B zeilendavor -A zeilendanach "einwenigtextderdatei"
/dev/hdx

Ausgabe der gefundenen Zeilen in der Datei delete.txt (Home-Verzeichnis), ohne Angabe des Dateinamen:

grep -ah -B zeilendavor -A zeilendanach "einwenigtextderdatei"
/dev/hdx > ~/delete.txt

zeilendavor meint die Anzahl der Zeilen die vor dem Suchwort in der Datei vorkommen (ungefährer Wert reicht natürlich, da man ja die Zeilen die nicht stimmen Wegschneiden kann)

zeilendanach sind die Anzahl der Zeilen die nach dem Suchtext **"einwenigtextderdatei"** in der gelöschten Datei stehen.

"einwenigtextderdatei" durch ein paar Worte der gelöschten Datei ersetzen.

/dev/hdx soll die Partition sein, auf der die Datei gelöscht wurde (z.B. /dev/hda2)

In der Ausgabe wird sich dann die gelöschte Datei befinden, die einfach in eine neue Datei kopiert werden kann.

Der ganze Suchprozess kann bei großen Partitionen einige Zeit in Anspruch nehmen - also viel Geduld mitbringen.

siehe auch: grep, proc-Dateisystem → /proc/filesystems

* * * * *

Datenspuren auf dem Rechner

Bei der Benutzung des Rechners entstehen zwangsläufig Datenspuren, von denen man mitunter nicht will, dass sie vom Netzwerkadministrator eingesehen werden.

Deshalb sollte man sich folgende Dateien oder Verzeichnisse etwas genauer ansehen und gegebenenfalls einige Datenspuren davon löschen.

Achtung: Beim Löschen von Daten, sollte man wissen was man tut.

.bash_history ... diese versteckte Datei liegt im Home-Verzeichnis und speichert die letzten 1000 Kommandos die man auf der Kommandozeile eingegeben hat

./thumbnails ... in ihr werden die Thumbnails von Text- und Bilddateien gespeichert - siehe Dateivorschau im Dateimanager

/tmp ... im tmp-Verzeichnis werden einige Daten zwischengespeichert, sie geben Rückschluss welche Programme in der letzten Zeit aufgerufen wurden - z.B. gpg, ssh

/usr/tmp ... im tmp-Verzeichnis werden einige Daten zwischengespeichert

/var/spool/cups ... dort erfährt man, welche Dokumente zuletzt gedruckt wurden

/var/spool/mail ... dort werden die lokalen E-Mails abgelegt, i.d.R. kann aber nur der Eigentümer die Mails lesen; E-Mails im mbox-Format; die Mails werden meist durch das Programm mail erstellt und angezeigt
/var/tmp ... dort speichern einige Programme benutzertypische Dateien
/home/<Benutzer> ... mitunter werden vom Programm mail gelöschte E-Mails, ins Home-Verzeichnis verschoben; Namen der gelöschten Mails -> mbox

/var/log ... dort werden die Log-Dateien des Betriebssystems gespeichert; ältere Log-Dateien werden dort im GZIP-Format abgelegt

diff

Unterschiede in Textdateien anzeigen. diff zeigt als Ergebnis immer das an, was in der ersten Datei geändert werden muss, damit sie der zweiten Textdatei entspricht.

diff [OPTIONEN] <Datei1> <Datei2>

diff -q <Datei1> <Datei2> ... Gibt es keine Unterschiede so schweigt diff

und sie sehen nach kurzer Zeit den Eingabe-Prompt wieder.

diff <Datei1> <Datei2>

1c1

< Das ist eine einfache Textdatei

> Das ist einfacher Text

Das kryptische Kürzel 1c1 bedeutet, dass sich beide Dateien in der 1. Zeile unterscheiden (c ... change, 19a19,25 bedeutet z.B. - das in der ersten Datei nach Zeile 19, die Zeilen 19 bis 25 angehängt werden müssen - a steht für append anhängen). Danach folgen die Ausgaben der ersten Datei, eine Trennlinie und die entsprechende Zeile der zweiten Datei.

Bei längeren Texten können die Unterschiede an den Betrachter less weitergegeben werden.

diff -u <Datei1> <Datei2> | less

Umleitung in eine Textdatei, um das Ergebnis z.B. mit dem Texteditor Kate zu betrachten.

diff <Datei1> <Datei2> > ergebnis.diff

Anschließend kann die ergebnis.diff mit Kate oder KWrite geöffnet werden, die einzelnen Zeilen werden farbig markiert. Wenn die Ansicht von Kate in zwei Fenster geteilt wird, kann z.B. im linken Fenster die erste Datei und im rechten Fenster die ergebnis.diff angezeigt werden. Die Datei kann dann entsprechend der Datei ergebnis.diff bearbeitet werden.

diff3 <Datei1> <Datei2> <Datei3>

diff3 kann 3 Dateien gleichzeitig vergleichen.

===2

1:1c

3:1c

Das ist eine einfache Textdatei

2:1c

Das ist einfacher Text

Hinter den drei Trennzeichen (===) steht die Nummer der Datei, die an dieser Stelle von den anderen abweicht (fehlt die Nummer so sind alle Dateien unterschiedlich). Es folgen die Zeilennummer und die Zeile selbst.

ding

ding ist ein Programm zur Nutzung von Wörterbüchern. Um das Programm

zu nutzen ist eine Wortliste mit Übersetzungen zu installieren, wobei jeweils ein Wort oder Ausdruck in zwei Sprachen in einer Zeile stehen muss. Die beiden Begriffe müssen durch irgendein Trennzeichen (::) voneinander abgegrenzt sein. Standardmäßig benutzt Ding das Deutsch- Englisch-Wörterbuch aus dem Paket **trans-de-en**, aber Sie können jede andere Übersetzungs-Wortliste mit einem Eintrag pro Zeile benutzen (**siehe auch:** /usr/share/trans/de-en). Für die Suche in dieser Wortliste benutzt ding das Programm agrep.

ding [Optionen] [Suchwort]

Beispiel:

ding Wildwasser ... Übersetzung des Wortes »Wildwasser« im Ding-Wörterbuch (hier: de-en) suchen

```
Datei: /usr/share/trans/de-en
# Beispieleinträge aus dem Wörterbuch de-en
[...]
Winterkleid {n} :: winter clothes
Winterlandschaft {f} :: winter landscape
Wintermantel {m} | Wintermäntel {pl} :: winter coat | winter coats
Wintermonat {m} | Wintermonate {pl} :: winter month | winter months
[...]
```

In der Wortliste ist pro Zeile nur ein Eintrag zulässig. Der Sprachentrenner ist hier ein doppelter Doppelpunkt (::) und als Trenner für Begriffsvarianten (z.B. Einzahl, Mehrzahl) wird der gerade Strich (|) verwendet. Optionale Zusatzinformationen wurden hier geschweifte Klammern ({}) eingeschlossen.

Alternative Programme mit grafischer Oberflächen sind stardict und opendict.

Hinweis: Unter Linux Mint 17 ist zusätzlich das Programmpaket **agrep** zu installieren. Weiterhin ist in der Datei /usr/bin/ding ist eine Ergänzung einzuarbeiten.

```
sudo gedit /usr/bin/ding
```

In dieser Datei /usr/bin/ding ist die 3. Zeile wie folgt zu ändern:

vorher: exec wish "\$0" "\$@"

nachher: `exec wish8.4 "$@" "$@"`

In den folgenden Versionen ist dieser Fehler möglicherweise beseitigt. Falls nicht, so ist die zu ergänzende Zahlenkombination evtl. an die neue Version anzupassen (**siehe auch:** INTERNET).

siehe auch: `ding --help`

dd

dd dient zum bit-genauen Kopieren von Festplatten, Partitionen oder Dateien. "Bit-genaues" Kopieren bedeutet, dass der Datenträger Bit-für-Bit, Byte-für-Byte bzw. Sektor-für-Sektor ausgelesen und beschrieben wird, unabhängig von dessen Inhalt und Belegung. **dd** funktioniert grundsätzlich mit allen Dateisystemen auf die Ubuntu / Linux zugreifen kann (z.B. ext2/3/4, reiserfs, vfat, ntfs etc.). Es funktioniert auch mit CD/DVD-Dateisystemen (iso9660, udf, etc.).

Hinweis: **dd** wird ohne weitere Rückfragen bzw. Sicherheitsabfragen ausgeführt. Bei unachtsamen Aufrufen könnten evtl. vorhandene Daten überschrieben werden!

Bevor man eine Partition oder komplette Platte sichert sollte diese ausgehängt werden, um sicherzustellen, dass während des Sicherungsvorgangs keine Daten auf die zu sicherende Platte geschrieben werden.

Zur Übernahme eines bestehenden Systems auf eine SSD (Solid-State-Drive) sollte **dd** nur mit äußerster Vorsicht genutzt werden. In den Standardeinstellungen verwendet **dd** eine Blockgröße von 512 Bytes, was bei modernen SSD zu unnötigen Schreibprozessen führt. Verwendet man unter Benutzung des Parameters `bs=` eine Blockgrößenangabe die der Blockgröße oder einem Vielfachen davon der SSD entspricht, besteht diese Gefahr nicht. Des Weiteren sollte man beachten, dass das Alignment (Ausrichtung) eingehalten wird, was ohne weitere Parameter höchstwahrscheinlich nicht der Fall ist.

`dd if=Quelle of=Ziel <Optionen>`

if ... Steht für "Input File", kann ein komplettes Gerät (z.B. `/dev/sda`), eine Partition oder eine Datei sein.

of ... Steht für "Output File", kann ein komplettes Gerät (z.B. `/dev/sdb`), eine Partition oder eine Datei sein.

dd kann ohne Root-Rechte aufgerufen werden. Man benötigt nur dann Root-Rechte, wenn von einem Gerät bzw. einer Partition gelesen bzw. darauf geschrieben werden soll, auf die nur Root Zugriff hat. Beim Lesen von CD/DVDs muss dd grundsätzlich mit Root-Rechten aufgerufen werden.

Wird if bzw. of weggelassen, so liest dd von der Standardeingabe bzw. schreibt auf die Standardausgabe. Dies ist dann nützlich, wenn dd in Kombination mit dem Pipe-Operator genutzt wird.

dd kann zwar grundsätzlich auch Dateien kopieren, allerdings ist hier in der Regel der Befehl cp komfortabler.

Optionen:

bs=BYTES ... Es werden Blöcke mit der Größe BYTES gelesen und geschrieben. Wird **bs** als Option benutzt, so ist **ibs = obs = bs**.
count=BLOCKS ... BLOCKS gibt an, wie viele Blöcke mit der durch **bs** / **obs** / **ids** festgelegten Größe gelesen und / oder geschrieben werden.
skip=BLOCKS ... BLOCKS gibt an, wie viele Blöcke der mit **ibs** oder **bs** festgelegten Größe zu Beginn des Lesevorgangs übersprungen werden sollen.

ibs=BYTES ... Es werden Blöcke der Größe BYTES gelesen.

obs=BYTES ... Es wird in Blöcken mit der Größe BYTES geschrieben.

seek=BLOCKS ... BLOCKS gibt an, wie viele Blöcke der mit **obs** oder **bs** festgelegten Größe zu Beginn des Schreibvorgangs übersprungen werden.

Für die Angaben BYTES und BLOCKS gilt:

- BYTES muss ganzzahlig sein. Ohne weiteres Suffix wird die Größe der Zahl BYTES in Byte interpretiert.
- BLOCKS muss ganzzahlig sein.

Des Weiteren kennt dd noch verschiedene andere Optionen, insbesondere zum Konvertieren der Daten zwischen Einlesen und Ausgabe. Diese werden bei "normaler" Benutzung eher selten gebraucht, können aber in den Manpages von dd nachgelesen werden.

Hinweis: Wenn man keine Blockgröße angibt verwendet dd eine kleine Standardgröße, was den Datentransfer durch den Overhead sehr langsam macht. Insofern ist es empfehlenswert, z.B. **bs=1M** anzugeben.

Suffixe für BYTES

Wie oben bereits erwähnt wird die Größe der Zahl BYTES standardmäßig in Byte interpretiert. Diese kann durch Hinzufügen von Suffixes geändert

werden.

b ... 512 Byte
kB ... 1000 Byte
K ... 1024 Byte
MB ... 1000000 Byte
M ... 1048576 Byte
GB ... 1000000000 Byte
G ... 1073741824 Byte

Gemäß dem in der Tabelle aufgezeigten Schema gibt es auch die Suffixe TB, T, PB, P, EB, E, ZB, Z, YB, Y - für alle, die wirklich große Datenmengen kopieren müssen.

Die gleichen Suffixe gelten auch für BLOCKS, d.h. z.B. mit **count=1K** werden 1024 Blöcke gelesen/geschrieben, mit **count=1MB** 1000000 Blöcke, usw.

Anwendungen

Hinweis: Da sich die Geräte-Bezeichnungen wie `/dev/sda` nach jedem Bootvorgang ändern können, sind vor der Verwendung von `dd` stets die aktuellen Gerätedateien zu überprüfen. Dies kann man zum Beispiel mit **sudo blkid** machen.

dd if=/dev/sda5 of=/dev/sdb1 ... Es wird die komplette fünfte Partition von `/dev/sda` in die erste Partition von `/dev/sdb` kopiert.

dd if=/dev/sdb1 of=/dev/sdc2 bs=1K count=10 ... Es werden die ersten zehn 1024 Byte großen Blöcke von der erste Partition von `/dev/sdb` auf die zweite Partition von `/dev/sdc` kopiert.

dd if=/dev/sr0 of=~/.dvd_image.img bs=2M ... Image von einer eingelegten DVD (Gerätedatei: `/dev/sr0`) erstellen; mit der optionalen Angabe der Blockgröße (`bs=2M`) wird der Vorgang ein wenig beschleunigt; das Image wird im Home-Verzeichnis abgelegt (~)

dd if=iso_image.iso of=/dev/sdd ... ISO-Image (z.B. ISO-Image einer Linux-Live-Distribution) auf ein USB-Stick schreiben; die Geräte-Datei des USB-Sticks ist ohne Partitionsziffer einzutragen; statt `/dev/sdd1` ist hier im Beispiel nur `/dev/sdd` einzutragen

Hinweis: Alle Daten auf dem USB-Datenträger werden unwiederbringlich gelöscht.

dd if=/dev/sda3 of=/dev/sda4 ibs=2KB obs=2KB skip=50 ... Es werden 2000 Byte große Blöcke von der dritten Partition von **/dev/sda** auf die vierte Partition von **/dev/sda** kopiert, wobei beim Einlesen die ersten 50 Blöcke (in diesen Fall $50 * 2000 = 100.000$ Byte) übersprungen werden, d.h. der Lesevorgang fängt bei Byte 100.001 an.

Festplatte klonen

Der folgende Befehl kloniert (kopiert) die komplette Festplatte **/dev/sda** inklusive aller Partitionen, MBR und Partitionstabelle auf die eine zweite Festplatte **/dev/sdb**:

dd if=/dev/sda of=/dev/sdb

Hinweis: Es sollte darauf geachtet werden, dass die beiden Festplatten gleich groß sind - oder zumindest das Ziel größer.
Falls man plant beide Platten gleichzeitig im selben PC zu betreiben, ist darauf zu achten, dass die **UUIDs** (Universally Unique Identifiers) der geklonten Platte geändert werden, da es sonst zu Konflikten kommt.
Komprimiert man ein solches Festplattenimage, wie im folgenden Absatz beschrieben, noch zusätzlich mit **gzip**, so sollte man vorher die Ausgabe von **fdisk -l** speichern und mit der gesicherten Imagedatei zusammen aufheben. Alternativ kann man die Startpositionen der Partitionen auch – sehr zeitaufwändig – aus dem gepackten Image auslesen.

Festplatte klonen – mit Angabe der Blockgröße und Zahl

fdisk -l ... ermitteln der Blockgröße und die Anzahl der Zylinder

Ausgabe:

Platte **/dev/hda**: 40.0 GByte, 40020664320 Byte

255 Köpfe, 63 Sektoren/Spuren, 4865 Zylinder

Einheiten = Zylinder von $16065 \times 512 = 8225280$ Bytes

[...]

Jetzt kann z.B. eine externe Festplatte über USB an das System angeschlossen werden. Mit dem folgenden Kommando wird die Festplatte **hda** auf die externe Festplatte kopiert.

dd if=/dev/hda of=<Pfad zur externen Festplatte>/hda.img bs=16065b count=4865

Das Zeichen b steht dabei für 512 Bytes (weitere Zeichen KB, MB etc. siehe weiter oben).

Restore, Wiederherstellung der Festplatte

dd if=<Pfad zur externen Festplatte>/hda.img of=/dev/hda

Festplatte (sicher) löschen

Hinweis: Alle Daten auf der Festplatte werden unwiderruflich gelöscht!

Der folgende Befehl löscht die komplette Festplatte **/dev/sda** durch Überschreiben mit Nullen:

dd if=/dev/zero of=/dev/sda conv=noerror

conv=noerror ... bei fehlerbehafteten Sektoren stellt dd standardmäßig seine Arbeit sofort ein, die Option noerror verhindert dies

Der folgende Befehl löscht die komplette Festplatte **/dev/sda** durch Überschreiben mit Zufallszahlen (zeitintensiv):

dd if=/dev/urandom of=/dev/sda

In Kombination mit dem Programm **pv** kann eine Statusanzeige ausgegeben werden (Überschreiben durch Nullen des Datenträgers '/dev/sdb'):

dd if=/dev/zero conv=noerror,notrunc,sync bs=10240 | pv -S > /dev/sdb

conv=notrunc ... Output File wird nicht gekürzt

conv=sync ... synchronisiert die Daten (Arbeitsspeicher und Festplatte bzw. andere Massenspeicher)

Anmerkung: Der Löschprozess kann maßgeblich beschleunigt werden, wenn man den Buffer (internen Zwischenspeicher) der Festplatte ausnutzt. Wie groß dieser für die aktuelle Platte ist, kann mit **sudo hdparm -i /dev/sdX** herausgefunden werden. Hier ist es der Wert BufferSize=, welcher exakt so (ohne kB) für den dd Parameter bs übernommen werden kann. Zudem kann der Löschprozess in den Hintergrund gelegt werden, um während der Durchführung des Löschvorgangs eine Fortschrittsanzeige auszugeben:

Für die Festplatte sda mit 8 MiB BufferSize sieht dann so aus:

```
dd if=/dev/zero of=/dev/sda bs=8M & pid=$!
```

Um nun die Fortschrittsanzeige auszugeben, kann folgendes Kommando auf derselben Konsole eingegeben werden:

```
while true; do kill -USR1 $pid && sleep 1 && clear; done
```

Partitionen klonen

Der folgende Befehl kloniert (kopiert) die komplette Partition **/dev/sda1** auf die Partition **/dev/sdb1**:

```
dd if=/dev/sda1 of=/dev/sdb1
```

Hinweis: Es sollte darauf geachtet werden, dass die beiden Partitionen gleich groß sind.

Partition in einem Image sichern

Der folgende Befehl erstellt ein Image von **/dev/sda1** in die Datei **image_sda1.img**, welche im Home-Verzeichnis gespeichert wird:

```
dd if=/dev/sda1 of=~/.image_sda1.img
```

Diese Art der Sicherung ist nicht wirklich zu empfehlen, da die Image-Datei die gleiche Größe wie die gesicherte Partition hat. Daher ist es sinnvoller, das Image zu komprimieren. Der folgende Befehl erstellt ein komprimiertes Image der Partition **/dev/sda1** und speichert dieses in die Datei **image-compress_sda1.img.gz** im Heimatverzeichnis. Durch das Weglassen von **of** im Befehlsaufruf werden die Daten auf die Standardausgabe geschrieben, wo sie dann per Pipe-Operator an **gzip** weitergeleitet werden:

```
dd if=/dev/sda1 | gzip > ~/.image-compress_sda1.img.gz
```

Hinweis: Im Regelfall reichen die Standardeinstellungen von **gzip** aus. Möchte man dennoch die beste Kompressionsstufe, so lautet der zu verwendende Befehl **gzip -9**. Zu beachten ist, dass bei hohen Kompressionsstufen sich der Zeitaufwand merklich erhöht (teilweise um ein Vielfaches), während der Speicherverbrauch nur geringfügig abnimmt.

Auch wenn das Image von **gzip** mit der höchsten Kompressionsstufe komprimiert wird, kann die Ausgabedatei unter Umständen trotzdem sehr

groß werden. Man sollte also auf ausreichend Platz auf dem Zieldatenträger achten!

Um das so erzeugte komprimierte Image wieder zurückzusichern, kann man folgenden Befehl verwenden:

```
gunzip -c ~/image-compress_sda1.img.gz | sudo dd of=/dev/sda1
```

Dateigröße des Images begrenzen

Für den Fall, dass zum Beispiel das Dateisystem des Ziellaufwerkes eine Dateigrößenbeschränkung hat, besteht die Möglichkeit zum Splitten der Imagedatei. Auf einem FAT32-Laufwerk beispielsweise ist die Dateigröße auf 4 GiB (1 GiB = 1024 * 1024 * 1024 Byte) beschränkt.

In folgendem Beispiel wird das Image an **split** übergeben und in Teile von je 3500 MiB (1 MiB = 1024 * 1024 Byte) gespeichert. Hierbei werden die jeweiligen Teile numerisch beschriftet.

```
dd if=/dev/sda1 | split -d -b 3500M - ~/image_sda1.img
```

Hinweis: Nicht den Punkt (.) hinter **sda1.img** vergessen! Dahinter steht dann die Folgenummer der Datei. Das Ergebnis sieht dann in diesem Fall so aus:

- image_sda1.img.00
- image_sda1.img.01
- image_sda1.img.02
- image_sda1.img.03

Zurückgespielt wird dann, indem das Image durch **cat** automatisch wieder zusammengefügt und an **dd** übergeben wird.

```
cat ~/image_sda1.img.* | dd of=/dev/sda1
```

MBR: Boot-Loader und Partitionstabelle sichern

Der MasterBootRecord (MBR) beherbergt den Boot-Loader, die Partitionstabelle und die MBR-Signatur. Der MBR ist exakt 512 Bytes lang und liegt am Beginn der Festplatte. Der Boot-Loader belegt die ersten 446 Bytes des MBR, dann folgen die Partitionstabelle (64 Bytes) und die MBR-Signatur, und, Achtung, zumindest GRUB nutzt je nach Konfiguration meist noch einige weitere Sektoren im sog. verborgenen Bereich vor der ersten

Partition.

Zur Sicherung ist ein geeignetes Medium notwendig. Nutzer einer LiveCD (z. B. Ubuntu Installations-CD) müssen zunächst ein Medium verfügbar machen:

sudo fdisk -l

zeigt die Bezeichnungen der eigenen Festplatten an und dient als Orientierungshilfe für folgende Kommandos.

Nun erstellt man einen Ordner im Dateisystem der Live-CD und hängt dort eine Partition ein, auf welcher die Sicherung des MBR erstellt wird.

```
sudo mkdir ~/sda3
```

```
sudo mount /dev/sda3 ~/sda3
```

```
cd ~/sda3
```

Es kann sich hierbei auch um einen USB-Stick, eine Netzwerkfreigabe oder ein anderes Medium handeln, auf welches man jederzeit Zugriff hat. Jetzt kann mit dem eigentlichen Sichern begonnen werden.

Achtung: Die Sicherung der Partitionstabelle mittels dd ist mit Vorsicht anzuwenden, da:

1. mit dem MBR nur die primären Einträge der Partitionstabelle (Bytes 446..509) gesichert werden. Die Einträge zu den logischen Partitionen stehen in den kaskadierten BRs (BootRecords) der erweiterten Partition, und fehlen damit hier komplett.
2. beim Einsatz einer GPT (GUID Partition Table) überhaupt keine Sicherung der Partitionstabelle erfolgt. Außerdem wird der Bootloader in eine eigene Boot-Partitionen installiert.

Um immer auf der sicheren Seite zu sein, empfiehlt es sich nach jeder Partitionsänderung die entsprechenden Tabellen neu zu sichern. Bei der MBR-Partitionstabelle kann man dazu das Programm **sfdisk** und bei der GUID-Partitionstabelle das Programm **sgdisk** nutzen. Spielt man eine alte MBR-Sicherung (mit alter und somit falscher Partitionstabelle) zurück, kann man auf die komplette Platte höchst wahrscheinlich nicht mehr zugreifen.

Hinweis: Die folgenden Anweisungen zum Sichern des MBR sollten nur als Muster der Möglichkeiten von dd verstanden werden.

Mit dem folgenden Befehlsaufruf würde der Boot-Loader der Festplatte **/dev/sda** als Datei **bootloader_sicherung** im aktuellen Verzeichnis gesichert. Die Partitionstabelle (Bytes 446 ... 509) und die MBR-Signatur sind in dieser Sicherung nicht enthalten:

```
sudo dd if=/dev/sda of=bootloader_sicherung bs=446 count=1
```

Der folgende Befehl sichert den gesamten MBR (inklusive Partitionstabelle) der Festplatte **/dev/sda** als Datei **mbr_sicherung.img** im aktuellen Verzeichnis:

```
sudo dd if=/dev/sda of=mbr_sicherung.img bs=512 count=1
```

Bei installiertem Boot-Manager, z.B. GRUB 2, sollte man ggf. auch den sog. verborgenen Bereich hinter dem MBR, falls dieser (Regelfall) dafür verwendet wird, ebenfalls sichern. Vorher sollte man in Erfahrung bringen (**fdisk -l**), wie viele Sektoren vor der ersten Partition frei sind:

```
sudo fdisk -l
```

Bei heutigen Festplatten sind das meist die Sektoren 0 - 62 (erste Partition beginnt also bei 63, seit Windows-Vista aber auch häufig erst bei Sektor 2048 (von der Sektorgröße der Festplatte abhängig)). Dann schaut man noch wie viele Bytes ein Sektor hat (meist 512) und passt den Befehl entsprechend an:

```
sudo dd if=/dev/sda of=mbr+grub_sicherung bs=512 count=63
```

Eine Sicherung des Boot-Loaders wird mit

```
dd if=boots_sicherung of=/dev/sda bs=446 count=1
```

zurückgespielt. Dieses Kommando kann auch unter Verwendung einer kompletten Sicherung des MBR bzw. obiger **mbr+grub_sicherung** verwendet werden: Es wird nur der Boot-Loader (Bytes 0..445) wiederhergestellt, die momentane Partitionstabelle (Bytes 446..509) und die MBR-Signatur bleiben dann in jedem Fall erhalten.

Will man den kompletten MBR (also inklusive Partitionstabelle) zurücksichern, so lautet der Befehl wie folgt:

```
dd if=mbr_sicherung.img of=/dev/sda bs=512 count=1
```

Will man im Fall eines installierten Boot-Managers zusätzlich auch diesen (falls im sog. verborgenen Bereich hinter dem MBR abgelegt) zurücksichern (unter Erhalt des zuvor zurückgesicherten Boot-Loaders und der Partitionstabelle), so lautet der Befehl wie folgt (falls die erste Partition bei Sektor 63 beginnt, siehe detaillierteren Hinweis weiter oben):

```
dd if=mbr+grub_sicherung of=/dev/sda bs=512 skip=1 seek=1 count=62
```

Mit dd erstellte Images einbinden

Image einer Partition einbinden

Ein mit dd erstelltes Image lässt sich als Loop-Device mit dem Befehl **mount** einbinden. So kann auf das Image wie auf ein normales Laufwerk zugegriffen werden. Dazu erstellt man als erstes ein Image, hier z.B. vom Device **/dev/sda1**, gespeichert in der Datei **loop_image.img** im Home-Verzeichnis:

```
dd if=/dev/sda1 of=~/.loop_image.img
```

Dann erzeugt man einen Einhängpunkt, z.B. **/media/loop_mount**:

```
sudo mkdir /media/loop_mount
```

Jetzt kann man das mit dd erzeugte Image mit **mount** einbinden:

```
sudo mount -o loop ~/.loop_image.img /media/loop_mount
```

Nun kann man auf alle Dateien, Verzeichnisse etc. des Images wie auf ein reguläres Laufwerk zugreifen. Nach der Benutzung muss man das Image dann wieder mit **umount** aushängen:

```
sudo umount /media/loop_mount
```

Bei Bedarf kann das (bearbeitete) Image jetzt auch wieder zurück gesichert werden.

Hinweis: ISO-Images müssen über den mount-Parameter **loop** in den Linux-Dateibaum eingebunden werden.

Partition aus einem Image der gesamten Platte einbinden

Hat man nicht nur eine Partition, sondern die gesamte Festplatte inklusive MBR gesichert, braucht man den Offset der jeweiligen Partition. Diesen

kann man mit dem Befehl

sudo fdisk -l /Pfad/zum/Image.img

herausfinden. Die Ausgabe sieht bei 3 primären Partitionen ungefähr so aus:

```
Platte /Pfad/zum/Image.img: 0 MByte, 0 Byte
255 Köpfe, 63 Sektoren/Spuren, 0 Zylinder, zusammen 0
Sektoren
Einheiten = Sektoren von 1 × 512 = 512 Bytes
Disk identifiziert: 0xd53d826f

Gerät      boot. Anfang      Ende      Blöcke      Id      System
/Pfad/zum/Image.img1 * 63 104872319 52436128+ 7
HPFS/NTFS
Partition 1 hat unterschiedliche phys./log. Enden:
    phys=(1023, 254, 63) logisch=(6527, 254, 63)

/Pfad/zum/Image.img2 104872320 109065284 2096482+ 82
Linux Swap/Solaris
Partition 2 hat unterschiedliche phys./log. Anfänge (nicht-
Linux?):
    phys=(1023, 0, 1) logisch=(6528, 0, 1)
Partition 2 hat unterschiedliche phys./log. Enden:
    phys=(1023, 254, 63) logisch=(6788, 254, 63)

/Pfad/zum/Image.img3 109065285 156296384 23615550 83
Linux
Partition 3 hat unterschiedliche phys./log. Anfänge (nicht-
Linux?):
    phys=(1023, 0, 1) logisch=(6789, 0, 1)
Partition 3 hat unterschiedliche phys./log. Enden:
    phys=(1023, 254, 63) logisch=(9728, 254, 63)
```

Der Wert hinter der entsprechenden Partition unter Anfang, ist der Offset, dieser muss jedoch noch mit der weiter oben angegebenen Sektorgröße multipliziert werden (hier 512). Der Offset für die 3. Partition wäre also $109065285 * 512 = 55841425920$. Nun Folgt der Mountbefehl mit dem entsprechenden Offset (hier wieder am Beispiel der 3. Partition):

sudo mkdir /media/loop_mount # Verzeichniss anlegen

**sudo mount -o loop,offset=55841425920 /Pfad/zum/Image.img
/media/loop_mount**

Zum Schluss wird das Image wieder freigegeben mit:

sudo umount /media/loop_mount

Hinweis: ISO-Images müssen über den mount-Parameter loop in den Linux-Dateibaum eingebunden werden.

Image im Netzwerk speichern

Ein mit dd erstelltes Image muss nicht zwangsläufig lokal gespeichert werden, sondern kann auch auf einen anderen Rechner im Netzwerk gesichert werden.

Im folgenden Beispiel wird mit dd ein Image von **/dev/sda1** erstellt, welches dann ssh-verschlüsselt auf den Rechner mit der IP-Adresse 192.168.0.100 übertragen und dort im Verzeichnis **/home/BENUTZER** in der Datei **image_sda1.img** gespeichert wird.

Damit dies funktioniert, muss "BENUTZER" ein Benutzerkonto auf dem entsprechenden Rechner haben und man selbst die notwendigen Rechte, um dort zu schreiben. Der Befehlsaufruf lautet:

Gzip-komprimiert und ssh-verschlüsselt

```
dd if=/dev/sda1 | gzip -9 - | ssh user@192.168.0.100 "cat > /home/BENUTZER/image_sda1.img.gz"
```

Um das Image zurückzusichern (z.B. auf den Rechner mit der IP-Adresse 192.168.0.50), gibt man folgenden Befehl ein:

```
ssh user@192.168.0.50 "cat /home/user/image_sda1.img.gz" | gunzip -c - | dd of=/dev/sda1
```

Bzip2-komprimiert und nicht verschlüsselt

Alternativ mit bzip komprimiert, aber im Transfer nicht:

Auf dem Zielrechner:

```
netcat -l -p 5555 | dd of=/home/user/image_sda1.img bs=16065b
```

Auf dem Quellrechner:

```
dd if=/dev/sda1 bs=16065b | pv | bzip2 -1 | netcat ZielIP 5555
```

FTP mit gzip Komprimierung und nicht verschlüsselt

Erstellen eines Images über FTP:

```
dd if=/dev/sda bs=4k | gzip -9 - | ncftpput -c -V -u FTPUSER -p  
FTPPASSWORD FTPSERVER /FTPPATH/NAME.img.gz
```

Und zum Einspielen vom erstellten Images:

```
ncftpget -c -V -u FTPUSER -p FTPPASSWORD FTPSERVER  
/FTPPATH/NAME.img.gz | gunzip -c - | dd of=/dev/sda bs=4k
```

Fortschritt von dd abfragen

Einmalige oder regelmäßige Abfrage mittels Senden des Signals -USR1 Informationen zum ermitteln von Prozessen siehe auch: man ps, pgrep oder pidof und zum senden von Signalen kill, pkill oder killall.

Wenn das dd-Kommando einmal abgesetzt wurde, wünscht man sich bei größeren Kopiervorgängen eine Kontrollmöglichkeit über den Fortschritt. Dies erreicht man indem man dem dd-Prozess das Signal -USR1 sendet.

ps und kill

In einem zweiten Terminal, ermittelt man die Prozessnummer, z.B. mit

```
ps -a
```

und setzt ein Signal -USR1 ab.

```
kill -USR1 <Prozessnummer>
```

Die bisher kopierte Datenmenge wird dann in dem Terminal angezeigt, in dem dd gestartet wurde. Mit einer Kombination aus dd und einer Schleife kann man dies auch automatisieren.

```
dd if=/dev/XXX of=/dev/XXX & ddpid=$! ; while [ $(ps -a | grep  
$ddpid) ]; do kill -SIGUSR1 $ddpid; sleep 10; done
```

pkill

Sofern nur ein Prozess mit dem Namen dd läuft kann man dies auch über den Prozessnamen.

pkill -USR1 -x dd

In einer Schleife kann das so aussehen:

```
dd if=/dev/XXX of=/dev/XXX & while [ ! $(pkill -USR1 -x dd) ]; do  
sleep 10 ; ; done
```

pv

Als Alternative zur oben beschriebenen Vorgehensweise kann man auch den Befehl `pv` verwenden, um sich den Fortschritt anzeigen zu lassen (`pv` muss vorher noch installiert werden). Dabei wird `pv` mittels einer Pipe dazwischen geschaltet, als Beispiel wird hier Festplatte klonen angenommen.

```
dd if=/dev/sdx bs=1M | pv | dd of=/dev/sdy bs=1M
```

Um sich anzeigen zu lassen, wie weit der Vorgang fortgeschritten ist und wann er voraussichtlich beendet sein wird, muss man allerdings die Größe der Partition bzw. der Festplatte kennen. Im folgenden Beispiel wird eine Festplatte mit 60 GB unterstellt und in eine Image-Datei geschrieben:

```
dd if=/dev/sdx bs=1M | pv -s 60G | dd of=/pfad/zum/Ziel/backup.img  
bs=1M
```

Die Ausgabe kann dann so aussehen:

```
24.2GB 0:44:21 [4.45MB/s] [=====> ] 40% ETA 1:05:38
```

ETA zeigt dabei die verbleibende Zeit an, bis die Operation (bei aktueller Geschwindigkeit) voraussichtlich fertig sein wird.

Live USB-Stick erstellen

Mit `dd` lässt sich auch auf einfachste Art und Weise ein Live USB-Stick (als Ersatz für eine Live-CD) erstellen. Zwingende Voraussetzung ist allerdings ein entsprechendes Hybrid-ISO-Image. Die Live-ISO-Images von Ubuntu und seinen Varianten sind erst ab Ubuntu 11.10 Hybrid-ISO-Images. Auch viele andere Linux-Distributionen stellen diese zur Verfügung.

Im folgenden Beispiel wird davon ausgegangen, dass der USB-Stick als `/dev/sdc` erkannt wurde und nicht eingebunden ist (aber bitte nicht auswerfen bzw. "sicher entfernen"):

```
sudo dd if=hybrid_iso_image.iso of=/dev/sdc bs=1M
```

Den Parameter bs=1M kann man auch weglassen, aber er beschleunigt den Kopiervorgang. Nach dem Befehl

sync

kann das Medium entfernt werden.

sync ... Normalerweise verwendet Linux einen Puffer (Cache) im Arbeitsspeicher, in dem sich ganze Datenblöcke eines Massenspeichers befinden. So werden Daten häufig temporär erst im Arbeitsspeicher verwaltet, da sich ein dauernd schreibender Prozess äußerst negativ auf die Performance des Systems auswirken würde.

Mit dem Kommando sync können Sie nun veranlassen, dass veränderte Daten sofort auf die Festplatte (oder auf jeden anderen Massenspeicher) geschrieben werden.

siehe auch: man dd, ddrescue, man pv

ddrescue

Trifft das Programm ddrescue beim Auslesen auf einen defekten Block, bricht das Tool anders als dd nicht mit einem Fehler ab, sondern setzt seine Arbeit beim nächsten verwertbaren Block fort. Damit eignet sich ddrescue schon deutlich besser zur Rettung von Daten auf defekten Medien.

ddrescue [OPTIONEN] QUELLE ZIEL [LOGDATEI]

ddrescue -b2048 /dev/sr0 output.iso logfile.log ... eingelegte CD oder DVD auslesen und die Daten in die Datei output.iso speichern

-n ... (engl.: no-split), fehlerhafte Bereiche nicht teilen oder mehrfach zu lesen versuchen

-b ... Blockgröße in Bytes - 512 ist Standard; CD-ROM und DVD 2048 Bytes (z.B. -b4096, Achtung: vor der Zahl ist **kein** Leerzeichen einzutragen)

-r ... maximale Versuche, um die Daten zu retten; bei -1 wird das Programm so lange ausgeführt, bis alle Fehler behoben wurden (z.B. -r-1, Achtung: vor der Zahl ist **kein** Leerzeichen einzutragen)

-f ... ein Gerät oder Partition überschreiben

-v ... ausführliche Meldungen

Grundsätzlich sollte bei jedem Befehl eine Logdatei angegeben werden, erst mit dieser Datei zeigt das Programm alle Fähigkeiten, die es hat. Die Logdatei wird schreibend und lesend von ddrescue verwendet und ausgewertet.

Das Programm **ddrescue** kopiert ein Medium, indem es große Blöcke einliest und kopiert. Die defekten Sektoren übergeht **ddrescue** zunächst und notiert das Auslassen der defekten Blöcke in einer Logdatei. Erst nachdem das ganze Medium durchlaufen wurde, schaut sich **ddrescue** die defekten Sektoren erneut an und unterteilt die großen defekten Bereiche in mehrere kleine Bereiche. Diese kleineren Bereichen versucht **ddrescue** zu lesen und zu kopieren.

Durch die Log-Datei kann eine Rettung jeder Zeit einfach abgebrochen (Tastenkombination: [Strg] + [C]) und später fortgesetzt werden.

Ein guter Ansatz ist es, zunächst einmal zu versuchen alles zu sichern, was zum Zeitpunkt der Sicherung fehlerfrei ist, und keine Zeit auf fehlerhafte Blöcke zu verwenden:

ddrescue -n QUELLE ZIEL ddrescue.log

Erst danach startet man einen weiteren Durchlauf, in dem nun versucht wird, möglichst viele von den im ersten Schritt als defekt markierten Daten doch noch zu retten:

ddrescue QUELLE ZIEL ddrescue.log

Dieses zweigeteilte Vorgehen ist zu bevorzugen, da durch die intensive Beanspruchung, noch Daten aus den defekten Blöcke zu bekommen, auch andere Teile des Speichermediums zerstört werden können.

Beispiel 1: eine SSD-Festplatte zeigt ein unerklärliches Verhalten

- Erscheinen beim nachfolgenden Befehlsaufruf ERROR-Meldungen (read error ... , I/O error ...), so sollte man so schnell wie möglich handeln. In den Ausgabemeldungen von **dmesg** sucht **grep** nach dem Suchbegriff **sd**.

dmesg | grep sd

Um weitere Schäden abzuwenden, sollte die betroffene Festplatte aus dem Linux-Verzeichnisbaum ausgehängen werden (**siehe auch:** **umount**). Ist die Systempartition betroffen, so ist der Rechner herunterzufahren und die Partition über einen Neustart mit einem Linux-Live-System (CD, DVD) einzuhängen.

Mittels des Befehls **lsblk** bzw. **sudo fdisk -l** wird eine Übersicht der angeschlossenen Datenträger angezeigt.

- Es ist empfehlenswert an **ddrescue** auch die Blockgröße der betroffenen Partition zu übergeben. Im nachfolgenden Beispiel, liegt das Home-Verzeichnis auf einer separaten Partition.

stat /home -f | grep Block

- **Durchlauf 1:** Sicherung der Daten aus den unbeschädigten Bereichen

sudo ddrescue -n -b4096 /dev/sda2 home.img ddrescue.log

- **Durchlauf 2:** Versuch die Daten aus den fehlerbehafteten Bereichen zu sichern; es erfolgen durch **ddrescue** maximal 4 Leseversuche

sudo ddrescue -r4 -b4096 /dev/sda2 home.img ddrescue.log

Anschließend kann das Image »home.img« über den Befehl **mount** in den Linux-Verzeichnisbaum eingehangen werden.

sudo mount -t ext4 -o ro,loop home.img /media/ddrescue

Das Verzeichnis »ddrescue« muss vorher mit ROOT-Rechten oder mittels **sudo** erst erstellt werden. Es kann auch ein anderes bereits existierendes Verzeichnis als Mountpoint gewählt werden. Mit dem Befehl **umount** wird das Image »home.img« wieder aus dem Linux-Verzeichnisbaum ausgehangen.

sudo umount /media/ddrescue

Beispiel 2: Daten von einem USB-Stick retten

- Name des USB-Sticks und die Blockgröße der Partition auf dem USB-Stick ermitteln

```
linux@tux ~ $ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sdb   8:16  0 298,1G 0 disk
└─sdb1 8:17  0 93,1G 0 part /
[...]
```

```
sdd    8:48  1 245,5M 0 disk
```

```
└─sdd1  8:49  1 245,5M 0 part /media/linux/MIGHTYDRIVE
```

```
linux@tux ~ $ stat /media/linux/MIGHTYDRIVE -f | grep Block
```

Blockgröße: 4096 Fundamentale Blockgröße: 4096

Mit dem Befehl **umount** wird der USB-Stick aus dem Linux-Verzeichnisbaum wieder ausgehängen.

```
linux@tux ~ $ umount /media/linux/MIGHTYDRIVE
```

- Sicherung der Daten aus den unbeschädigten Bereichen des USB-Sticks

```
sudo ddrescue -n -b4096 /dev/sdd1 usbstick.img usbstick.log
```

In diesem Beispiel wird darauf verzichtet, Daten aus den beschädigten Bereichen des USB-Sticks zu sichern.

- Zum Auslesen der geretteten Daten wird das Image »usbstick.img« in einem neu erstellten Mountpoint »usbstick-gerettet« eingehangen.

```
mkdir ~/usbstick-gerettet
```

```
sudo mount -o loop usbstick.img ~/usbstick-gerettet
```

Mit dem Befehl **umount** wird das Image »usbstick.img« aus dem Linux-Verzeichnisbaum wieder ausgehängen.

```
umount ~/usbstick-gerettet
```

siehe auch: man ddrescue, stat, dmesg, grep, lsblk, mount, umount

df

df informiert sie, wie viel Platz noch auf ihren Partitionen frei ist. Mit df -h werden die Bytes in für Menschen lesbarer Form angezeigt (h = human readable).

df -T ... gibt den Typ der Dateisysteme aus, wie z.B. ext3, ext4, reiserfs oder vfat aus

siehe auch: df --help

du

Zeigt den Platzverbrauch des aktuellen Verzeichnisses in Byte an. Mit du -h werden die Bytes in für Menschen lesbarer Form angezeigt (h = human readable).

du -bhs ... gibt die Gesamtsumme der Dateien des aktuellen Verzeichnisses aus, z.B. 464 K

du -bs ... gibt die Gesamtsumme der Dateien des aktuellen Verzeichnisses in Byte aus, z.B. 464 413

du -bs ./<Verzeichnisname> ... gibt die Gesamtsumme der Dateien des aktuellen Verzeichnisses <Verzeichnisname> in Byte aus, z.B. 464 413

du -sch ... gibt die Gesamtsumme des Platzverbrauchs auf der Festplatte aus - die Summe ist meistens größer als die Gesamtsumme der einzelnen Dateien (du -bs), da die einzelnen Speicherblöcke bei kleinen Dateien häufig nur teilweise belegt sind

du -sh --exclude=mu* ... Gesamtsumme des Platzverbrauchs im aktuellen Verzeichnis samt seiner Unterverzeichnisse mit Ausnahme der Verzeichnisse die mit **mu** beginnen, also z.B. musketiere, music, mutter etc.

du /home/<benutzername>/bin | sort -rn | less ... die beiden Parameter -rn, sorgen dafür, dass das Programm die größten Verzeichnisse oder Dateien zuerst anzeigt

siehe auch: du --help

Dienste starten, anhalten und deaktivieren

In Linux sind Dienste (Services) vom System zum Startzeitpunkt automatisch ausgeführte Prozesse, die zum Betrieb unerlässlich sind oder Serverdienste (Webserver, Samba, Druckserver ...) bereitstellen. Um die Dienste kümmert sich der Init-Prozess, der bei den verschiedenen Linux-Distributionen in abweichenden Varianten vorliegt.

Debian:

Debian nutzt das System-V-Init, das von Unix geerbt wurde.

/usr/bin/service --status-all ... Auflistung aller Dienste, keine root-Rechte erforderlich

Laufende Dienste haben ein Plus-Zeichen (+) und deaktivierte eine Minus-Zeichen (-). Statuslose Systemdienste sind mit einem Fragezeichen (?) versehen. Für die nachfolgenden Befehle zum Starten, Anhalten und

Neuladen (start, stop, restart, reload) sind root-Rechte erforderlich.

/usr/bin/service [Dienstname] start ... Dienst starten bzw.

sudo /etc/init.d/[Dienstname] start ... Dienst starten

/usr/bin/service [Dienstname] stop ... Dienst anhalten bzw.

sudo /etc/init.d/[Dienstname] stop ... Dienst anhalten

/usr/bin/update-rc.d [Dienstname] disable ... Dienst dauerhaft deaktivieren

/usr/bin/update-rc.d [Dienstname] enable ... einen Dienst für den automatischen Start beim Booten aktivieren

Ubuntu:

Schon seit der Version 6.10 verwendet Ubuntu die Eigenentwicklung Upstart, das den parallelen Start von Diensten ermöglicht. Ubuntu startet aber immer noch einen Teil der Dienste auf dem alten Weg wie Debian mit Init-Skripts.

Was wie gestartet wird, zeigt nur ein Blick in die Verzeichnisse /etc/init und /etc/init.d. Ersteres enthält die Upstart-Dienste, letzteres die alten System-V-Init-Dienste. Laufende Dienste sind durch Angabe »start/running« gekennzeichnet.

initctl list Auflistung aller Dienste, keine root-Rechte erforderlich

sudo initctl start [Upstart-Dienstname] ... Dienst starten

sudo initctl stop [Upstart-Dienstname] ... Dienst anhalten

Zum dauerhaften Deaktivieren muss für einen Upstart-Dienst eine Datei mit den Namen [Upstart-Dienstname].override im Verzeichnis /etc/init angelegt werden.

sudo echo 'manual' | sudo tee -a /etc/init/[Upstart-Dienstname].override

Um den Dienst wieder ab dem Boot automatisch zu starten, ist es nur nötig, die Datei [Upstart-Dienstname].override im Verzeichnis /etc/init zu löschen.

CentOS:

Wie Ubuntu nutzt CentOS Upstart für einige Dienste, für andere dagegen System-V-Init. Über Upstart laufen i.d.R. nur Systemdienste, die im Normalbetrieb nicht geändert werden müssen. Alle anderen Dienste werden über das Programm service angesprochen.

service --status-all ... Auflistung aller Dienste, keine root-Rechte erforderlich

sudo service [Dienstname] start ... Dienst starten

sudo service [Dienstname] stop ... Dienst anhalten

sudo chkconfig [Dienstname] off ... Dienst dauerhaft deaktivieren

sudo chkconfig [Dienstname] on ... einen Dienst für den automatischen Start beim Booten aktivieren

OpenSuse:

OpenSuse verwendet Systemd zur Steuerung von Diensten.

systemctl -t service ... Auflistung aller Dienste, keine root-Rechte erforderlich; in der Liste mit den Pfeiltasten scrollen; Liste schließen über die Taste [Q]

sudo systemctl start [Dienstname].service ... Dienst starten

sudo systemctl stop [Dienstname].service ... Dienst anhalten

sudo systemctl disable [Dienstname].service ... Dienst dauerhaft deaktivieren

sudo systemctl enable [Dienstname].service ... einen Dienst für den automatischen Start beim Booten aktivieren

Hinweis: Systemd zur Steuerung von Diensten wird in Zukunft System-V-Init und Upstart ablösen.

siehe auch: Init (SysVinit), Upstart, systemd – Das Init-System

dmesg

dmesg (dump messages) zeigt die Bootmeldungen an.

Die ersten Sekunden des Bootvorganges können nicht auf der Festplatte geloggt werden (da zu diesem Zeitpunkt, noch kein beschreibbares

Dateisystem gemountet wurde). Wichtige Infos zur Fehlerbehebung könnten so verloren gehen. Darum speichert der Kernel diese Logmeldungen im Arbeitsspeicher zwischen und dmesg kann diese dort auslesen und anzeigen.

Einige Distributionen verfügen über ein Startskript, in dem das Programm dmesg aufgerufen und der Output nach /var/log/boot.log geschrieben wird.

siehe auch: man dmesg

dos2unix

siehe auch: fromdos, todos

DOS / Linux - Befehlsunterschiede

Die wichtigsten Befehle von MS-DOS und ihr Analogon in Linux/Unix. Das soll den Einstieg in die Kommandozeile ein wenig vereinfachen.

DOS	Linux	
attrib	chmod	... Dateiattribute (wie schreibgeschützt) ändern
chkdsk	fsck	... den Datenträger überprüfen
cd	cd	... Verzeichnis wechseln
cd	pwd	... Anzeigen der Position im Verzeichnisbaum
copy	cp	... Dateien kopieren
date, time	date	... Zeit und Datum anzeigen / ändern
del	rm	... Dateien löschen
deltree	rm -R	... einen Verzeichnisbaum löschen
diskcopy	dd	... ein Datenträger kopieren
dir	ls	... den Inhalt eines Verzeichnisses anzeigen
dir	df	... Anzeige des verwendeten Speichers von Dateisystemen
echo	echo	... einen Text auf dem Bildschirm ausgeben
fc	diff	... Dateien miteinander vergleichen
fdisk	fdisk	... Festplatte partitionieren
find	grep	... Dateien nach einem Schlüsselwort durchsuchen
find	find	... eine Datei suchen
format	mkfs	... einen Datenträger formatieren
md, mkdir	mkdir	... ein neues Verzeichnis erstellen
more	more, less	... den Inhalt einer Datei am Bildschirm ausgeben
move	mv	... Datei oder Verzeichnis verschieben
rd, rmdir	rmdir	... ein Verzeichnis löschen
rename	mv	... Datei oder Verzeichnis umbenennen
sort	sort	... Dateien oder Verzeichnisse sortieren

DOS	Linux	
type	cat	... den Inhalt einer Datei am Bildschirm ausgeben
xcopy	cp -a	... mehrere Dateien oder ganze Verzeichnisse kopieren

dosbox

Dosbox ist ein Emulator für DOS, mit dem man ältere DOS-Spiele oder DOS-Software wieder zum Laufen bringt - aber diesmal unter Linux. Dosbox ist Bestandteil vieler Linux-Distributionen.

In den nachfolgenden Beispielen gehe ich davon aus, dass die DOS-Programme im folgenden Verzeichnis gespeichert sind:

/home/<Benutzername>/bin/DOS/

Dies ist insofern wichtig, dass beim Aufruf der Dosbox dies Verzeichnis mit übergeben wird. Dosbox spricht dieses Verzeichnis dann während einer Sitzung als Laufwerk C: an.

dosbox /home/<Benutzername>/bin/DOS

Dosbox öffnet daraufhin ein Dosbox-Fenster, in den man dann wie gewohnt DOS-Befehle eingeben kann (dir, cd etc.). Das angegebene Verzeichnis ist jetzt innerhalb von Dosbox das Wurzelverzeichnis vom Laufwerk C:\

dosbox /home/<Benutzername>/bin/DOS -fullscreen

Dosbox öffnet sich daraufhin im Vollbildschirm-Modus.

mount d /media/cdrom -t cdrom -used 0

bzw.

mount d /media/cdrecorder -t cdrom -used 0

Mountet das CD-ROM Laufwerk als Laufwerk D:
Mit d: wird in das Laufwerk D:\ gewechselt.

config -writeconf dosbox.conf

Schreibt die aktuellen Einstellungen in die dosbox.conf. Die

Konfigurationsdatei wird im Home-Verzeichnis gespeichert.

Tastenkombinationen:

[Alt] + [ENTER] ... umschalten zwischen Fensterdarstellung und Vollbilddarstellung

[Strg] + [F5] ... Screenshot erstellen

[Strg] + [F9] ... kill dosbox

[Strg] + [F12] ... Geschwindigkeitserhöhung der Emulation (Increase DOSx Cycles). Diese Tastenkombination sollte man dann versuchen, wenn der Sound oder die Maus ruckelt. Die Tastenkombination ist häufig mehrmals zu betätigen, meisten bevor das Programm gestartet wird.

Weitere Tastenkombinationen:

`/usr/share/doc/packages/dosbox/README`

Ergänzung für die dosbox.conf:

Damit das CD-ROM Laufwerk gleich beim Start der Dosbox gemountet wird, ist am Schluss der dosbox.conf folgende Zeile einzufügen.

[...]

[autoexec]

Lines in this section will be run at startup.

mount d /media/cdrecorder -t cdrom -usecd 0

help ... ruft eine Kurzhilfe auf

exit ... beendet Dosbox

Die dosbox-Hilfe findet man unter dem Pfad:

`/usr/share/doc/packages/dosbox/README`

* * * * *

Domain- und Namensauflösung

Wann brauchen Sie einen eigenen Nameserver?

Eigene Nameserver sollte man immer dann einrichten, wenn ein lokales Netz an das Internet angeschlossen ist. Lokale Nameserver haben folgende Aufgaben:

- Verwalten der Namen für das lokale Netz (Hosting genannt),
- weiterleiten der DNS-Anfragen an den DNS-Server des Providers (Caching).

In kleineren Netzen ist ein eigener Nameserver nicht immer notwendig. Hier kann man die vorhandenen Rechner einfach in die Hosts-Datei eines **jeden**

Rechners eintragen.

Das Format dieser Datei ist für Linux und Windows identisch. Bearbeiten können Sie die Datei entweder direkt mit einem Texteditor oder mit der entsprechenden Funktion im Verwaltungszentrum ihrer Linux-Distribution.

Die Datei **hosts**, muss bei Windows 9x im Windows-Verzeichnis (meist **c:\windows**), bei Windows NT/XP unter **winnt\system32\drivers\etc**, bei Linux im Verzeichnis **/etc** gespeichert werden (hosts - die Datei hat **keine** Endung; eine Beispieldatei sollte sich in jedem Windows-Verzeichnis unter dem Namen **HOSTS.SAM** befinden).

Datei: **/etc/hosts**

#Dateianfang: hosts

```
# hosts      This file describes a number of hostname-
#            to-address mappings for the TCP/IP
#            subsystem.
#            It is mostly used at boot time, when no
#            name servers are running.
#            On small systems, this file can be used
#            instead of a "named" name server.
```

Syntax:

IP-Address Full-Qualified-Hostname Short-Hostname

127.0.0.1 **localhost**

127.0.1.1 tux

special IPv6 addresses

::1 localhost ipv6-localhost ipv6-loopback

fe00::0 ipv6-localnet

ff00::0 ipv6-mcastprefix

ff02::1 ipv6-allnodes

ff02::2 ipv6-allrouters

ff02::3 ipv6-allhosts

192.168.1.2 **boss.lokales-netz.de** **boss**

Dateiende: hosts

Zumindest die Zeilen, die den lokalen Rechner beschreiben, hier die beiden hervorgehobenen Zeilen, müssen sich immer in der Hosts-Datei befinden. So kann der Server zumindest seine eigenen Adressen immer auflösen.

Einen großen Teil der Datei können Sie ignorieren, er ist für die Erweiterung des IP-Adressformats auf 6 Byte bedeutsam.

In größeren Netzen kommt man nicht umhin einen Nameserver zu installieren. Die Installation und Einrichtung wurde in dieser Kurzreferenz nicht mit aufgenommen.

siehe auch: Netzwerk manuell aufsetzen, hostname, ifconfig

dpkg

Mit dpkg rufen Sie den Paketmanager debian package auf, und die Option `-i` installiert dieses Paket. Dazu muss man sich natürlich in dem entsprechenden Verzeichnis befinden, in dem sich auch das .deb-Paket befindet.

sudo dpkg -i dateiname.deb ... installiert das angegebene Programmpaket

dpkg -l ... listet alle installierten Programmpakete auf

dpkg -l | grep apt ... listet alle Einträge mit den Namensbestandteil apt auf

dpkg -L apt-utils ... zeigt den Inhalt eines Pakets und somit alle installierten Dateien an

dpkg -s apt-utils ... Details zum Paketstatus anzeigen

Mit dpkg installierte Programme kann man mit apt-get auch wieder deinstallieren (`sudo apt-get remove <Paketname>`).

Mit **gdebi** gibt es ein Tool mit grafischer Oberfläche. Nach einem Rechtsklick auf das entsprechende Debian-Paket wählen Sie im Kontextmenü **Mit »GDebi Paketinstaller« öffnen** aus.

In dem sich öffnenden Fenster des Paket-Installers werden Informationen und die Abhängigkeiten des Pakets angezeigt. Nach einem Klick auf **Paket installieren** werden das Paket und gegebenenfalls die vorhandenen Abhängigkeiten installiert.

Hinweis: Falls Sie ein Paket installieren wollen, das sich ebenfalls in den offiziellen Repositorys befindet, dann werden Sie durch einen Hinweis benachrichtigt, dass die Installation aus den klassischen Paketquellen zu bevorzugen sei. Sie können diesen Hinweis getrost ignorieren, wenn Sie sich über die Herkunft des Pakets sicher sind oder wissen, was Sie tun.

siehe auch: man dpkg, apt-get, apt-cache

Drucker

siehe auch: lpr, TurboPrint

E

echo <Zeichenkette>

Der Befehl gibt die angegebene Zeichenkette auf dem Bildschirm aus - die Zeichenkette kann auch innerhalb von Anführungszeichen (") stehen. Er ist äußerst nützlich beim Erstellen von Shellskripts oder bei der Überprüfung von Variableninhalte z.B. echo \$PATH.

Beim Erstellen von Skripts kann die Option -n behilflich sein, z.B. echo -n "hier steht Text". Sie bewirkt, dass der Cursor in derselben Zeile wie der angezeigte Text bleibt - standardmäßig wechselt er zur nächsten Zeile.

EFI

Für die Installation von Linux auf einem Rechner mit einem EFI-BIOS oder auch UEFI (Unified Extensible Firmware Interface) können keine allgemein gültigen Hinweise gegeben werden. Die aktuellen Linux-Betriebssysteme sollten alle das EFI-BIOS unterstützen.

Falls das gewählte Linux-Betriebssystem mit dem EFI-BIOS des Rechners Probleme hat, so kann der SECURE-Boot im BIOS deaktiviert werden. Die entsprechenden Auswahlfelder, werden i.d.R. erst nach der Vergabe eines Passwortes für den BIOS-Zugang anwählbar.

UEFI-Bootprozess

Aufgabe von UEFI ist es, eine Schnittstelle zwischen Hardware und Betriebssystem zu stellen und das System nach dem Start dem Betriebssystem zu übergeben. Zusätzlich umfasst die UEFI-Spezifikation eine API-Schnittstelle, um den Bootprozess und die Booteinträge von einem laufenden Betriebssystem aus konfigurieren zu können.

1. Nach dem Einschalten eines UEFI- Systems erwacht die Hardware zunächst wie bei einem herkömmlichen BIOS mit einem internen Selbsttest (Power-on Self-Test, kurz POST).
2. Ist alles in Ordnung, wird UEFI als Firmware geladen. Sie übernimmt die Initialisierung aller für den Start nötigen Geräte und lädt optionale Erweiterungen wie Secure-Boot.
3. Anders als Bios kann UEFI mit GPT- sowie MBR-Partitionen und dem FAT-Dateisystem umgehen. Es sucht alle Datenträger nach einer per GUID (Globally Unique Identifier) markierten EFI System Partition (ESP) ab. Auf dieser Partition sind die EFI-Programme hinterlegt, die Bootloader für UEFI-Werkzeuge und für installierte Betriebssysteme enthalten.
4. Der UEFI-Bootmanager führt den im NVRAM des Firmware-Chips als

Standard festgelegten Bootloader für ein Betriebssystem aus oder präsentiert ein vom Hardware-Hersteller gestaltetes Menü zur manuellen Auswahl von EFI-Bootloadern oder Partitionen. Natürlich können nicht alle Betriebssysteme mit UEFI umgehen und entsprechende Bootloader auf der ESP hinterlegen. Deshalb hat UEFI laut Spezifikation einen im Firmware-Menü aktivierbaren Kompatibilitätsmodus (CSM – Compatibility Support Module), der das System wie altes BIOS per MBR starten kann.

5. Wenn es das Betriebssystem zulässt, kann es der EFI-Bootloader direkt starten. Im Fall von Linux-Systemen lädt der EFI-Bootloader aber erst einen vorgeschalteten Bootloader wie Grub 2, der schließlich den Linux-Kernel und das Initramfs startet. Nach dem Start ist das Betriebssystem für den Rechner zuständig und kann mit eigenen Tools UEFI-Bootloader im Bootmanager der Firmware einrichten und ändern.

Beispiel: Secure-Boot bei einem Phoenix-BIOS ausschalten

1. Start des Rechners und die Taste [F2] mehrmals betätigen
2. Registerkarte: Security → Set Supervisor Password → Passwort vergeben → [F10] (Save and Exit)
3. Neustart des Rechners und die Taste [F2] mehrmals betätigen
4. BIOS-Passwort eingeben
5. Registerkarte: Advanced → Fast Boot: Disabled; CSM: Enabled (CSM: Compatibility Support Module, Service for legacy BIOS services)
6. Registerkarte: Exit → Exit Saving Changes

siehe auch: INTERNET

Emulatoren

siehe auch: Wine

exiv2

Exiv2 ist eine freie und quelloffene Bibliothek zur Bearbeitung der Metadaten von Digitalbildern. Mit exiv2 können Exif-Informationen von JPEG, PNG, TIFF und diversen RAW-Formaten ausgelesen und verändert werden.

Funktionen:

- Bildinformationen auslesen und ändern
- Vorschau bild extrahieren, löschen und ändern
- Ändern des Kommentarfelds von JPEG-Bildern
- Setzen und ändern des Exif-Zeitstempels

exiv2 [Optionen] [Aktionen] Datei(en)

Beispiele:

exiv2 picture.jpg ... Exif-Informationen anzeigen

exiv2 *.jpg ... eine Zusammenfassung der Exif-Informationen aller JPEG-Bilder eines Verzeichnisses anzeigen

exiv2 rename CIMG0438.JPG ... Umbenennung des Bildes in 20131017_124714.JPG; der Datumsstempel wird aus den Exif-Informationen des Bildes entnommen

exiv2 -r':basename:_%Y-%m-%d' rename CIMG0438.JPG ... Umbenennung des Bildes - unter Beibehaltung des Basisnamen – in CIMG0438_2013-10-17.JPG; der Datumsstempel wird aus den Exif-Informationen des Bildes entnommen

exiv2 -r':basename:_%Y-%m-%d_%H:%M:%S' rename *.JPG ... Umbenennung aller Bilder des aktuellen Verzeichnisses - unter Beibehaltung des Basisnamen; der Datums- und Zeitstempel wird aus den Exif-Informationen des Bildes entnommen

exiv2 ad -a 02 -k picture.jpg ... die Exif-Zeit für picture.jpg um 2 Stunden erhöhen; der Dateizeitstempel bleibt erhalten

exiv2 -et CIMG0445.JPG CIMG0446.JPG ... Thumbnail-Bilder (z.B. 160x120 Pixel) von den Bildern CIMG0445.JPG und CIMG0446.JPG erzeugen und mit der Namensweiterung -thumb speichern

exiv2 -it CIMG0445.JPG ... einen vorhandenenes Thumbnail-Bild mit der Namensweiterung -thumb.jpg (CIMG0445-thumb.jpg) im Bild CIMG0446.JPG einfügen

exiv2 -pp CIMG0445.JPG ... verfügbare Vorschaubilder im Bild CIMG0445.JPG auflisten

exiv2 -ep1 CIMG0445.JPG ... das im Bild CIMG0445.JPG enthaltene Preview-Bild extrahieren und mit der Namensweiterung -preview speichern

exiv2 -c 'Kommentar eintragen' CIMG0445.JPG ... einen JPEG-

Kommentar für das Bild setzen

exiv2 -pc CIMG0447.JPG ... den JPEG-Kommentar auslesen

exiv2 -da picture.jpg ... alle Exif-Informationen und sonstige unterstützten Meta-Daten im Bild löschen

exiv2 -da *.jpg ... alle Exif-Informationen und sonstige unterstützten Meta-Daten in allen JPEG-Bildern des aktuellen Verzeichnisses löschen

siehe auch: exiv2 -h, man exiv2

exiftool

ExifTool ist ein in Perl geschriebenes Kommandozeilen-Werkzeug (ExifTool ist Bestandteil des Paketes **libimage-exiftool-perl**) zum Lesen und Schreiben der Meta-Informationen von Bild-, aber auch von Audio- und Video-Dateien (unterstützten Dateiformate: man exiftool). Es ist vor allem ein Werkzeug, zur Änderung von Exif-, IPTC/IIM- und XMP-Metadaten in digitalen Bildern.

Entscheidender Vorteil ist, dass die Metadaten im Bild gespeichert werden - ohne das Bild selbst zu beeinflussen. Damit entspricht diese Vorgehensweise dem Festhalten von Informationen auf der Rückseite von Papierbildern. Während Exif-Daten direkt in der Kamera erzeugt und gespeichert werden, dienen IPTC/IIM und das neuere XMP für Bildinformationen zur Katalogisierung und Veröffentlichung. Eine Ausnahme stellt XMP insofern dar, dass Metadaten auch in einer zusätzlichen Datei pro Bild (sidecar) abgelegt werden können.

Unterstützt werden die Formate Exif (inkl. GPS-Daten), IPTC/IIM, XMP, JFIF, GeoTIFF, ICC Profile, Photoshop IRB, FlashPix, ACP und ID3-Meta-Informationen. Ebenso die (RAW-)Meta-Informationen, die verschiedene Digitalkameras (von Canon, Casio, Fuji, JVC/Victor, Kodak, Leaf, Minolta/Konica-Minolta, Nikon, Olympus/Epson, Panasonic/Leica, Pentax/Asahi, Ricoh, Sanyo und Sigma/Foveon) in den Bildern speichern. Des Weiteren werden auch viele andere Dateiformate wie PDF und Office-Dokumente wie .doc oder .odt unterstützt.

exiftool [Optionen] DATEINAME.EXT

Wird ExifTool ohne Optionen aufgerufen, so werden alle Informationen zur angegebenen Datei auf der Standardausgabe ausgegeben (also in der Regel auf dem Bildschirm).

Beispiele:

exiftool DATEiname.EXT ... alle Informationen ausgeben

exiftool -L DATEiname.EXT ... alle Informationen ausgeben;
Konvertierung aller am Bildschirm ausgegebenen Zeichen in Windows
Latin1 (cp1252); nützlich wenn exiftool teilweise Zeichensalat ausgibt

exiftool -p '\$Filename \$ImageSize' *.jpg ... Ausgabe von Dateiname und
Bildgröße (Breite x Höhe) für JPG-Dateien

exiftool -Make CIMG0379.JPG ... Make-Tag (Kamera-Hersteller)
anzeigen; hier: CASIO COMPUTER CO.,LTD.

**exiftool -p '\$Filename' -m -if '\$make eq "CASIO COMPUTER
CO.,LTD.'" *.JPG ...** Anzeige aller JPEG-Bilder im aktuellen
Verzeichnis, die mit einer Kamera von Casio aufgenommen wurden

**exiftool -Comment='This is a new comment' -overwrite_original
CIMG0308.JPG ...** neuen JPEG-Kommentar eintragen

exiftool -Comment CIMG0309.JPG ... JPEG-Kommentar auslesen

**exiftool -all= -comment='Meta-Informationen gelöscht' -
overwrite_original CIMG0308.JPG ...** entfernen aller Metadaten aus
einer JPG-Datei (**Hinweis:** vor dem Dateinamen ist ein Leerzeichen
notwendig) und einen neuen JPEG-Kommentar eintragen

exiftool -AllDates='JJJJ:MM:TT HH:MM:SS' DATEiname.EXT ...
Korrektur von Datum und Uhrzeit (z.B. bei falsch eingestellter Kamera)

**exiftool -AllDates+='HH:MM' -overwrite_original DATEiname.EXT
... Verschieben sämtlicher Bildzeiten um eine bestimmte Zeitdauer (hier:
Addition -AllDates+='02:06'; z.B. bei falsch eingestellter Kamera)**

**exiftool -b -ThumbnailImage DATEiname.EXT >
DATEiname_thumb.jpg ...** extrahieren eines Thumbnail-Bildes (z.B.
120 x 160 Pixel) aus dem Bild (sofern vorhanden)

**exiftool -b -PreviewImage DATEiname.EXT >
DATEiname_preview.jpg ...** Extrahieren eines Preview-Bildes (z.B. 120
x 160 Pixel) aus dem Bild (sofern vorhanden)

exiftool -Xresolution=150 -Yresolution=150 -ResolutionUnit=inches *.jpg ... verlustfreie Änderung des dpi-Wertes mehrerer JPG-Dateien

exiftool -q -r -t -f -S -n -csv -fileName -GPSPosition -Model -FocalLength -ExposureTime -FNumber -ISO -BrightnessValue -LensID "." > DATEiname.csv ... extrahieren von Metadaten aller Bilder aus einem Verzeichnis in eine CSV-Datei

exiftool -P -'Filename<DateTimeOriginal' -d %Y-%m-%d_%H:%M:%Ss_Handy.jpg tmp/CIMG0307.JPG ... Umbenennung des Bildes in 2011-06-04_21h:28m:05s_Handy.jpg; der Datums- und Zeitstempel wird aus den Exif-Informationen des Bildes entnommen

exiftool -P -'Filename<\${FileName}_\${DateTimeOriginal}' -d %Y-%m-%d_%H:%M:%S.%e tmp/*.JPG ... Umbenennung aller JPEG-Bilder im Verzeichnis tmp - unter Beibehaltung des Basisnamen; der Datums- und Zeitstempel wird aus den Exif-Informationen des Bildes entnommen; Beispiel: CIMG0307.JPG_2013-10-14_12h:59m:26s.JPG

wget -qO - http://a.domain.com/bigfile.jpg | exiftool -fast - ... Meta-Informationen eines Bildes aus dem Internet extrahieren

exiftool -all= DATEiname.JPG ... entfernen aller Metadaten aus einer JPG-Datei (**Hinweis:** vor dem Dateinamen ist ein Leerzeichen notwendig)

exiftool -all= *.JPG ... entfernen aller Metadaten aus allen JPG-Dateien im aktuellen Verzeichnis (**Hinweis:** vor dem Dateinamen ist ein Leerzeichen notwendig); die Originaldateien erhalten die Namensweiterung **_original** und die Metadaten in den umbenannten Originaldateien bleiben erhalten

exiftool -all= -overwrite_original *.JPG ... entfernen aller Metadaten aus allen JPG-Dateien im aktuellen Verzeichnis (**Hinweis:** vor dem Dateinamen und vor dem Parameter **-overwrite_original** ist ein Leerzeichen notwendig); die Originaldateien werden überschrieben und es werden keine Kopien von den Dateien erzeugt

siehe auch: man exiftool, exiv2

faillog

faillog zeigt die fehlgeschlagenen Anmeldeversuche an. Bei der Standardinstallation werden die fehlgeschlagenen Anmeldeversuche (z.B. wenn sich nicht autorisierte Personen Zugang verschaffen wollen) in einer Logdatei (Verzeichnis: /var/log/) protokolliert.

siehe auch: last

* * * * * * * * * *

fdisk

fdisk ist ein kleiner Partitionsmanager. fdisk kann nur als Systemadministrator root aufgerufen werden.

fdisk <Partition>

Es gibt zwar auch Partitionsmanager mit grafischer Oberfläche, wie z.B. qtparted oder partgui, diese sind aber nicht immer verfügbar. Die vorgenannten Partitionsmanager qtparted und partgui sind in der Linux-Live-Distribution Knoppix (www.knoppix.net bzw. <http://www.knopper.net/knoppix-mirrors/>) enthalten.

Achtung: Beim Anlegen von Dateisystemen mit **qtparted** nicht das **ext3**-Dateisystem benutzen. **qtparted** hat anscheinend ein Problem mit dem Anlegen von **ext3**-Partitionen (Info-Stand: 2005). ext2-Partitionen kann man nachträglich zu ext3-Partitionen konvertieren (**man tune2fs**).

Die Partitionsmanager **qtparted** und **partgui** sind zwar anwendungsfreundlicher als **fdisk**, aber sie sind nicht immer verfügbar - vor allem auf Rettungs-CD's. **fdisk** dagegen dürfte wohl auf jedem Linuxsystem verfügbar sein.

fdisk /dev/hda ... startet fdisk um die Partitionstabelle der ersten Festplatte hda zu bearbeiten

fdisk -l ... gibt die Partitionstabelle aller Festplatten aus, einschließlich Zylinderanzahl und Blockgröße

fdisk -s /dev/sda2 ... gibt die Größe der Partition sda2 in Blöcke aus

Das fdisk-Programm von Linux dient zur Erstellung von Linuxpartitionen auf Festplatten. Um DOS-Partitionen einzurichten, sollte das gleichnamige Programm von DOS verwendet werden.

Mit **fdisk /dev/hda** bearbeitet fdisk die Partitionstabelle der ersten

Festplatte. Um die zweite Festplatte, /dev/hdb, zu partitionieren, muss das Programm als **fdisk /dev/hdb** aufgerufen werden. Die erste SCSI-Festplatte wird entsprechend als /dev/sda angesprochen.

Wenn es korrekt aufgerufen wurde, meldet sich fdisk mit dem Prompt:

Command (m for help): _

fdisk erwartet einen einzelnen Buchstaben gefolgt von einem RETURN als Kommando.

Das **m** z.B. zeigt ein kleines Hilfemenü.

p ... Das print-Kommando zeigt die aktuelle Partitionstabelle an.

u ... Das unit-Kommando ändert die Einheiten, in denen die Partitionstabelle angezeigt und geändert wird. Das Kommando schaltet zwischen Sektoren und Zylinder als Einheit um. Weil MS-DOS die Partitionen auf komplette Zylinder erwartet, ist es sinnvoll, mit der voreingestellten Zylindereinheit zu arbeiten. Der Beginn einer Partition muss nicht mit dem Start des Datenbereichs übereinstimmen, wie das bei MS-DOS-Partitionen üblich ist, bei der die ersten Sektoren für Verwaltungsdaten freigehalten werden. Das Linux-fdisk setzt den Start des Datenbereichs normalerweise auf den Partitionsbeginn. Das auf Partitionsgrößen folgende +-Zeichen markieren Partitionen mit ungerader Sektorzahl. Weil Linux immer Blocks zu zwei Sektoren, also mindestens 1024 Bytes belegt, bleibt auf diesen Partitionen ein Sektor ungenutzt.

l ... Zeigt die bekannten Dateisystemtypen an. Für neue Linuxpartitionen ist die ID 83 vorgesehen. Diese Kennzahl ermöglicht es anderen Betriebssystemen, die Linuxpartition als fremd zu erkennen.

v ... Das verify-Kommando überprüft die Partitionstabelle auf Fehler und gibt die Anzahl der nicht belegten Sektoren aus.

q ... Das quit-Kommando beendet fdisk. Die Veränderungen an der Partitionstabelle werden nicht automatisch geschrieben. Eine Veränderung an der Partitionstabelle wird erst durch ein write-Kommando auf der Festplatte gesichert. Es ist möglich, fdisk zu jedem Zeitpunkt mit [Strg] + [C] zu beenden.

d ... Das delete-Kommando löscht eine Partition aus der Partitionstabelle. Die von dieser Partition belegten Sektoren werden freigegeben. Alle auf diesen Sektoren gespeicherten Daten gehen dabei normalerweise verloren. Das delete-Kommando fragt nach einer Partitionsnummer, die es löschen soll. Soll doch lieber keine Partition gelöscht werden, kann fdisk mit [Strg] + [C] beendet werden.

Wenn eine nicht existierende Partition angegeben wird, erscheint eine Fehlermeldung. Wenn die erweiterte Partition gelöscht wird, verschwinden automatisch auch alle logischen Partitionen auf der erweiterten Partition.

Wenn eine logische Partition gelöscht wird, werden alle darüber liegenden

logischen Partitionen sofort neu nummeriert, so dass alle logischen Partitionen immer fortlaufende Nummern haben.

n ...Das new-Kommando erlaubt das Anlegen einer neuen Partition.

Voraussetzung zum Anlegen einer neuen Partition ist natürlich, dass noch freie Sektoren existieren.

- Wenn ein Platz in der primären Partitionstabelle frei ist und Sektoren außerhalb einer evtl. existierenden Partition frei sind, kann eine Primärpartition angelegt werden.
- Wenn ein Platz in der primären Partitionstabelle frei ist und noch keine erweiterte Partition existiert, kann eine erweiterte Partition angelegt werden.
- Wenn innerhalb der erweiterten Partition Sektoren frei sind, kann eine logische Partition angelegt werden.

Wenn mehr als eine Möglichkeit zum Anlegen einer neuen Partition existiert, wird gefragt, ob eine primäre, erweiterte oder logische Partition angelegt werden soll. Wenn eine primäre oder erweiterte Partition angelegt werden soll, muss eine freie Partitionsnummer im Bereich 1 - 4 gewählt werden.

Nun muss der Beginn der neuen Partition angegeben werden. Dazu erscheint beispielsweise folgende Meldung:

First cylinder (403-1001): _

Die Zahlen bezeichnen den ersten und den letzten freien Zylinder. Es müssen nicht alle Zylinder in dem angegebenen Bereich frei sein, weil innerhalb des Bereichs noch eine komplette andere Partition liegen kann. Die Angabe eines bereits belegten Zylinders wird einfach abgelehnt und es wird erneut nach dem ersten Zylinder der neuen Partition gefragt.

Wenn ein gültiger Zylinder für den Beginn bestimmt ist, wird nach dem letzten Zylinder gefragt:

Last cylinder (403-1001): _

Hierbei sind alle Zylinder in dem angegebenen Bereich erlaubt. Wenn nicht alle freien Zylinder belegt worden sind, kann das Kommando erneut aufgerufen werden.

Bevor die veränderte Partitionstabelle gesichert wird, sollte auf jeden Fall das verify-Kommando aufgerufen werden.

Die veränderte Partitionstabelle wird in jedem Fall nur durch ein ausdrückliches write-Kommando auf die Festplatte geschrieben. In diesem Fall erscheint die Aufforderung, den Rechner neu zu starten:

The partition table has been altered.
Please reboot before doing anything else.

Es ist ohne Problem möglich, fdisk sofort wieder zu starten, um weitere Änderungen an der Partitionierung vorzunehmen.

Achtung: Es darf auf keinen Fall eines der Programme mkfs, mkswap, mount oder swapon aufgerufen werden, bevor der Rechner neu gestartet wurde!

Beispiel:

1. fdisk /dev/hda
2. [m] ... Hilfe
3. [p] ... Partitionstruktur anzeigen
4. [q] ... beenden ohne speichern (falls Änderungen gespeichert werden sollen - vorher Taste [w] eingeben und bestätigen)

Es ist möglich, fdisk zu jedem Zeitpunkt mit [Strg] + [C] zu beenden.

Um den verschiedenen Betriebssystemen die Möglichkeit zu geben, Partitionen anderer Formate zu erkennen, werden die Partitionen mit Kennzahlen (ID) versehen. Linux selbst kümmert sich nicht wirklich um die Partitions Kennung, schließlich kann Linux mit einer Vielzahl fremder Dateisysteme umgehen. Der Kernel prüft beim Mounten eines Dateisystems immer die für das jeweilige Dateisystem typische magische Zahl und erkennt so, ob es sich um ein gültiges Dateisystem eines bestimmten Typs handelt.

Voreinstellung für neue Linuxpartitionen ist 83 für das Linuxdateisystem. Diese Kennzahl ermöglicht es anderen Betriebssystemen, die Linuxpartition als fremd zu erkennen.

Nach der Partitionierung der Festplatte muss noch das Dateisystem eingerichtet werden, z.B. mit dem Systemprogramm **mke2fs**. Anschließend muss das Dateisystem in den Linux-Verzeichnisbaum eingehangen werden. Das Systemprogramm für diese Prozedur heißt **mount** bzw. **umount** um es aus dem Verzeichnisbaum wieder zu entfernen.

Ein typisches Beispiel für ein Kommando zum Einhängen einer Partition in

das Dateisystem:

```
mount -t ext2 -o grpquota /dev/hdb1 /usr/src
```

Durch das Kommando wird die erste Partition der zweiten Festplatte am ersten IDE-Controller /dev/hdb1, auf dem Verzeichnis /usr/src gemountet. Durch die Option -t wird der Typ des Dateisystems festgelegt, in diesem Fall ext2. Mit der Option -o wird dem Kernelsystem beim Mounten des Dateisystems ein Schlüsselwort übergeben, in diesem Fall grpquota, das den Treiber veranlasst, die Quotierung des Festplattenplatzes auf der Basis von der Gruppenzugehörigkeit abzurechnen.

* * * * *

Festplatte mounten

Sie können unter Linux jederzeit Festplatten bzw. freie Partitionen von Festplatten einbinden.

Wenn Sie zum Beispiel in /opt mehr Platz benötigen, können Sie dort eine zusätzliche Festplatte einhängen, mounten. Die genaue Vorgehensweise:

1. Festplatte einbauen und Linux starten.
2. Als Benutzer root anmelden oder als Hauptbenutzer die folgenden Befehle über sudo aufrufen (sudo fdisk).
3. Die Festplatte mit **fdisk** partitionieren, z.B. als **/dev/hdb1**
4. Formatieren der Partition mit **mke2fs /dev/hdb1**
5. Nun folgende Befehle eingeben:

```
cd /opt
mkdir /opt2
mount /dev/hdb1 /opt2
cp -axv . /opt2
```

Überprüfen Sie nun sorgfältig, ob alle Daten kopiert wurden. Danach können Sie das alte Verzeichnis verschieben und einen neuen leeren Mountpoint anlegen:

```
mv /opt /opt.old
mkdir /opt
```

Tragen Sie nun die neue Partition mit einem Editor zusätzlich in die /etc/fstab ein; das könnte so aussehen:

/dev/hdb1 /opt ext2 defaults 1 2

Jetzt sollten Sie den Rechner herunterfahren und neu booten.

6. Wenn der Rechner neu gebootet hat, vergewissern Sie sich mit dem Befehl **mount**, ob **/dev/hdb1/** auch wirklich unter **/opt** eingehängt wurde. Wenn alles wunschgemäß funktioniert, können Sie jetzt die alten Daten unter **/opt.old** entfernen:

```
cd /  
rm -fr opt.old
```

siehe auch: mount

* * * * *

file

Ermittelt den Typ von Dateien und bei Textdateien die Kodierung (z.B. text/plain; charset=iso-8859-1).

-z, --uncompress ... gibt Informationen über komprimierte Dateien aus - z.B. gzip-komprimierte Dateien

-i, --mime ... gibt dem Mime-Typ aus und bei Textdateien evt. auch die Kodierung

file -i iso.txt

iso.txt: text/plain; charset=iso-8859-1

Die Textdatei iso.txt verwendet die Kodierung ISO-8859-1.

file -i iso2.txt

iso2.txt: text/plain; charset=unknown

Von der Textdatei iso2.txt konnte die Kodierung nicht ermittelt werden. Diese Textdatei wurde wahrscheinlich unter einem anderen Betriebssystem erstellt. Tritt dieser Fall ein, so kann man die Textkodierung nur erraten.

Beispiel:

Die Textdatei kommt von einem System mit der wahrscheinlichen Systemsprache deutsch. In diesem Fall kann man davon ausgehen, dass die Textkodierung entweder UTF-8 oder iso-8859-1 ist.

siehe auch: iconv, Konvertierung von Textkodierungen, stat --help, recode, fromdos, todos

find

Das Programm find ist ein sehr nützlicher Befehl sowohl zum Auffinden von Dateien nach bestimmten Kriterien, als auch zum Ausführen bestimmter Operationen mit den gefundenen Dateien z.B. in Shellskripts.

find [PFAD] [OPTIONEN] [DATEI bzw. Suchkriterium]

find / -iname "*xyz*" ... Sucht die angegebene Datei ("*xyz*" sucht nach Dateinamen, in denen xyz vorkommt). Bei der Suche wird im Wurzelverzeichnis / begonnen. Es wird **nicht** zwischen Groß- und Kleinschreibung unterschieden.

find . ! -name '*.tex' ... nach allen Dateien suchen, die nicht auf .tex enden; bei der Suche wird im aktuellen Verzeichnis begonnen (. Punkt); Es wird zwischen Groß- und Kleinschreibung unterschieden

find /srv/ -iname "*hand*" ... hier wird ausschließlich im Verzeichnis /srv nach einer Datei mit den Namensbestandteil "*hand*" gesucht

find /opt -uid 1002 ... hier wird nach Dateien und Verzeichnissen gesucht die dem Benutzer mit der UID 1002 gehören

find ./ -maxdepth 1 -type f -iname '*.jpg' -exec mv {} Urlaub \; ... alle JPEG-Dateien des aktuellen Verzeichnisses in das Verzeichnis Urlaub verschieben; **Achtung:** Zwischen {} und Urlaub, sowie zwischen Urlaub und \; muss sich ein Leerzeichen befinden.

find ./ -type f -regex "^.*\.html\$" -exec grep -Hin "^.*SUCHWORT.*" {} \; ... HTML-Dateien nach einem Suchwort durchsuchen; Ausgabe: Dateiname:Zeilennummer:Zeile mit dem gefundenen Suchwort; **Achtung:** Zwischen {} und \ muss sich ein Leerzeichen befinden.

Optionen:

-type f ... Suche nach gewöhnlichen Dateien (f). Es kann auch nach Verzeichnissen (d), speziellen Textdateien (c), Blockdateien (b) und symbolischen Links (l) gesucht werden.

-user root ... Suche nach Dateien, die dem Administrator gehören. Der Besitzer kann auch ein anderer Benutzer sein, dessen Name oder Benutzer-ID (UID) angegeben wird.

-name cuxs ... Suche nach der Datei mit dem Namen cuxs.

-perm -4000 ... Suche nach Dateien mit dem gesetzten SUID-Bit. Der Rest der Bitmaske wird nicht berücksichtigt (-Zeichen vor 4000). Die Zugriffsrechte können auch in der längeren Form wie z.B. rwxr-xr-x angegeben werden.

-size +2048k ... sucht nach Dateien, die mindestens 2 Megabyte groß sind

Hinweis: Merkmale für Suchkriterien können auch mit dem Additions- oder

dem Subtraktionszeichen spezifiziert werden.

-mtime **[+][-]n** ... Suche nach Dateien, die vor mehr als (+n), weniger (-n) oder genau n Tagen modifiziert wurden.

-ctime **[+][-]n** ... Suche nach Dateien, die vor mehr als (+n), weniger (-n) oder genau n Tagen angelegt bzw. modifiziert wurden. Die Änderungen beziehen sich auf den Inhalt und die Attribute der Datei.

-atime **[+][-]n** ... Suche nach Dateien, auf die zuletzt vor mehr als (+n), weniger (-n) oder genau n Tagen zugegriffen wurde.

Hinweis: find denkt im 24 Stunden-Rhythmus $\rightarrow +3 = 3 * 24$ Stunden.

-amin **[+][-]**, **-cmin** **[+][-]**, **mmin** **[+][-]** ... diese Optionen sind den obigen sehr ähnlich, außer dass statt Tagen Minutenperioden überprüft werden.

-exec Kommando **{** **;** ... Ausführung des angegebenen Befehls für jede gefundene Datei. Zum Beispiel zeigt `find / -name *.conf -exec ls -l {} \;` alle Dateien, deren Namen mit der Zeichenkette .conf enden, im Vollformat an.

Achtung: Zwischen **{** und **;** muss sich ein Leerzeichen befinden.

Beispiele:

find . -name "*.pdf" -type f -exec du -ch {} + | tail -n1 ... Größe in MByte (K .. Kilobyte, M .. Megabyte, G .. Gigabyte) aller PDF-Dateien im aktuellen Verzeichnis, sowie ihrer Unterverzeichnisse; für andere Dateitypen ist pdf durch die entsprechende Dateierendung (mp3, gz, zip ...) zu ersetzen; Hinweis: `tail -n 1` (1 .. Zahl: Eins)

find / -path /proc -prune -o -type f -perm 666 ... durchsucht das gesamte Dateisystem (ausgenommen: /proc) nach Dateien mit den oktalen Rechten 666

find / -path /proc -prune -o -path /opt -prune -o -type f -perm 666 ... durchsucht das gesamte Dateisystem (ausgenommen: /proc und /opt) nach Dateien mit den oktalen Rechten 666

find / -path /proc -prune -o -type f -perm 777 ... durchsucht das gesamte Dateisystem (ausgenommen: /proc) nach Dateien mit den oktalen Rechten 777

find / -path /proc -prune -o -type d -perm 777 ... durchsucht das gesamte Dateisystem (ausgenommen: /proc) nach Verzeichnissen mit den oktalen Rechten 777

find / -perm +6000 -exec cp {} /root/tmp \; ... Sucht auf dem gesamten System nach Dateien mit gesetzten Set-UID- und Set-GID-Bit und kopiert sie in das tmp-Verzeichnis von root. Dateien mit gesetzten Set-UID- und Set-GID-Bit stellen ein Sicherheitsrisiko dar, deshalb sollte root immer auf

dem laufenden sein und Veränderungen kritisch beobachten.

find ./directory -maxdepth 1 -type f -iregex "^.*(\.(jpg|\.jpeg|\.png))\$"
... findet im Verzeichnis directory alle Bilder vom Typ JPEG und PNG;
Unterverzeichnisse werden nicht berücksichtigt; Verzeichnistiefe 1 (-maxdepth 1)

find ./picture_directory -maxdepth 2 -type f -not -iregex "^.*(\.(jpg|\.jpeg|\.png))\$" -exec rm -f {} \; ... löscht im Verzeichnis picture_directory alle Dateien die **nicht** vom Typ JPEG und PNG sind; ein Unterverzeichnis wird berücksichtigt; Verzeichnistiefe 2 (-maxdepth 2)

find ../ -maxdepth 1 -type f -iregex "^.*(\.(htm|\.html))\$" -not -iregex "^.*(index\.htm|gallery\.htm)\$" ... findet im übergeordneten Verzeichnis alle HTML-Dateien, mit Ausnahme der Dateien index.html und gallery.htm; Unterverzeichnisse werden nicht berücksichtigt; Verzeichnistiefe 1 (-maxdepth 1)

Hinweis: Bei ungenügenden Zugriffsrechten werden Fehlermeldungen ausgegeben (»... Keine Berechtigung«) - evtl. als root anmelden oder dem Terminalbefehl ein sudo voranstellen.

siehe auch: man find, mv, Skript-Listings: Suche in Dateien

finger

finger zeigt alle eingeloggten Benutzer auf dem System an - lokale wie entfernte (remote) Benutzer.

FLAC

siehe auch: Sound

ftp

FTP arbeitet mit je einem Verbindungskanal zum Steuern der Übertragung (Port: 21) und für die Übertragung (Port: 20) selbst:

- Auf dem Kommandokanal wartet der FTP-Server auf Befehle.
- Die eigentlichen Daten versendet oder empfängt der FTP-Server dann über einen gesonderten Datenkanal.

Als Kommandos erwartet der Server Befehle, die den üblichen Unix- oder DOS-Kommandos entsprechen. Darunter sind Befehle zum Bewegen im Verzeichnisbaum und für die Datenübertragung.

Die wichtigsten FTP-Kommandos:

Befehl

ftp ftp.server.de

Erläuterung

Verbindung mit einem FTP-Server herstellen z.B. ftp.server.de bzw. **ftp ftp://benutzername:passwort@ftp.server.de** (Hinweis: Im Benutzernamen, Passwort und im FTP-Server sollte **kein** at-Zeichen - @ - vorkommen)

ftp

Umschalten in den FTP-Modus (ftp>)

ls, dir

Anzeige des Inhaltsverzeichnis

cd <Zielverzeichnis>

Verzeichniswechsel auf dem Server

lcd <Zielverzeichnis>

Verzeichniswechsel auf dem Client

ascii

ASCII-Übertragungsmodus einschalten

binary

binären Übertragungsmodus einschalten

type

gibt den aktuellen Typ aus - ascii oder binär

rename <alter Dateiname>

<neuer Dateiname>

Dateien umbenennen

delete <Dateiname>

Datei löschen (remote-file)

get <Datei>

Angegebene Datei vom Server laden.

mget <Datei(en)>

Mehrere Dateien vom Server holen, Wildcards * und ? sind erlaubt.

put <Datei>

Datei zum Server übertragen.

mput <Datei(en)>

Mehrere Dateien zum Server übertragen, Wildcards * und ? sind erlaubt.

help <Befehl>

Hilfe im FTP-Modus anzeigen

help

Befehlsübersicht

quit, bye

Programm beenden.

free

Free zeigt die Summe des gesamten und des genutzten Arbeitsspeichers (RAM) bzw. des Swapspeichers an.

free -b ... Anzeige in Bytes

free -k ... Anzeige in Kilobytes

free -m ... Anzeige in Megabytes

fromdos

Konvertiert Textdateien von DOS ins UNIX-Format. fromdos ist Bestandteil des Paketes tofromdos (fromdos, todos).

Die verschiedenen Betriebssysteme (Linux, Unix, Windows) benutzen für das Zeilenende jeweils ein anderes nicht sichtbares Steuerzeichen (Linux \n; MAC \r; Windows \r\n); mit den Terminalprogrammen fromdos (Windows/

DOS → Linux) und todos (Linux → Windows/DOS) können die Zeilenumbrüche von Textdateien konvertiert werden.

fromdos a.txt b.txt ... a.txt und b.txt werden ins UNIX-Format konvertiert; Originaldatei wird dabei überschrieben

fromdos -b a.txt ... von der Originaldatei a.txt wird ein Backup (a.bak) erstellt

siehe auch: man fromdos, todos, man tofromdos, recode, iconv, Konvertierung von Textkodierungen

fuser

Mit fuser kann man herauszufinden, welche Prozesse oder Benutzer zur Zeit auf bestimmte Dateien zugreifen.

Das Unmounten eines Dateisystems ist nur möglich, wenn kein Prozess auf das entsprechende Gerät (Geräte-Datei, Device) zugreift. Schlägt umount fehl, kann man mit fuser die Prozesse ermitteln, die das Gerät derzeit nutzen – z.B. **fuser -v /media/cdrom**. Mit der Option -k beendet fuser alle Prozesse, die den Unmount verhindern – z.B. **fuser -k /media/cdrom**.

fuser [OPTIONEN] DATEI

Optionen:

-a ... Es werden auch Dateien ausgegeben, auf die gerade kein Prozess zugreift.

-u ... Gibt in Klammern den zur PID gehörenden Benutzernamen aus.

-v ... verbose-Modus; fuser gibt mehr Informationen aus

-k ... fuser beendet den Prozess der auf die angegebene und geöffnete Datei zugreift

Beispiele:

fuser -v /media/<benutzername>/* bzw. **fuser -v /media/*** ... bei angeschlossenen bzw. eingelegten Datenträgern (USB-Stick, DVD) wird von fuser der Benutzer, die Prozess-ID (PID) und weitere Angaben im Terminal ausgegeben

fuser -v 32797/tcp ... zeigt welches Programm den Port 32797 über das Protokoll TCP zur Zeit benutzt

fuser datei.pdf ... Angabe der Prozess-ID die auf die angegebene und

geöffnete Datei zugreift

fuser -v datei.pdf ... verbose-Modus; neben der Prozess-ID werden noch weitere Angaben im Terminal ausgegeben

fuser -k datei.pdf ... fuser beendet den Prozess (z.B. PDF-Betrachter) der auf die angegebene und geöffnete Datei zugreift

In der Ausgabe folgt i.d.R. nach der PID ein Buchstabe:

c.... Die Datei wird vom Prozess als Verzeichnis behandelt.

e.... Die Datei ist ausführbar und wird vom Prozess ausgeführt.

f.... Die Datei wurde vom Prozess geöffnet.

m.... Die Datei ist eine Bibliothek, die gemeinsam von den Prozessen genutzt wird.

r.... Dieser Buchstabe wird verwendet, wenn es sich um das Wurzelverzeichnis (root-Verzeichnis) handelt.

siehe auch: man fuser, netstat

GNU

GNU ... ist ein Wortspiel «GNU is not Unix»

GPL

GPL ... General Public License. GPL ist ein GNU-Projekt. Die Original General Public License - siehe <http://www.gnu.org/copyleft/gpl.html>.

Die Lizenzen für die meiste Software sollen verhindern, dass die Programme weitergegeben und geändert werden. Im Gegensatz dazu will GNU General Public License sicherstellen, dass freie Software von jedem benutzt und verändert werden kann. Die General Public License gilt für alle Programme deren Autoren ihr Werk der GPL unterstellt haben.

Wer als Autor sein neues Programm der GPL unterstellt, sollte am Anfang jeder Quelldatei eine Copyright-Zeile einfügen, sowie ein kurzer Hinweis darauf, wo die vollständige General Public License zu finden ist.

1. Zeile: Eine Zeile mit dem Programmnamen und einer kurzen Beschreibung.
2. Zeile: Copyright (C) 2012 Name des Autors
3. Zeile: Dieses Programm ist freie Software. Sie können es unter den Bedingungen der GNU General Public License, wie von der Free Software Foundation veröffentlicht, weitergeben und modifizieren. Eine Kopie der GNU General Public License haben sie mit diesem Programm erhalten.

gksu und kdesu

GNOME und KDE liefern mit gksu sowie kdesu jeweils ein Frontend für die Ausführung von Programmen mit Superuser-Rechten.

gksu gedit /etc/shadow ... die Datei shadow im Texteditor gedit mit root-Rechten öffnen
bzw.

Tastenkombination **[Alt] + [F2]** ... im darauf erscheinenden Fenster gksu gedit /etc/shadow eingeben.

gpg

siehe auch: Verschlüsselung

gpasswd

Mit gpasswd werden die beiden Dateien /etc/group und /etc/gshadow verwaltet.

`gpasswd [Optionen] <GRUPPE>`

Optionen:

-a, --add NUTZER ... NUTZER zu einer GRUPPE hinzufügen
-d, --delete NUTZER ... NUTZER aus einer GRUPPE entfernen
-r, --remove-password ... Passwort der GRUPPE entfernen
-M, --members NUTZER,[...] ... Liste mit den Mitglieder einer GRUPPE festlegen
-A, --administrators ADMIN,[...] ... Liste mit den Administratoren einer GRUPPE festlegen

Hinweis: Außer für -A und -M können die Optionen nicht kombiniert werden.

gpasswd buchhaltung ... der Gruppenadministrator für die Gruppe buchhaltung kann hiermit ein Gruppenpasswort setzen bzw. ändern

siehe auch: man gpasswd, man newgrp, usermod

groupadd

Mit groupadd kann dem System eine neue Benutzergruppe hinzugefügt werden.

`groupadd <Gruppenname>`

Um einzelne Benutzer zu einer Gruppe hinzuzufügen, ohne deren primäre Mitgliedschaft zu ändern ist das Programm usermod zu verwenden.

Gruppenverwaltung in Stichworten:

- Der Benutzer bezieht einen Teil seiner Rechte aus der Zugehörigkeit zu einer Gruppe
- Ein Benutzer kann zu mehreren Gruppen gleichzeitig angehören
- Die effektiven Rechte ergeben sich aus der Summe aller Rechte die man bezogen hat
- Die Reihenfolge der Gruppen spielt dabei keine Rolle, aber es gibt die primäre Gruppe und die zusätzlichen Gruppen. In Bezug auf die Rechte sind jedoch alle Gruppen gleich. die primären Gruppen beziehen sich nur auf das Eigentum
- GID = 0, Gruppe root, wenn andere Benutzer zur Gruppe root hinzugefügt werden erhalten Sie Rechte der Gruppe root, sind

jedoch keine Administratoren. D.h. die Rechte der Gruppe root und des Benutzers root unterscheiden sich.

- die Datei /etc/group enthält die Gruppen-Konten und spielt die gleiche Rolle für die Gruppen wie /etc/passwd für die Benutzer.
- Man kann auch für Gruppen Passwörter vergeben und auf manchen Systemen gibt es auch ein Shadowsystem (gshadow, siehe auch: man gshadow) für die Gruppenpasswörter. Dies spielt in der Praxis aber nur eine untergeordnete Rolle.

Beispieleintrag in der Datei /etc/group:

dialout:x:20:ben,karl,sabine

1. Feld: Gruppenname
2. Feld: verschlüsseltes Passwort; x ... kein Passwort für die Gruppe erforderlich
3. Feld: Gruppen-ID, GID
4. Feld: Liste mit den Benutzernamen

Beispiele:

sudo groupadd -g 500 finanz ... die Gruppe finanz mit der GID 500 wird angelegt

sudo groupadd buchhaltung ... die Gruppe buchhaltung wird angelegt; die Gruppen-ID (GID) wird vom System festgelegt

-g ... Hiermit wird die Gruppen-ID festgelegt werden. Der Wert muss positiv und einmalig sein. Standardmäßig wird automatisch die nächst höhere freie ID verwendet.

-p, --password PASSWORT ... verwende ein **verschlüsseltes** Passwort für die neue GRUPPE (siehe auch: mkpasswd)

siehe auch: groupadd -h, useradd, usermod, userdel, groupdel, groupmod

groupdel

groupdel entfernt eine Gruppe.

groupdel <Gruppenname>

groupdel buchhaltung ... löscht die Gruppe »**buchhaltung**«, die Gruppe sollte vorhanden sein (siehe auch: groupdel -help); Gruppen die durch die Standardinstallation von Linux eingerichtet werden, sollten nicht gelöscht

werden - sie werden vom System benötigt;

siehe auch: man groupdel, groupadd, groupmod, useradd, userdel

groupmod

groupmod dient zum Ändern von Gruppeninformationen.

groupmod <Optionen> <Gruppe>

groupmod -n finanz buchhaltung ... die Gruppe buchhaltung wird in finanz umbenannt

-g, --gid GID ... neue GID für die GRUPPE erzwingen

-n, --new-name NEUE_GRUPPE ... neuen Namen NEUE_GRUPPE für die GRUPPE erzwingen

-p, --password PASSWORT ... verwende **verschlüsseltes** Passwort als neues Passwort (siehe auch: mkpasswd)

siehe auch: groupmod -h, groupadd, usermod, useradd

groups

groups zeigt an welchen Gruppen ein Benutzer angehört. Ein normaler Benutzer lässt sich mit der Eingabe von groups seine Gruppenzugehörigkeiten anzeigen. root kann sich auch die Gruppenzugehörigkeit anderer Benutzer anzeigen lassen, indem er groups zusammen mit dem Benutzernamen aufruft.

groups ... groups zeigt die Gruppenzugehörigkeit des aktuellen Benutzers an

sudo groups ben ... mit root-Rechten lässt sich die Gruppenzugehörigkeit auch von anderen Benutzern anzeigen (hier: Benutzer ben)

siehe auch: id, groupadd

gzip

gzip komprimiert und dekomprimiert Dateien.

gzip -d <Dateiname> ... Dekomprimiert gepackte gzip-Dateien, sodass sie wieder normal bearbeitet werden können (entspricht dem Aufruf von **gunzip <Dateiname>**).

gzip -c <Dateiname> ... Gibt die komprimierte Datei auf dem Bildschirm aus (Standard-Ausgabe). Nützlich bei einer Weiterleitung des Datenstroms an ein anderes Kommando (Pipe: |).

gzip -1 <Dateiname> ... Komprimiert eine Datei sehr schnell.

gzip <Dateiname> ... Komprimiert eine Datei mit dem Kompressionsgrad 6 (Default-Wert).

gzip -9 <Dateiname> ... Komprimiert eine Datei am besten - max. Komprimierung.

Die Namen der reduzierten Dateien enden dann auf .gz und müssen vor erneuter Benutzung wieder entpackt, dekomprimiert werden. Um mehrere Dateien oder ganze Verzeichnisse zu komprimieren, muss zusätzlich der Befehl **tar** (z.B. **tar -cvzf test.tar.gz test**) verwendet werden.

siehe auch: tar, unzip, unrar

grabc

grabc ist ein einfaches Programm zur Farbwertbestimmung (hexadezimal und dezimal) durch Anklicken eines Pixels. Wenn das Programm läuft (**grabc** in ein Terminal eingeben), wird der Mauszeiger zu einem Fadenkreuz. Die Farbe des angeklickten Pixels werden als Hexadezimalwert und als Dezimalwerte (RGB) am Bildschirm ausgegeben. Ein alternatives Programm mit grafischer Oberfläche ist Gcolor2.

siehe auch: grabc -h

grep

grep sucht Ausdrücke in Textdateien.

grep [OPTIONEN] <Suchwort> <MeineDateien>

-i, --ignore-case ... Groß- und Kleinschreibung wird ignoriert

-n, --line-number ... zeigt zusätzlich die Nummern der Zeilen, in den grep fündig geworden ist

-L, --files-without-match ... zeigt nur die Namen der Dateien in denen keine Übereinstimmung gefunden wurde

-l ... zeigt nur die Namen der Dateien in denen eine Übereinstimmung gefunden wurde

-w, --word-regexp ... das Suchwort muss als ganzes Wort enthalten sein

-H, --with-filename ... Dateinamen bei jeder Übereinstimmung anzeigen

-h, --no-filename ... Dateinamen nicht anzeigen

-a, --text ... berücksichtigt nur Textdateien

-E, --extended-regexp ... erweiterter Interpreter für reguläre Ausdrücke wird aktiviert

-v, --invert-match ... Invertierung des Suchmusters; findet die Zeilen in denen das Suchmuster nicht vorkommt

-B <ZAHL> ... Zeilen vor dem gesuchten Text anzeigen
-A <ZAHL> ... Zeilen nach dem gesuchten Text anzeigen
-C <ZAHL> ... Zeilen vor und nach dem gesuchten Text anzeigen

Beispiele:

grep -i test file.txt ... Es wird in der Datei file.txt nach dem Suchwort test gesucht. Bei der Suche wird nicht zwischen Groß- und Kleinschreibung unterscheiden (-i).

grep -liw tar * ... Es werden nur die Dateinamen ausgegeben in denen tar als ganzes Wort vorkommt. Groß- und Kleinschreibung, wird bei der Suche ignoriert. Durch Angabe des Wildcards * wird in allen Dateien des aktuellen Verzeichnisses gesucht.

grep -n " file.txt | cut -s -f 1-20 ... Zeigt alle Zeilen nummeriert an, in denen sich ein Tabulator befindet.

grep 'Tee|Kaffee' *.txt ... Es werden im aktuellen Verzeichnis in allen Textdateien nach den Ausdrücken **Tee** und **Kaffee** gesucht.

grep -E 'Tee.*Kaffee|Kaffee.*Tee' *.txt ... Es werden im aktuellen Verzeichnis in allen Textdateien nach den Zeilen gesucht, in denen die Ausdrücken **Tee** und **Kaffee** bzw. **Kaffee** und **Tee** gleichzeitig enthalten sind.

grep 'Tee' *.txt | grep -v 'Kaffee' ... Es werden im aktuellen Verzeichnis in allen Textdateien nach den Zeilen gesucht, in denen der Ausdruck **Tee** enthalten ist, jedoch nicht auch noch der Ausdruck **Kaffee** vorkommt.

grep -ah -B 5 -A 200 "hier den Suchtext eingeben" /dev/hda2 > ./suchtext.txt & ... Dieser Befehl sucht den Suchtext in der gesamten Partition hda2 und speichert das Ergebnis in der Datei suchtext.txt. Es werden 5 Zeilen vor dem gefundenen Text und 200 Zeilen nach dem gefundenen Text ausgegeben. Da dieser Befehl einige Zeit in Anspruch nimmt, läuft dieser Befehl als Hintergrundprozess (&).

for i in \$(find ./ -iname "*.doc"); do antiword \$i | grep --label=\$i -nh SUCHWORT; done

Der gesamte Ausdruck ist in einer einzigen Zeile einzugeben. Der Ausdruck sucht im aktuellen Verzeichnis mit dem Programm antiword nach dem SUCHWORT - mehrere Suchwörter in doppelte Hochkommas (") einschließen. Es wird die Zeilennummer und die Zeile als Ergebnis ausgegeben. Falls statt der Option **-nh** die Option **-nl** gewählt wird, wird als

Ergebnis nur der Dateiname ausgegeben.

Vorher mittels `cd` in das Verzeichnis mit den Dateien im .doc-Format wechseln. **Beachte:** antiword kann in Version: 0.37 (Oktober 2009) nur Dateien im .doc-Format bis zur Word-Version 2000 öffnen und lesen.

siehe auch: antiword, Dateien: versehentlich gelöschte Dateien mittels des Befehls `grep` wiederherstellen, Anhang: Einführung in die Shellprogrammierung

Grafikkarte prüfen: 3D-Beschleunigung Yes oder No

glxinfo

Mit dem Befehlsaufruf **glxinfo** wird in der dritten Zeile - der umfangreichen Bildschirmausgabe - angezeigt, ob die 3D-Beschleunigung (direct rendering: Yes) funktioniert - No oder Yes.

Ist die 3D-Beschleunigung nicht aktiviert, so befragen sie die Hilfeseiten ihrer Distribution oder das Internet. Häufig kann man die 3D-Grafikkartentreiber für Linux, einschließlich der Installationssoftware, auch bei der Herstellerfirma (ATI, Nvidia) bekommen. Vor der Installation der 3D-Grafikkartentreiber für Linux ist es immer ratsam, vorher einige Erfahrungsberichte aus dem Internet oder Zeitschriften aufmerksam zu lesen.

Hinweis: Der beste Test für die Funktionstüchtigkeit der 3D-Treiber, ist das 3D-Spiel »TuxRacer«.

Um die 3D-Konfiguration zu überprüfen, kann man auch das Diagnosetool **3Ddiag** verwenden. 3Ddiag ist ein Kommandozeilentool, dass in einem Terminal aufgerufen werden muss. Befolgen Sie die Anweisungen von 3Ddiag, wenn es zu »**failed**«-Meldungen kommt. Im Erfolgsfall werden ausschließlich »**done**«-Meldungen auf dem Bildschirm ausgegeben.

siehe auch: 3Ddiag -h

Grub 2

Der Grand Unified Bootloader 2 – kurz GRUB 2 – ist die zweite Version von GNU GRUB und damit der Nachfolger von GRUB (GRUB Legacy).

Hinweis: An dieser Stelle wird nur sehr kurz auf die Funktion, Konfiguration und Bearbeitung von Grub 2 eingegangen (siehe auch: INTERNET).

Ein Bootloader ist notwendig, um Betriebssysteme auf einem Computer

überhaupt starten zu können. GRUB 2 ist eine vollständige Neuentwicklung, so dass er sich von GRUB Legacy – insbesondere was die Konfiguration anbelangt – in vielen Punkten unterscheidet.

Einige neue Merkmale von GRUB 2 sind:

- Unterstützung von EFI
- Unterstützung von GUID Partition-Table (GPT), auch ohne EFI
- Unterstützung von EFI-BIOS mit secure-boot
- direkte Unterstützung von UUID und Festplatten-Bezeichnung (Labels)
- Modulares Laden der Komponenten zur Laufzeit
- Grafische Benutzeroberfläche, mit der Möglichkeit das Aussehen von GRUB anzupassen
- Unterstützt unterschiedliche Systemplattformen
- Unterstützung von Skripten
- Benutzerdefinierte Boot-Einträge
- Rettungsmodus zum Beheben von Bootproblemen

Systemstart

Folgende Schritte werden beim Starten eines Rechners durchlaufen:

- Das BIOS sucht je nach Einstellung
 - im MBR auf dem ersten Datenträger nach einem Bootmanager bzw. Bootloader.
 - auf der EFI-Partition nach der Startdatei grubx64.efi
- Der Bootloader GRUB 2 lädt sich nach und nach die Dateien bzw. Images
 - boot.img
 - core.img
 - /boot/grub/grub.cfg
 - erforderliche mod-Dateien (Treiber) für jeweils spezielle Aufgaben.

Damit kann dann die Anzeige auf dem Monitor angeboten und innerhalb eines Zeitlimits (timeout) die Auswahl von der Tastatur ausgewertet, umgesetzt und ein Betriebssystem gestartet werden.

MBR mit Master-Partitionstabelle (MPT)

Bei der Installation von GRUB 2 wird u.a. ein freier Bereich direkt hinter

dem Master-Boot-Record (MBR) benötigt, um die obigen Informationen abzulegen und ein ordnungsgemäßes Booten zu ermöglichen. Es muss zusätzlich darauf geachtet werden, dass dieser freie Bereich ausreichend groß ist - die dazugehörigen, aktuellen Werkzeuge für die Partitionierung (u.a.: GParted) berücksichtigen diese Anforderung. Insbesondere bei einer Installation auf eine SSD ist es wegen dem Alignment wichtig, dass hierbei die Blockgrenzen beachtet werden. Die erste Partition sollte dann erst ab Sektor 2048 beginnen - typisches Beispiel im Terminal mit:

sudo fdisk -l

ergibt dann:

Gerät	boot.	Anfang	Ende	Blöcke	Id
System					
/dev/sdc1		2048	75499519	37748736	7
HPFS/NTFS/exFAT					
...					

MBR mit GUID-Partitionstabelle (GPT)

Wird eine Festplatte neu mit einer GPT versehen und werden dazu die aktuellen Werkzeuge für eine Formatierung, wie z.B. GParted verwendet, so unterscheidet sich die Vorbereitung nicht von der unter Master-Partitionstabelle beschriebenen Maßnahmen.

Bei Verwendung einer vorhandenen GPT muss man aber zusätzlich, je nach Installationsart (BIOS oder EFI), einen gesonderten Bereich (hier Partition) anlegen, um die GRUB 2-Informationen zu hinterlegen.

- ohne EFI

Um GRUB 2 mit einer GUID-Partitionstabelle ohne EFI nutzen zu können, muss eine gesonderte Bootloader-Partition eingerichtet sein. Dieses kann in der GUI des Installers einer Desktop-CD/DVD bei der Auswahl und Einstellung der Partitionen vorgenommen werden.

- Partitionsnummer: erste freie Stelle auf dem Datenträger
- Kennung: ef02
- Name: bios_grub (bei GParted) - BIOS Boot-Partition (bei gdisk)
- Dateisystem: keins – RAW-Zustand
- GUID: 21686148-6449-6E6F-744E-656564454649
- Größe: 1024 KiB (1 MiB)
- kein Mountpunkt

- mit EFI

Um GRUB 2 mit EFI nutzen zu können, wird eine EFI-System-Partition (ESP) benötigt:

- Partitionsnummer: 1 (Empfehlung)
- Kennung: ef00
- Name: EFI System
- Dateisystem: FAT (Standard 32 bit)
- GUID: C12A7328-F81F-11D2-BA4B-00A0C93EC93B
- Größe: 100-200 MiB
- Mountpunkt: /boot/efi

Konfiguration

- Änderungen an den GRUB 2-Konfigurationsdateien sollten stets nur mit Bedacht durchgeführt werden, weil im schlimmsten Fall das System danach nicht mehr richtig startet.
- Vor Experimenten mit der Datei /etc/default/grub und vor allem mit den Skripten in /etc/grub.d sollte ein Backup angelegt werden!

Grundsätzlich ist GRUB 2 von seinen Machern so konzipiert, dass es, sobald das Kommando `update-grub` bzw. `grub-mkconfig` ausgeführt wird, alle auf einem Computer installierten Betriebssysteme findet und automatisch in das Auswahlmenü aufnimmt. Eine manuelle Konfiguration von GRUB 2 ist für die grundlegende Funktionalität daher nicht notwendig. Möchte man GRUB 2 trotzdem manuell anpassen - und sei es auch nur aus optischen Gesichtspunkten - so geschieht das über die Datei `/etc/default/grub` und über die Skripte im Verzeichnis `/etc/grub.d`. Über die Datei `/etc/default/grub` kann man einige grundlegende Einstellungen vornehmen, die Skripte sind für aufwendigere Anpassungen aber auch für individuelle Menü-Einträge gedacht.

Immer an die aktuelle Ausgabe angepasste Informationen zur Konfiguration der Datei `/etc/default/grub` kann man sich im Terminal anzeigen lassen mit:

`info -f grub -n 'Simple configuration'`

Mit dem Grub Customizer existiert ein grafisches Konfigurationswerkzeug.

Datei: `/etc/default/grub`

Die Datei `/etc/default/grub` ist eine einfache Textdatei die eine Reihe von Standardvariablen enthält.

Bedeutung der Variablen:

GRUB_DEFAULT=<Zahl> ... Gibt an, welcher Eintrag im Menü standardmäßig hervorgehoben wird. Dieser Eintrag wird geladen, falls keine andere Auswahl getroffen wird. Die Zählung beginnt mit 0. Der dritte Eintrag im Menü würde also durch eine 2 hervorgehoben. Diese Vorgabe ist statisch.

GRUB_HIDDEN_TIMEOUT <Zahl> ... Diese Funktion sollte mit einer vorangestellten Raute (#) anstelle einem Wert = 0 (Null) unwirksam gemacht werden (Hinweis: Entfällt zukünftig zugunsten der Variablen `GRUB_TIMEOUT_STYLE`). Ein Zahlenwert > 0 gibt die Zeit in Sekunden an, bis der Bootvorgang ohne Anzeige des Auswahlmenü ausgeführt wird. Mit der Umschalt-Taste kann das Auswahlmenü innerhalb der eingestellten Zeitspanne sichtbar gemacht werden.

GRUB_HIDDEN_TIMEOUT_QUIET=true ... Wenn dieser Wert auf false gesetzt, so wird nur der unter `GRUB_HIDDEN_TIMEOUT` eingestellte Wert auf dem Monitor als Countdown angezeigt. Mit der Shift-Taste kann das Auswahlmenü innerhalb der Zeitspanne sichtbar gemacht werden (Hinweis: Entfällt zukünftig zugunsten der Variablen `GRUB_TIMEOUT_STYLE`).

GRUB_TIMEOUT=<Zahl> ... Ein Zahlenwert gibt die Zeit in Sekunden an, wie lange das Auswahlmenü angezeigt wird, bevor der Standard-Eintrag geladen wird. Bei 0 wird direkt der unter `GRUB_DEFAULT` eingestellte Eintrag geladen, ohne dass das Auswahlmenü angezeigt wird, bei -1 wird der Zähler abgeschaltet und man muss den zu ladenden Eintrag immer von Hand wählen. Wurde `GRUB_HIDDEN_TIMEOUT` aktiviert, ist diese Funktion solange wirkungslos, bis `GRUB_HIDDEN_TIMEOUT` mit der Umschalt-Taste deaktiviert wird.

GRUB_TIMEOUT_STYLE=menu ... Ohne Eintrag oder mit dem Eintrag `menu` wird das `GRUB_2`-Menü auch dann angezeigt, wenn kein zweites Betriebssystem (Windows / Linux) gefunden wurde. Das System wird mit der Auswahl (Pfeiltasten und Enter) oder nach Ablauf der mit `GRUB_TIMEOUT` eingestelltem Zeit gestartet. Mit dem Eintrag `hidden` wird kein `GRUB_2`-Menü angezeigt und die mit `GRUB_TIMEOUT` eingestellte Zeit gewartet, bis das System nach der voreingestellten Zeit startet.

Durch kurzzeitiges Drücken der Esc-Taste innerhalb dieses Zeitfensters kann das Grub-Menü angezeigt und eine Auswahl vorgenommen werden. Mit dem Eintrag `countdown` wird kein GRUB_2-Menü angezeigt und die mit `GRUB_TIMEOUT` eingestellte Zeit sichtbar heruntergezählt. Danach startet das System mit dem mit `GRUB_DEFAULT` voreingestellten Eintrag. Durch kurzzeitiges Drücken der Esc-Taste innerhalb dieses Zeitfensters kann das Herunterzählen abgebrochen und eine Auswahl vorgenommen werden.

Skripte in `/etc/grub.d`

Die Menü-Einträge zur Auswahl des zu startenden Betriebssystems und auch das genaue grafische Erscheinungsbild werden bei GRUB 2 über die Skripte im Verzeichnis `/etc/grub.d` konfiguriert.

Die Skriptnamen werden von einer Nummer (XX_) angeführt. Die Skripte werden aufsteigend dieser Nummer entsprechend abgearbeitet. Haben zwei Skripte die gleiche Nummer, so bestimmt deren weitere alphanumerische Reihenfolge die Abarbeitung.

Grub-Konfiguration updaten

Nach jeder Änderung an der Datei `/etc/default/grub` oder an Skripten im Verzeichnis `/etc/grub.d`, muss die Grubkonfiguration - die in der Datei `/boot/grub/grub.cfg` gespeichert ist - aktualisiert werden. Andernfalls werden Änderungen beim nächsten Systemstart nicht sichtbar.

Um eine Information darüber zu erhalten, wie die Datei `/boot/grub/grub.cfg` mit den momentanen Einstellungen aussieht, gibt man im Terminal ein:

`sudo grub-mkconfig`

und kann dann das Ergebnis überprüfen. Dieses wird noch nicht in die Grub-Konfiguration übernommen. Ist man mit dem Ergebnis zufrieden, so kann man mit

`sudo update-grub`

die Datei `/boot/grub/grub.cfg` neu schreiben lassen.

GRUB-2-Shell

Die GRUB-2-Shell ist die Laufzeit-Umgebung von GRUB 2 und kann abhängig vom System-Status verschiedene Modi starten. Diese sind der Auswahlmenü-Modus (= "Menu Mode"), die Kommandozeile (= "Command Line Interface" oder "CLI-Mode") und schließlich noch der Rettungsmodus (= "Rescue Mode"). Außerdem erlaubt es die GRUB-2-Shell aus dem Auswahlmenü-Modus heraus den Menü-Bearbeitungs-Modus

(= "Edit Mode") bei Bedarf aufzurufen.

Hinweis: In der gesamten GRUB-2-Shell und damit in allen Modi steht nur eine amerikanische Tastaturbelegung zur Verfügung.

Auswahlmenü-Modus

Normalerweise startet GRUB 2 im Auswahlmenü-Modus (= "Menu Mode"), wobei das Auswahlmenü standardmäßig vor dem Benutzer verborgen bleibt, sofern sich nur ein einziges Betriebssystem auf dem Rechner befindet oder GRUB 2 bei der Installation keine anderen Betriebssystem erkannt hat. GRUB 2 startet dann einfach das System ohne das Menü anzuzeigen.

Man kann die Anzeige des Auswahlmenüs aber auch in einem solchen Fall durch Drücken der Shift-Taste während des Startvorgangs des Rechners erzwingen. Dazu direkt nach dem BIOS die Shift-Taste drücken und gedrückt halten bis das Menü erscheint.

Steuertasten im Auswahlmenü-Modus

Folgende Tasten-Kombinationen können im Auswahlmenü verwendet werden:

Pfeil-Tasten: Ab und Auf ... Menü-Eintrag hervorheben

Enter- oder Eingabetaste ... hervorgehobenen Menü-Eintrag starten

Taste [E] ... Für den ausgewählten Menü-Eintrag in den Bearbeitungs-Modus wechseln.

Taste [C] ... in die Kommandozeile wechseln

[Strg] + [Alt] + [Entf] ... System-Neustart

Hinweis: Die Tasten [E], [C] und die Enter-Taste können durch das Einbringen eine Passwort-Schutzes gesperrt werden.

Starten im Auswahlmenü-Modus

Im Auswahlmenü-Modus einfach den gewünschten Eintrag durch Drücken der Pfeiltasten (Ab und Auf) auswählen und dann durch Drücken der Eingabetaste starten. Startet das System nicht und fällt wieder zurück in das Auswahlmenü, dann den Eintrag im Bearbeitungs-Modus überprüfen und temporär korrigieren.

Menü-Bearbeitungs-Modus

Der Menü-Bearbeitungs-Modus (= "Edit Mode"), welchen man aus dem Auswahlmenü durch Drücken der Taste [E] erreicht, erlaubt es einen bestehenden Menü-Eintrag zu bearbeiten und einmalig für den Systemstart

zu korrigieren. Da ein falscher Menü-Eintrag in der Regel zum Auswahl-Menü zurückführt, kann man mit Hilfe des Bearbeitungs-Modus im Zweifel solange an einem Menü-Eintrag herumprobieren, bis er das System startet.

Erfolgversprechender ist es natürlich, wenn man die Startumgebung zuvor in der Kommandozeile analysiert und den fehlerhaften Eintrag dann gezielt korrigiert.

Hat man das System durch Bearbeitung eines Menü-Eintrages erfolgreich gestartet, so muss die Menü-Konfiguration im laufenden System dauerhaft angepasst werden.

Der Bearbeitungs-Modus ist immer nur aus dem Auswahlmenü heraus aufrufbar und wird von GRUB 2 nicht automatisch geladen.

Steuertasten im Bearbeitungs-Modus

Von den Einschränkungen, die sich aus der Tastaturbelegung ergeben abgesehen, stehen im Bearbeitungs-Modus alle wesentlichen Tasten zur Verfügung. Folgende Tasten sind dabei besonders hervorzuheben:

Pfeil-Tasten: Ab und Auf ... Die Eingabemarke (Cursor) um eine Zeile nach unten oder oben bewegen.

Pfeil-Tasten: Rechts und Links ... Die Eingabemarke um ein Zeichen nach links oder rechts bewegen.

Taste [Tab] ... Zeigt durch ein und mehrmaliges Drücken die mögliche Fortsetzung einer Pfadangabe an. Das ist sehr nützlich um Pfadangaben korrekt angeben zu können.

[Strg] + [X] ... Das System unter Verwendung des bearbeiteten Menü-Eintrages starten.

[Strg] + [C] ... In die Kommandozeile wechseln.

[Esc] ... Bearbeitungs-Modus verlassen und zum Menü-Modus zurückkehren.

[Strg] + [Alt] + [Entf] ... System-Neustart

Starten mit Hilfe des Menü-Bearbeitungs-Modus

Hat man den Menü-Eintrag wie gewünscht angepasst, so kann man das System durch Drücken der Tastenkombination **[Strg] + [X]** starten. Ist der Eintrag noch fehlerhaft, kehrt GRUB 2 in den Bearbeitungs-Modus für den entsprechenden Menü-Eintrag zurück und setzt die Eingabemarke in die noch fehlerhafte Zeile. Dies geschieht teilweise auch erst nach Ausgabe einer Fehlermeldung und Drücken einer beliebigen Taste.

Kommandozeile

Die Kommandozeile (Command Line Interface oder CLI-Mode) ist das Kernstück der GRUB-2-Shell. Mit ihr kann das System Schritt für Schritt analysiert und gestartet werden. Auch die vorkonfigurierten Menü-Einträge werden von ihr nach Auswahl in der aufgeführten Reihenfolge Zeile für Zeile abgearbeitet. Sie ist in ihrer Funktion mit der Bash vergleichbar, bietet aber eben nur die für einen Systemstart wichtigen Befehle und Analysefunktionen.

Die Kommandozeile erreicht man aus dem Auswahlmenü durch Drücken der Taste [C] . Außerdem landet man bei einem Systemstart automatisch in der Kommandozeile, wenn GRUB 2 die Konfigurationsdatei grub.cfg nicht finden oder verarbeiten kann.

Steuertasten in der Kommandozeile

Von den Einschränkungen, die sich aus der Tastaturbelegung ergeben abgesehen, stehen in der Kommandozeile alle wesentlichen Tasten zur Verfügung. Folgende Tasten sind dabei besonders hervorzuheben:

Pfeil-Tasten: Ab und Auf ... In der Befehls-Historie bereits zur Laufzeit eingegebener Befehle blättern.

Pfeil-Tasten: Rechts und Links ... Die Eingabemarke um ein Zeichen nach links oder rechts bewegen.

Taste [Tab] ... Zeigt nach Drücken alle möglichen Fortsetzung des eingegebenen Befehls oder Befehlsteils an. Diese Funktion unterstützt je nach verwendeten Befehl bereits eine Basis-Analyse, verhindert aber vor allem syntaktische Fehler bei der Befehlseingabe.

help + [Enter- oder Eingabetaste] ... Zeigt die Liste verfügbarer GRUB 2-Befehle.

[Esc] ... Kommandozeile verlassen und zum Menü-Modus zurückkehren. Dies ist natürlich nur möglich, sofern die Kommandozeile anfangs manuell aus dem Auswahl-Menü aufgerufen wurde.

[Strg] + [Alt] + [Entf] ... System-Neustart

Rettungs-Modus

Der Rettungs-Modus (Rescue Mode) ist eine stark eingeschränkte Form der GRUB-2-Kommandozeile. Er bietet nur wenige ganz grundlegende Befehle und keinerlei Komfort-Funktionen, wie das Ergänzen von Eingaben mittels Tabulatortaste oder das Blättern in der Historie bereits eingegebener Befehle.

Trotzdem kann man das System auch sehr häufig noch aus dem Rettungsmodus heraus starten. Das ist vor allem dann geboten, wenn man das System nicht anderweitig starten kann, weil gerade kein anderes

Startmedium zur Verfügung steht.

Das Auftreten des Rettungs-Modus deutet stets daraufhin, dass etwas mit der Installation von GRUB 2 grundlegend nicht stimmt, weswegen GRUB 2 dann immer neu installiert werden muss. Man kann also in einem solchen Fall das System alternativ auch gleich mit einer Desktop-CD starten und dann GRUB 2 direkt neu installieren, wobei dabei unbedingt nach der chroot-Methode vorgegangen werden muss.

Starten mit Hilfe des Rettungsmodus

Hinweis: Da der Befehlsumfang im Rettungsmodus deutlich reduziert ist, gestaltet sich die Analyse hier deutlich umständlicher. Man kann einzig mit dem Befehl `ls` Verzeichnis-Inhalte auflisten lassen und so dann manuell nach den richtigen Datenträgern und Verzeichnissen suchen. Hat man keinen richtigen Überblick über das eigene System, sollte man bevorzugt auf die Super-GRUB2-Disk zurückgreifen.

1. Zunächst verschafft man sich einen Überblick über die von GRUB 2 erkannten Datenträger:

ls

2. Außerdem müssen die Umgebungsvariablen `prefix` und `root` überprüft werden, da sie häufig den Start im Rettungsmodus verursachen können - z.B. nach dem das System umpartitioniert wurde ohne GRUB 2 anschließend neu zu installieren:

set

3. Gegebenenfalls `prefix`

set prefix=(hdX,Y)/boot/grub

4. und `root`

set root=(hdX,Y)

Der aktuellen System-Situation entsprechend anpassen.

Hinweis: Die nachfolgenden Befehle setzen voraus, dass `prefix` und `root` richtig gesetzt wurden, weil es andernfalls zu Fehlermeldungen bezüglich des Pfades kommt. Insbesondere die Variable `prefix` ist

zwingend erforderlich. Ist sie nicht oder nicht richtig gesetzt, so funktionieren die nachfolgenden Befehle unter Umständen nicht einmal dann, wenn man sie unter Verwendung des vollständigen Pfades eingibt!

5. Die Anpassungen noch mal überprüfen:

set

6. Zunächst probiert man, das Modul `normal.mod` aus dem grub-Verzeichnis zu laden:

insmod normal

Kommt es beim Laden des Moduls zu einer Fehlermeldung, so setzt man diese Anleitung mit Schritt 8 fort.

7. Konnte das Modul `normal.mod` erfolgreich geladen werden, so versucht man, den normalen GRUB-2-Modus in Gang zu setzen:

normal

Misslingt die Ausführung des normalen Modus, so geht es mit dem nächsten Schritt weiter. Gegebenenfalls muss dazu der Rechner neu gestartet werden. Falls der Rechner normal starten sollte, muss schließlich noch Schritt 12 befolgt werden.

8. Das Modul `linux.mod` aus dem unter `prefix` angegebenen Verzeichnis nachladen:

insmod linux

9. Nun den zu startenden Kernel über den Symbolischen Link auswählen:

linux /vmlinuz root=/dev/sdXY ro

10. Dann den Pfad zur Ramdisk-Datei angeben :

initrd /initrd.img

11. Schließlich das System starten:

boot

12. Nach erfolgreichem Systemstart muss GRUB 2 im laufenden System unbedingt neu installiert und anschließend die Datei grub.cfg neu erstellt werden.

Gelingt der Systemstart aus dem Rettungsmodus heraus nicht, dann muss GRUB 2 neu installiert werden bzw. durch ein vorhandenes Backup ersetzt werden.

Ursache für den automatischen Start im Rettungs-Modus

In den Rettungs-Modus fällt GRUB 2 automatisch zurück, wenn GRUB 2 keinen oder keinen vollständigen Zugriff auf das Verzeichnis /boot/grub oder die darin für das Ausführen der Kommandozeile notwendigen Module - insbesondere das Modul normal.mod - hat. Sein Erscheinen deutet stets auf Probleme bei der Installation von GRUB 2 hin. Das kann folgende Ursachen haben:

- Die Module sind aufgrund einer fehlerhaften GRUB-2-Paketinstallation gar nicht im Verzeichnis /grub abgelegt oder wurden beschädigt abgelegt.
- Die Variable prefix, die bei der Installation von GRUB 2 in den MBR und den verborgenen Bereich im Anschluss an den MBR dort hinterlegt wird und die den Pfad zum /grub-Verzeichnis mit den dort enthaltenen Modulen enthält einen falschen Pfad.
- GRUB kann das Dateisystem der Partition auf der das Verzeichnis /grub liegt nicht lesen.
- Die Grundinstallation in den MBR hat zwar funktioniert, der dort geschriebene Code enthält aber Fehler, so dass ein Auslesen der Variablen prefix und/oder root aufgrund von Fehlfunktionen nicht möglich ist.

Alternativer Systemstart mit Hilfe der Super-GRUB2-Disk

Mit der Super-GRUB2-Disk kann man ein nicht mehr über den internen GRUB-Bootloader startendes System starten. Die Super-GRUB2-Disk bietet eine GRUB 2 Shell, die von CD startet. Hinter den vorkonfigurierten Auswahlmenü-Einträgen der Super GRUB2 Disk, verbergen sich Skripte, die einem weitestgehend die System-Analyse abnehmen. So werden auf dem betreffenden Computer die installierte Betriebssysteme selbständig

gefunden oder aber dort liegende GRUB 2 Konfigurations-Dateien.

Webseite: www.supergrubdisk.org

Alternativer Systemstart mit Hilfe der Live-Distribution Rescatux

Rescatux ist ein spezialisiertes Rettungssystem für den Grub-Bootloader in den Versionen 1 und 2. Rescatux ist von den gleichen Entwicklern wie die Super-GRUB2-Disk, startet aber mit einem eigenen Reparaturwerkzeug – die Rescapp.

Hinweis: Rescatux muss entsprechend zum defekten System im 32 Bit-Modus oder im 64 Bit-Modus starten.

Das Live-System kann defekte oder überschriebene Bootloader von installierten Linux-Systemen wiederherstellen. Die Wiederherstellung des Grub-Bootloader kann man mit vielen Live-Systemen wiederherstellen, doch bei Rescatux erledigt ein grafisches Tool diese Aufgabe. Rescatux erkennt die installierten Systeme selbständig. Im Menüpunkt »Grub (+)« kann man mit »Restore Grub« einen neuen Grub-Bootloader in der Version 1 oder 2 schreiben – je nach installierter Linux-Distribution - in den MBR der Festplatte und dabei alle automatisch erkannten Betriebssysteme in das neue Bootmenü einbinden.

Die Funktion »Update Grub Menus« greift auf die bereits vorhandenen Konfigurationsdateien der jeweiligen Linux-Distribution auf der Festplatte zu, um Grub 2 anhand der Konfigurationsdateien im Originalzustand wiederherzustellen.

Für verschlüsselte Systeme und LVM (Logical Volume Manager, mehrere Festplatten zu einer logischen Festplatte zusammenschließen) ist Rescatux zur Zeit noch nicht geeignet (Stand: 2014).

Download: <http://sourceforge.net> → auf der Webseite nach Rescatux suchen

Dokumentation: www.supergrubdisk.org/rescatux/

siehe auch: `info grub`, `grub-mkconfig --help`, `update-grub --help`

H

help

Der Befehl `<Programmname> --help` liefert eine kurze Beschreibung des Programms. Dabei wird in der Regel auf den Zweck des Programms, Aufruf und die wichtigsten Optionen bzw. Tastenkombinationen eingegangen.

`<Befehlsname> --help` bzw. `<Befehlsname> -h`

Hinweis: Die Ausgabe umfasst in der Regel eine Bildschirmseite. Falls der Text länger sein sollte, übergibt man die Ausgabe einfach dem Programm `less`. Der Befehl lautet dann `<Programmname> --help | less`. Nun sollte es möglich sein mit den Pfeiltasten (`↑`, `↓`) zu scrollen. Man beendet `less` durch einen Druck auf die Taste [`q`].

head

Gibt ohne Parameter die ersten 10 Zeilen einer Datei aus.

`head [OPTIONEN] <Dateiname>`

`head -n 5 <Dateiname> ...` gibt die ersten 5 Zeilen

siehe auch: `man head`, `tail`

hostname

Mit `hostname` können Rechnernamen bzw IP-Adressen von Rechner ermittelt werden.

`hostname ...` zeigt den Rechnernamen

`hostname --long ...` zeigt den kompletten Domainnamen z.B. `tux.site`

`hostname --ip-address ...` IP-Adresse des Rechners anzeigen

siehe auch: Netzwerk manuell aufsetzen, `cat /etc/hostname`

halt

Shutdown, `halt` fährt den Rechner herunter.

siehe auch: `reboot`, `shutdown`, `init`

hdparm

Ruckelt die Film-DVD beim Abspielen, liegt die Lösung oft ganz nah. Mit

hdparm beschleunigen Sie DVD-Laufwerke, aber auch Festplatten und CD-ROM-Laufwerke.

Für die Benutzung von **hdparm** brauchen Sie zuerst die internen Laufwerksbezeichnungen, die man über den Befehl **mount** erfährt. Im weiteren wird davon ausgegangen, dass das Laufwerk **/dev/hdb** heißt. Alle nachfolgenden **hdparm**-Befehle können nur mit Root-Rechten oder vorangestellten **sudo** ausgeführt werden.

hdparm -d /dev/hdb

/dev/hdb:

using_dma = 1 (on)

Die Ausgabe zeigt hier, dass der DMA-Modus eingeschaltet ist. Falls hier **using_dma** = 0(off) steht, sollte man zuerst testen welchen DMA-Modus das Laufwerk überhaupt unterstützt.

hdparm -i /dev/hdb

UDMA modes: **udma0 udma1 *udma2**

Hier sollte unter anderen diese Ausgabe stehen. An dieser Stelle kann man auch gleich ein Performance-Test durchführen.

hdparm -Tt /dev/hdb

Idealerweise sollten Sie diesen Test 2 bis 3-mal wiederholen. Nun kann man zur Tat schreiten und den DMA-Modus aktivieren.

hdparm -d1 /dev/hdb

Testen Sie nun Ihr Laufwerk. Stürzt Ihr Rechner ab, so sollten Sie besser auf eine Nutzung des DMA-Modus verzichten oder Sie wählen einen anderen DMA-Modus.

Achtung: Um das Feature bei jedem Neustart automatisch zu aktivieren, sind im Verwaltungszentrum ihrer Linux-Distribution oder direkt in den zuständigen Konfigurationsdateien einige Änderungen einzutragen.

Welche Performance-Optionen für Ihr Laufwerk überhaupt existieren und ob diese aktiv oder inaktiv sind, zeigt das Kommando

hdparm /dev/hdb

an, wobei Sie für **hdb** den Namen Ihres Laufwerkes eintragen.

siehe auch: man **hdparm**, INTERNET

Highlighting

Mit dem Programm **highlight** werden Quelltexte in Farben dargestellt (Highlighting). Nachfolgendes Beispiel verwendet das Farbschema von Kwrite. Der Quelltext in der Datei **shellscript.sh** wird eingefärbt und in der RTF-Datei **quelltext.rtf** gespeichert. RTF-Dateien können mit jedem Textverarbeitungsprogramm geöffnet und bearbeitet bzw. der Inhalt in ein anderes Textdokument kopiert und eingefügt werden. Die Formatierung der Quelltexte bleibt i.d.R. erhalten.

Zeilennummerierung ist 5-stellig (Vorgabe; **--linenumbers**), fehlende Stellen werden mit 0 (NULL; **--zeroes**) aufgefüllt:

```
highlight --linenumbers --zeroes --rtf --syntax=bash --style=kwrite --input=./shellscript.sh --output=quelltext.rtf
```

Zeilennummerierung ist 3-stellig (**--linenumbers**; **--line-number-length**), fehlende Stellen werden mit 0 (NULL; **--zeroes**) aufgefüllt:

```
highlight --linenumbers --line-number-length=3 --zeroes --rtf --syntax=bash --style=kwrite --input=./shellscript.sh --output=quelltext.rtf
```

--rtf ... Ausgabeformat ist RTF (Rich Text Format)

--input=./shellscript.sh ... Name der Quelldatei

--output=quelltext.rtf ... Name der Ausgabedatei

--style=kwrite ... gewählter Style; Darstellung wie im Editor KWrite

--syntax=bash ... gewählte Syntaxdarstellung; z.B. bash, sh, php, html, css, js (für Javascript) java, py (für Python), xml

highlight --list-themes ... Auflistung der unterstützten Styles

siehe auch: man highlight

history

Anzeige alle bisher eingegebenen und von der bash gespeicherten Kommandos (Vorgabe: Speicherung der letzten 1000 Kommandos).

history ... anzeigen aller bisher eingegebenen Kommandos

history > history.txt ... alle bisher im Kommandozeilenfenster eingetippten Kommandos können so in eine Textdatei kopiert werden

history -c ... löschen des gesamten Kommandozeilenspeichers

history -d 369 ... löschen des Kommandos Nr. 369

history | grep mount ... anzeigen aller Befehle die den Begriff mount enthalten

!48 ... Befehl mit der laufenden Nummer 48 in der history-Ausgabe wiederholen

!350:p ... Befehl aus history-Ausgabe bearbeiten; der Befehl mit der laufenden Nummer 350 rutscht in der Befehls-History auf den ersten Platz und mit einem Druck auf die Taste [Pfeil nach oben] kann der Befehl bearbeitet werden

siehe auch: man history

Hintergrundprozess starten

<MeinProgramm> &

Startet das Programm im Hintergrund, z.B. `kcalc &`. Ohne das `&`-Zeichen, würde der Prompt an der Kommandozeile erst nach dem Beenden des Programms wieder erscheinen (z.B. sehr nützlich bei der Erstellung von Shellskripts).

Je nachdem, wie das Terminal eingestellt ist, erscheinen die Meldungen des Hintergrundprogramms auf dem Bildschirm oder das Hintergrundprogramm wird für die Ausgabe angehalten.

Durch die Umleitung des Standardausgabekanals in eine Datei kann die Ausgabe eines Hintergrundprogramms auch für eine spätere Bearbeitung gespeichert werden.

Beispiel:

```
sort adressen | uniq > Adressenliste &  
[1] 19365
```

Im Beispiel wird eine sortierte Adressenliste, die von doppelten Einträgen befreit wurde, in der Datei Adressenliste gespeichert. Die Zahlen, die nach dem Abschluss der Kommandozeile erscheinen, werden von der Shell ausgegeben - [1] ist die Jobnummer und 19365 ist die Prozessnummer des gerade im Hintergrund gestarteten Prozesses.

Es gibt noch eine weitere Möglichkeit, Prozesse in den Hintergrund zu bringen. Mit der Tastenkombination [Strg] + [z] können im Vordergrund laufende Programme angehalten werden. Die Shell zeigt wie oben die Jobnummer und zusätzlich den Status Stopped und den Namen des angehaltenen Prozesses an und gibt schließlich wieder eine Eingabeaufforderung aus.

Mit den sogenannten Jobcontrol-Funktionen können die Prozesse im Hintergrund oder Vordergrund weiterlaufen.

Beispiel:

1. `sort adressen | uniq > Adressenliste &`
2. Mit `[Strg] + [z]` wird der Prozess angehalten.
3. Ausgabe: `[1]+ Stopped sort adressen | uniq > Adressenliste &`
4. `bg %1 ...` der Prozess läuft im Hintergrund weiter - `%1` bezeichnet die Jobnummer
5. Ausgabe: `[1]+ sort adressen | uniq > Adressenliste &`

Das Shellkommando `fg` setzt den gleichen Prozess im Vordergrund fort, z.B. `fg %1`

Hinweis: Mit dem Kommando **jobs** kann man sich alle laufenden Hintergrundprozesse anzeigen lassen.

Wenn zum Zeitpunkt des Ausloggens noch Jobs im Hintergrund laufen, werden diese Jobs beim Verlassen der Shell automatisch an den `init`-Prozess übereignet und laufen so weiter (z.B. ein User meldet sich ab und meldet sich anschließend mit einem anderen Namen wieder an).

host

Der Befehl **host `www.domainname.de`** zeigt die zu diesem Server passende IP-Adresse. Das funktioniert auch umgekehrt: Geben Sie als `host`-Parameter eine IP-Adresse ein, erfahren Sie den Servernamen. Allerdings erhalten Sie in dieser Richtung häufig keine Antwort, da nicht jede IP-Adresse einen Namen trägt.

host -a ... zeigt alle verfügbaren Informationen

Mit den Kommandozeilentools `ifconfig`, `host`, `route`, `ping`, `traceroute` bzw. `mtr` und `netstat` diagnostizieren Sie bei Netzwerkstörungen - richtig eingesetzt - gezielt jeden Teilaspekt einer Verbindung, intern oder im Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert - die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So finden Sie möglicherweise heraus, ob es sich bei einer nicht funktionierenden internen oder Internet-Verbindung um ein Problem mit einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS oder eines der Gegenseite handelt.

siehe auch: man `nslookup`, `ifconfig`, `route`, `ping`, `traceroute` bzw. `mtr` und `netstat`, man `dig`

hosts.allow, hosts.deny

Die Datei `hosts.allow` dient der Zugangskontrolle von Nutzern/Diensten anderer Rechner auf den zu schützenden Rechner. Für bestimmte Hosts/Netzwerke kann hier der Zugriff auf bestimmte lokale Dienste explizit gestattet werden.

Die Dateien `/etc/hosts.allow` und `/etc/hosts.deny` (auch als `hosts_access`-Dateien bezeichnet) werden u.a. vom TCP-Wrapper (Schnittstelle zu den einzelnen Anwendungen) ausgewertet. Dabei wird zuerst die `hosts.allow` und dann die `hosts.deny` ausgewertet. Es gilt: Wurde etwas in der Datei `hosts.allow` explizit gestattet, so kann es in der Datei `hosts.deny` nicht mehr verboten werden. Wenn dennoch ein entsprechender Eintrag in Datei `hosts.deny` steht, wird er ignoriert.

Der TCP-Wrapper (`tcpd`) bietet Diensten eine hostbasierte Zugriffskontrolle an, die speziell durch `inetd` (wird auch als Super-Daemon bezeichnet) gestartete Dienste nutzen. Aber auch einige Standalone-Server wie `ssh` haben den TCP-Wrapper-Zugriff einkompiliert. Für den TCP-Wrapper-Zugriff verwenden sie die `libwrap`-Bibliothek. Über das Terminal-Programm `ldd` lässt sich herausfinden, ob ein Dienst den TCP-Wrapper verwendet.

`ldd /usr/sbin/ssh | grep wrap`

Die sicherste Methode für die Regeln in den `hosts_access`-Dateien (`hosts.allow`, `hosts.deny`) ist folgende Vorgehensweise: in der Datei `/etc/hosts.deny` ist der Eintrag **ALL: ALL** aufzunehmen - der erstmal alles verbietet und anschließend sind in der Datei `/etc/hosts.allow` die Ausnahmen einzutragen.

Syntax

Beide Dateien (`hosts.allow` und `hosts.deny`) benutzen dieselbe Syntax, nur dass eben die Logik entgegengesetzt funktioniert.

Leerzeilen und Kommentarzeilen (mit `#` am Anfang) werden ignoriert.

Achtung: Am Ende der jeweiligen Dateien muss unbedingt ein Zeilenvorschub stehen. Steht am Ende der Datei eine Regel ohne abschließenden Zeilentrenner, so wird diese nicht korrekt verarbeitet.

Jede Zeile besteht aus dem Namen des Services, einem Doppelpunkt und einer Liste der berechtigten oder unberechtigten Clients und einem optionalen Kommando.

<service list> : <host list> [: command]

In der Service-Liste muss der Name des Programms (sshd, in.smtpd, in.fingerd, ftpd, in.telnetd) eingetragen werden, **nicht** der Name des Ports oder des Dienstnamens (ssh, smtp, finger, ftp, telnet), auf dem der Dienst lauscht! Das spezielle Wort ALL bezeichnet alle Dienste bzw. in der Host-Liste alle Hosts (Rechner).

ALL: 192.168.12.3 ... Eintrag in der hosts.allow - Zugriff von 192.168.12.3 auf alle Protokolle erlauben

ALL: 192.168 ... Eintrag in der hosts.allow - Zugriff von allen Adressen aus diesem Netzwerk zulassen (z.B. 192.168.1.1, 192.168.12.3 etc.); alle Protokolle sind erlaubt

ALL: 192.168.12.3 ... Eintrag in der hosts.deny - Zugriff von 192.168.12.3 auf alle Protokolle verweigern

Eintrag in der hosts.deny - Zugriff auf alle Protokolle verweigern

ALL: 74.52.109.98

ALL: 203.200.160.166

ALL: 70.84.177.26

ALL: 192.168.12.0/255.255.255.0 192.168.13.14 ... es können auch mehrere IPs, sowie Adressenbereiche angegeben werden

Sowohl für Dienste als auch für Hosts lassen sich folgende Wildcards nutzen:

ALL ... Feld - Service-Liste: gilt für alle Dienste; Feld - Hosts-Liste: gilt für alle Hosts

LOCAL ... gilt für alle Hosts in deren Namen kein Punkt (.) enthalten ist

UNKNOWN ... trifft auf jeden Benutzer dessen Name unbekannt ist und auf jeden Host dessen Namen oder Adresse sich nicht auflösen lässt

KNOWN ... trifft auf jeden Benutzer dessen Name bekannt ist und auf jeden Host dessen Namen oder Adresse sich auflösen lässt

PARANOID ... trifft auf jeden Host zu, dessen Name bei der Reverse-Namensauflösung (von IP-Adresse in einen Namen) nicht zur Adresse passt; ein mit -DPARANOID kompilierter tsdp (Default) verwirft ohnehin jede Anfrage solcher Clients

EXCEPT ... mit dem Operator EXCEPT werden Ausnahmen von der Regel definiert (**siehe:** Beispiele)

Regeln in der Datei /etc/hosts.allow haben eine höhere Priorität, als die Regeln in der Datei /etc/hosts.deny.

Regeln die in den beiden Dateien hosts.allow und hosts.deny weiter oben stehen besitzen eine höhere Priorität.

Hinweis: Die Abarbeitung der Regeln wird bei der ersten Übereinstimmung sofort abgebrochen.

Beispiele:

Datei: /etc/hosts.deny

alles verbieten, was nicht explizit erlaubt wird

ALL: ALL

Datei: /etc/hosts.allow

alle Verbindungen vom selben Rechner (Localhost) erlauben

ALL: localhost

Mail ist jedem gestattet

in.smtpd: ALL

Telnet und FTP wird nur Hosts derselben Domain und

dem Rechner tuxhausen erlaubt

in.telnetd, ftpd: LOCAL, tuxhausen.outside.all

Finger ist jedem Rechner aus der lokalen Domain erlaubt,

aber root wird per Mail darüber informiert

in.fingerd: LOCAL: (finger @%h | mail -s "finger from %h" root)

unseren Hallo-Welt-Service für zwei bestimmte Rechner und

ein Subnetz freigeben

echo: 192.168.0.0/24 10.2.3.4 testserver.woauchimmer.de

SSH für ein lokales Netz erlauben; Das verwendete Schlüsselwort

EXCEPT schließt einen einzelnen Rechner aus einer vorher

genannten Gruppe aus. Somit ist es möglich, seine Dienste nur

für diejenigen anzubieten, die sie wirklich nutzen (und nicht ausnutzen).

sshd: 192.168.1.0/24 EXCEPT 192.168.1.76

Datei: /etc/hosts.allow

```
# allen Hosts wird der Zugriff auf den telnet-Dienst verwehrt,  
# mit Ausnahme der Hosts aus dem lokalen Netzwerk  
in.telnetd: ALL EXCEPT LOCAL : DENY
```

Anstatt nur den Zugang zu erlauben oder zu verbieten kann man auch Terminal-Kommandos ausführen lassen. Dafür stehen eine Reihe von Wildcards zur Verfügung (z.B. %a, %h). Die vollständige Liste der Wildcards findet man im Manual - man 5 hosts_access.

Datei: /etc/hosts.allow

```
# Mit spawn führt der TCP-Wrapper den angegebene Befehl aus.  
# Im Beispiel schickt er eine Mail an root, wenn sich jemand per telnet  
# auf dem Rechner anmeldet und der Verbindungsaufbau wird abgelehnt.  
in.telnetd: ALL : spawn (echo "Telnet-Zugriff von %h" | mail -s "Telnet-  
Zugriff" root) & : DENY
```

Datei: /etc/hosts.allow

```
# Im Beispiel erhält der Benutzer, der sich per telnet am System  
# anmeldet nur einen Hinweis, dass telnet unsicher ist.  
in.telnetd: ALL : twist (echo "Telnet ist unsicher. Benutze ssh.")
```

Datei: /etc/hosts.allow

```
# Mit banners /verzeichnis sucht der TCP-Wrapper im angegebenen  
# Verzeichnis nach einer Datei, die so heißt wie der entsprechende Dienst  
# (z.B. telnetd) und gibt den Inhalt der Datei an den Client zurück.  
in.telnetd: ALL : banners /data : DENY
```

siehe auch: man hosts_access bzw. man 5 hosts_access, man hosts_options, chroot, UFW, Uncomplicated Firewall

htop

htop ist ein übersichtlicher Systemmonitor (Prozessor-Aktivität, Speicherauslastung, ...) für die Linux-Shell, mit dem man z.B. Leistungstests für den eigenen Server überwachen kann. Der Systemmonitor kann ohne root-Rechte aufgerufen werden.

htop ... startet den Systemmonitor; durch die Taste [Q] wird der Systemmonitor beendet

siehe auch: man htop, ab, siege

I

iconv

Kodierung von Dateien in eine andere Kodierung konvertieren.

iconv [Option ...] [Datei ...]

- f ... Kodierung für den ursprünglichen Text, z.B. latin1
- t ... Kodierung für die Ausgabe, z.B. utf-8
- l ... Liste mit möglichen Kodierungen
- o <Dateiname> ... Speicherung in eine Ausgabedatei bzw.
- output=<Dateiname>

iconv -f latin1 -t utf-8 ./text.txt > ./text3.txt

Konvertierung der Textdatei text.txt von iso-8859-1 oder Latin1 in das Format UTF-8 und anschließender Speicherung in der Textdatei text3.txt.

Achtung: Die konvertierte Datei muss in einer neuen Datei gespeichert werden, ansonsten wird die zu konvertierende Datei unwiederbringlich zerstört. Bei großen HTML- bzw. PHP-Dateien hat iconv Probleme - die Dateien werden nur teilweise konvertiert bzw. verstümmelt.

siehe auch: recode (Syntax: **recode VORHER..NACHHER**

<Dateiname>, z.B. recode UTF8..LATIN1 datei.php; **Achtung:** Diesen Befehl mit denselben Kodierungen nicht mehrmals auf die gleiche Datei anwenden - d.h. LATIN1..UTF-8 und danach nicht wieder LATIN1..UTF-8.)

Hinweis: Die Systemeinstellung für die aktuell verwendete Textkodierung ist in der Datei **/etc/sysconfig/language** zu finden (z.B. RC_LANG="de_DE.UTF-8"). Die aktuelle Einstellung kann mit dem Befehl **locale** abgefragt werden.

Beispiel:

Beim Öffnen einer Textdatei (hier im Beispiel: iso.txt) wird festgestellt, dass einige Zeichen nicht oder falsch dargestellt werden (z.B. deutsche Umlaute, Sonderzeichen).

Achtung: Die Textdatei nur schließen, auf keinen Fall mittels eines Textprogramms speichern, ansonsten wird diese Datei irreparabel beschädigt.

1. Ermittlung der Standard-Systemkodierung des eigenen Systems

locale

LANG=de_DE.UTF-8


```
LC_CTYPE="de_DE.UTF-8"  
LC_NUMERIC="de_DE.UTF-8"  
LC_TIME="de_DE.UTF-8"  
[...]
```

Das eigene System verwendet UTF-8 als Standardkodierung (LANG=de_DE.UTF-8).

2. Ermittlung der Kodierung der zu konvertierenden Datei

file -i iso.txt

iso.txt: text/plain; charset=iso-8859-1

Die Textdatei iso.txt verwendet die Kodierung ISO-8859-1.

oder

file -i iso.txt

iso.txt: text/plain; charset=unknown

Von der Textdatei iso.txt konnte die Kodierung nicht ermittelt werden.

Diese Textdatei wurde wahrscheinlich unter einem anderen Betriebssystem erstellt. Tritt dieser Fall ein, so kann man die Textkodierung nur erraten.

Beispiel: Die Textdatei kommt von einem System mit der wahrscheinlichen Systemsprache deutsch. In diesem Fall kann man davon ausgehen, dass die Textkodierung entweder UTF-8 oder iso-8859-1 ist.

3. Konvertierung der Datei iso.txt und Speicherung in einer neuen Datei utf.txt

iconv -f ISO-8859-1 -t UTF-8 iso.txt -o utf.txt

Zusatz: Dateinamen nach UTF-8 konvertieren

Dateien in Dateisystemen, die früher erstellt wurden, verwenden (sofern nicht anders angegeben) keine UTF-8-Kodierung für die Dateinamen.

Sollten diese Dateien andere als ASCII-Zeichen enthalten, werden sie nun »verstümmelt« angezeigt. Zur Berichtigung kann das Skript **convmv** verwendet werden, welches die Kodierung der Dateinamen nach UTF-8 umwandelt.

Hinweis für die Zukunft: Für Datei- und Verzeichnisnamen sollten nur das englische Alphabet (ohne das Leerzeichen), die Zahlen 0 ... 9, der Unterstrich () und der Punkt (.) verwendet werden. Für den Anfang von Dateinamen sollten auch keine Zahlen verwendet werden. Falls Zahlen notwendig werden, dann am besten mit einem Unterstrich beginnen - z.B.

_06TEST.txt. Groß- und Kleinschreibung kann verwendet werden, diese wird aber nur auf UNIX-artigen Systemen auch interpretiert (Windows-Systeme unterscheiden bis jetzt nicht zwischen Groß- und Kleinschreibung). Bei Einhaltung dieser Empfehlung dürften die Datei- und Verzeichnisnamen von allen bekannten Systemen seit 1970 und für die Zukunft richtig interpretiert werden.

siehe auch: Konvertierung von Textkodierungen, fromdos, todos, recode

id3

id3 (ID3 v1.1 Tag-Editor) ist ein kommandozeilenorientiertes Programm, dass ID3-Tags aus einer Datei (FLAC, MP3 und OGG/Vorbis) auflisten, ändern und löschen kann. Mit ID3-Tags können Musikdateien identifiziert werden. Sie können den Künstler, das Album, den Titel, die Tracknummer, das Jahr, das Genre sowie einen 28 Buchstaben langen Kommentar in einem Tag speichern.

Hinweis: Mit ID3v2 Tags kann id3 nicht umgehen.

id3 -l musik.mp3 ... ID3-Tag auslesen und die Liste am Bildschirm ausgeben

id3 -d musik.mp3 ... ID3-Tag löschen

id3 -d *.mp3 ... ID3-Tags von allen MP3-Dateien im aktuellen Verzeichnis löschen

id3 -L ... Liste mit allen Genres ausgeben

id3 -A 'Neuer Albumtitel' -a 'Künstlername' -T 7 -t 'Musiktitel' -y 2014 -g 8 -c 'Label: New Record XXL' musik.mp3 ... ID3 Tag neu setzen: Albumtitel (0 ... 28 Zeichen) – Künstlername (0 ... 28 Zeichen) – Track-Nummer (0 ... 255) – Musiktitel (0 ... 28 Zeichen) – Erscheinungsjahr (4 Zeichen) – Genre (0 ... 147 oder z.B. Jazz, Rock, Pop etc.; siehe: id3 -L) Kommentar (0 ... 28 Zeichen)

id3 -g 'Pop' musik.mp3 ... Genre neu setzen; für Genre können Zahlen (0 ... 147) oder vordefinierte Namen (z.B. Jazz, Rock, Pop etc.; siehe: id3 -L) genutzt werden; eigene Genre-Namen müssen über den Kommentar (-c 'Genre: K-Pop') gesetzt werden

Ein alternatives Programm mit grafischer Oberfläche ist Kid3 (Paketname unter KDE: kid3; ansonsten kid3-qt) bzw. das Kommandozeilenprogramm

id3tool.

siehe auch: man id3, id3v2

id3v2

id3v2 (ID3v1 und ID3v2 Tag-Editor) ist ein kommandozeilenorientiertes Programm, dass ID3-Tags aus einer Datei (FLAC, MP3 und OGG/Vorbis) auflisten, ändern und löschen kann. Mit ID3-Tags können Musikdateien identifiziert werden. Sie können den Künstler, das Album, den Titel, die Tracknummer, das Jahr, das Genre sowie einen Kommentar in einem Tag speichern.

Beispiele:

id3v2 -l musik.mp3 ... ID3-Tag auslesen und die Liste am Bildschirm ausgeben

id3v2 -C musik.mp3 ... ID3-Tag Version 1 in Version 2 umwandeln

id3v2 -s musik.mp3 ... ID3-Tag Version 1 löschen

id3v2 -s *.mp3 ... ID3-Tags Version 1 von allen MP3-Dateien im aktuellen Verzeichnis löschen

id3v2 -d musik.mp3 ... ID3-Tag Version 2 löschen

id3v2 -d *.mp3 ... ID3-Tags Version 2 von allen MP3-Dateien im aktuellen Verzeichnis löschen

id3v2 -D musik.mp3 ... ID3-Tag Version 1 und Version 2 löschen

id3v2 -L ... Liste mit allen Genres ausgeben

id3v2 -A 'Neuer Albumtitel' -a 'Künstlername' -T 7 -t 'Musiktitel' -y

2014 -g 8 -c 'Label: New Record XXL' musik.mp3 ... ID3 Tag neu setzen: Albumtitel – Künstlername – Track-Nummer – Musiktitel – Erscheinungsjahr (4 Zeichen) – Genre (0 ... 147 oder z.B. Jazz, Rock, Pop etc.; siehe: id3v2 -L) Kommentar; Hinweis: die 28 Zeichen-Grenze für die Texteingabe wurde in der Version 2 aufgehoben

id3v2 -g 'Pop' musik.mp3 ... Genre neu setzen; für Genre können Zahlen (0 ... 147) oder vordefinierte Namen (z.B. Jazz, Rock, Pop etc.; siehe: id3v2 -L) genutzt werden; eigene Genre-Namen müssen über den Kommentar (-c 'Genre: K-Pop') gesetzt werden

Ein alternatives Programm mit grafischer Oberfläche ist Kid3 (Paketname unter KDE: kid3; ansonsten kid3-qt) bzw. das Kommandozeilenprogramm id3tool.

siehe auch: man id3v2

if

Zur Verwendung in Shellskripts. Überprüft den Exit-Code des zuletzt ausgeführten Befehls, Bedingung nach dem Schlüsselwort **if**. Wenn der Code 0 (wahr) ergibt, also das Programm erfolgreich ausgeführt wurde, werden die Befehle nach dem Schlüsselwort **then** aufgerufen. Sonst, wenn das Programm einen anderen Wert als 0 (einen Fehlercode) ausgibt, werden die Befehl nach dem Schlüsselwort **else** ausgeführt. Häufig wird das if-Konstrukt zusammen mit dem Befehl **test** zur Prüfung einer Bedingung verwendet. Zum Beispiel, ob die Datei /etc/passwd existiert.

```
if <Bedingung, Befehl>; then <Befehl>; else <Befehl>; fi
```

```
if test -f /etc/passwd  
then echo "Die Datei /etc/passwd ist vorhanden"  
else echo "Die Datei /etc/passwd ist nicht vorhanden"  
fi
```

Wenn die Bedingung von test in eckigen Klammern eingeschlossen ist, kann auf das Wort test verzichtet werden:

```
if [ -f /etc/passwd ]  
then echo "Die Datei /etc/passwd ist vorhanden"  
else echo "Die Datei /etc/passwd ist nicht vorhanden"  
fi
```

Einige Argumente des Befehls test:

test <wert1>=<wert2> ... Überprüfung, ob die Zeichenkette wert1 und wert2 übereinstimmen.
test -d <Datei> ... Überprüfung, ob die Datei ein Verzeichnis ist.
test -f <Datei> ... Überprüfung, ob die Datei vorhanden ist.
test -r <Datei> ... Überprüfung, ob ich Leserecht für die Datei habe.
test -w <Datei> ... Überprüfung, ob ich Schreibrecht für die Datei habe.
test -h <Datei> ... Überprüfung, ob die Datei ein symbolischer Link ist.
test -s <Datei> ... Überprüfung, ob die Größe der Datei mehr als Null ist.
test -u <Datei> ... Überprüfung, ob die Datei das Bit SUID eingestellt hat.
test -x <Datei> ... Überprüfung, ob die Datei vorhanden und ausführbar ist.

siehe auch: Anhang: Einführung in die Shellprogrammierung

ifconfig

Zeigt die Netzwerkeinstellungen an und ermöglicht auch einige Änderungen zur Laufzeit. Änderungen mit ifconfig sind nur für die aktuelle Sitzung gültig, d.h. nach einem Neustart des Rechners werden wieder die alten Einstellungen verwendet.

ifconfig ... gibt alle aktiven Netzwerkgeräte aus

ifconfig -a ... gibt zusätzlich auch alle derzeit inaktiven Netzwerkgeräte aus

ifconfig <Netzwerkgerät> ... gibt nur Auskunft über das gewählte

Netzwerkgerät (z.B. eth0, eth1, ppp0, wlano, ippp0); eth0 - 1.

Netzwerkkarte, ppp0 - analoges oder DSL-Modem, wlano -

Funknetzwerkkarte, ippp0 - ISDN-Karte

ifconfig eth0 192.168.1.1 netmask 255.255.255.0 ... ändert die IP-Adresse und die Subnetzmaske des Netzwerkgerätes eth0; die Änderungen sind nur für die aktuelle Sitzung gültig, d.h. nach einem Neustart des Rechners werden wieder die alten Einstellungen - entsprechend der Konfigurationsdateien im Verzeichnis /etc - verwendet; dieser Befehl kann nur mit root-Rechten oder vorangestellten sudo ausgeführt werden

Beispiel:

Sie wollen Ihren Rechner kurzzeitig innerhalb eines fremden Netzwerkes (z.B. Internet-Café) betreiben, einschließlich Internet-Zugang. Dazu geben Sie folgende zwei Befehle an der Kommandozeile als Benutzer root ein (freie IP-Adresse: 192.168.1.100, Subnetz-Maske: 255.255.255.0, IP-Adresse des Internet-Gateway's: 192.168.1.2):

```
ifconfig eth0 192.168.1.100 netmask 255.255.255.0  
route add default gw 192.168.1.2
```

Mit dem Befehl

```
route del default
```

können Sie die Default-Route wieder löschen. Sie können den Rechner aber auch einfach herunterfahren, um die sitzungsbezogenen Änderungen wieder rückgängig zu machen.

Hinweis: ifconfig muss gegebenenfalls für normale Benutzer mit der vollen Pfadangabe aufgerufen werden, z.B. /sbin/ifconfig.

In der ifconfig-Ausgabe sehen Sie in der ersten Zeile hinter dem

Gerätenamen das Protokoll und im Falle von LAN- oder WLAN-Karten die Hardware-spezifische MAC-Adresse (Media Access Control). Darüber identifiziert sich jedes Netzwerkgerät eindeutig; nach außen hin kann der Anwender diesen Eintrag jedoch manipulieren.

In der nächsten Zeile erfahren Sie die IP-Adresse des Geräts. Bei lokalen Netzen folgen unter »Bcast« Informationen über den verwendeten Netzwerkteilbereich. Point-to-Point-Geräte, die sich ins Internet einwählen, notieren stattdessen unter P-t-P die Adresse der Gegenseite.

Neben den realen Netzwerkgeräten taucht in der ifconfig-Ausgabe ein Eintrag namens lo auf. Unter Protokoll steht hier lediglich »Lokale Schleife«, denn es handelt sich um ein virtuelles Gerät, das ein rechnerinternes Netzwerk aufbaut. So erreichen Programme auf dem eigenen Rechner laufende Dienste über die IP-Adresse 127.0.0.1 (localhost). Mit den Kommandozeilentools ifconfig, host, route, ping, traceroute bzw. mtr und netstat diagnostizieren Sie bei Netzwerkstörungen - richtig eingesetzt - gezielt jeden Teilaspekt einer Verbindung, intern oder im Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert - die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So finden Sie möglicherweise heraus, ob es sich bei einer nicht funktionierenden internen oder Internet-Verbindung um ein Problem mit einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS oder eines der Gegenseite handelt.

Virtuelle Netzwerkkarten einrichten

Mit ifconfig kann man einer Netzwerkkarte auch mehrere Ip-Adressen zuweisen.

reale Netzwerkkarte:

```
ifconfig eth0 192.168.100.1 netmask 255.255.255.0
```

virtuelle Netzwerkkarten:

```
ifconfig eth0:0 192.168.100.2 netmask 255.255.255.0
```

```
ifconfig eth0:1 192.168.100.3 netmask 255.255.255.0
```

Diese virtuellen Netzwerkkarten können mit

```
ifconfig eth0:0 down
```

bzw.

```
ifconfig eth0:1 down
```

heruntergefahren werden, d.h. sie sind dann unter dieser IP-Adresse nicht mehr ansprechbar. Das Hochfahren der virtuellen Netzwerkkarten geschieht wieder mit dem Befehl

```
ifconfig eth0:0 192.168.100.2 netmask 255.255.255.0
```

bzw.

```
ifconfig eth0:0 192.168.100.3 netmask 255.255.255.0
```

Um alle Netzwerkkarten, die reale wie auch die virtuellen Netzwerkkarten, gleichzeitig herunterzufahren und auch wieder hochzufahren benutzen sie folgende Befehle.

```
ifconfig eth0 down
```

bzw.

```
ifconfig eth0 up
```

Hinweis: Beim Herunterfahren des Rechners, werden die sitzungsbezogenen Änderungen gelöscht.

WLAN: MAC-ACL unwirksam

Einige Access Points (AP) erlauben es ihren Admins, MAC-ACLs (Access Control Lists) einzusetzen. Die APs akzeptieren dann nur noch bestimmte MAC-Adressen. Diesen Zugangsschutz kann man aber sehr einfach umgehen, ein ifconfig-Aufruf setzt die MAC-Adresse beliebig:

```
ifconfig eth0 down
```

```
ifconfig eth0 hw ether 00:02:AB:01:AB:01
```

```
ifconfig eth0 up
```

Das funktioniert bei den meisten Treibern und Geräten; denn MAC-Adressen sind in der Regel nicht fest in die Hardware eingegraben. Ein MAC-Filter verschreckt daher einen beharrlichen Angreifer kaum, der Eindringling muss lediglich eine gültige Adresse kennen. Die erfährt er, wenn er das Netzwerk abhört.

Damit sind umfangreiche MAC-ACLs eher eine organisatorische Plage als ein größerer und tatsächlicher Sicherheitsgewinn. Auch der Versuch, auf den DHCP-Server zu verzichten und stattdessen die IP-Adressen statisch festzulegen, bringt keine Sicherheit. Programme wie Kismet ermitteln den

IP-Bereich auch ohne Hilfe eines DHCP-Servers. Besonders in großen Netzen sind wieder organisatorische Probleme zu erwarten, die sich kaum rechtfertigen lassen.

Die Unsicherheit der Funknetze wirkt sich oft auch auf das kabelgebundene Netz aus, an das sie angeschlossen sind. Mit ARP-Cache-Poisoning-Attacken beeinflussen Angreifer auch den Datenverkehr im Kabelnetzwerk.

siehe auch: Netzwerkkarte, host, route, ping, ip, traceroute, mtr, netstat, Netzwerk manuell aufsetzen

ip

ip ist der mächtige Nachfolger von arp, ifconfig und route.

ip addr show ... zeigt alle aktiven Netzwerkgeräte, einschließlich der IP- und der MAC-Adressen

ip address ... zeigt die eigene IP- und MAC-Adresse

ip addr ... zeigt die eigene IP- und MAC-Adresse

ip a ... Kurzform von ip address und ip addr

ip -br addr ... zeigt die eigene IP—Adressen (IPv4 und IPv6)

ip -br link ... zeigt alle Netzwerkkarten und deren MAC-Adressen an - aktive, inaktive und virtuelle Netzwerkgeräte (-br steht für Brief)

ip link show ... zeigt alle Netzwerkkarten und deren MAC-Adressen an - aktive, inaktive und virtuelle Netzwerkgeräte

ip route show ... zeigt die Routen in einem Netzwerk an

ip neighbour show ... ARP-Cache anzeigen

ip neigh show ... neigh ist die Abkürzung für neighbour (Nachbar)

ip addr del 192.168.1.1 dev eth0 ... IP-Adresse der Netzwerkkarte eth0 löschen

Beispiel:

Sie wollen Ihren Rechner kurzzeitig innerhalb eines fremden Netzwerkes (z.B. Internet-Café) betreiben, einschließlich Internet-Zugang. Dazu geben Sie folgende zwei Befehle an der Kommandozeile als Benutzer root ein (freie IP-Adresse: 192.168.1.100, Subnetz-Maske: 255.255.255.0, IP-Adresse des Internet-Gateway's: 192.168.1.2):

ip addr add 192.168.1.100/24 dev eth0

ip route add default via 192.168.1.2

Anmerkung: 24 => Subnetz-Maske 255.255.255.0, 16 => Subnetz-Maske 255.255.0.0, 8 => Subnetz-Maske 255.0.0.0

Mit dem Befehl

ip route del default via 192.168.1.2

können Sie die Default-Route wieder löschen. Sie können den Rechner aber auch einfach herunterfahren, um die sitzungsbezogenen Änderungen wieder rückgängig zu machen.

siehe auch: route, ifconfig, arp, man ip

ImageMagick

Nach der Installation von ImageMagick kann man dieses Programm nur über das Schnellstartfenster (Tastenkombination: [Alt] + [F2]) starten.

display logo:

Beim Erstellen einer Verknüpfung mittels rechter Maustaste, ist dieser Befehl ebenfalls zu verwenden (»Neu erstellen ...« »Verknüpfung mit Programm« »Ausführen« »Befehl«)

ImageMagick – Artikel Nr. 1

Anmerkung: ImageMagick bringt zahlreiche Programme mit, deren Stärken vor allem bei der Erstellung von Shellskripten zum Vorschein kommen.

siehe auch: Anhang: Einführung in die Shellprogrammierung

Zauberhafte Bildbearbeitung

Es muss nicht immer ein grafisches Bildbearbeitungsprogramm wie Gimp sein. Manches geht auf der Shell schneller – z.B., wenn Sie einen Schwung Foto's in Form bringen wollen. Wir stellen Tools vor, mit denen fast alles fast wie von selbst geht. von Heike Jurzik

ImageMagick enthält zahlreiche Programme, die Screenshots erstellen, Bilder in verschiedene Formate konvertieren, die Größe zurechtstutzen, Fotos rotieren und sogar beschriften. Viele grafische Programme nutzen die ImageMagick-Funktionen und -Bibliotheken im Hintergrund, aber die Tools machen auch im Alleingang etwas her. Wer nur zwei oder drei Fotos bearbeiten muss, fragt sich zurecht, warum er auf ein lieb gewonnenes Programm verzichten soll – gerade wenn die meisten Funktionen vertraut und bequem per Mausklick erreichbar sind. Liegen allerdings zahlreiche Fotos auf der Platte, bieten die ImageMagick-Tools eine schnellere Alternative. Neben ihren Standardfunktionen zeigen wir, wie Sie die Shellkommandos gleich auf eine ganze Gruppe von Dateien anwenden.

Format, Format

Das Programm **convert** ist ein echter Alleskönner – es wandelt verschiedene Grafikformate um, erstellt PostScript-Dateien, beschriftet Ihre Bilder und wendet Filter an. Um ein Foto beispielsweise vom TIFF- ins JPEG-Format zu konvertieren, tippen Sie:

convert bild.tif bild.jpg

Nach dem Befehlsaufruf folgt der Name der Originaldatei, und danach geben Sie den Namen der neuen Datei an. In welches Format **convert** umwandeln soll, erkennt das Programm an der Dateiendung **.jpg**; die Originaldatei **bild.tif** bleibt erhalten. Ein Blick in die Handbuchseiten von ImageMagick verrät, welche Grafikformate die mitgelieferten Programme (darunter auch **convert**) unterstützen.

Die richtige Größe

Wer Bilder für eine Galerie aufbereitet, beispielsweise um diese anschließend auf einer Webseite zu präsentieren, sollte die Größe der Bilder zurechtstutzen. Eine Auflösung von 1076 x 768 oder 800 x 600 reicht in der Regel völlig aus. Welches Format Ihre Fotos haben, finden Sie mit den Befehl **identify** heraus:

identify *.jpg

```
bild1.jpg JPEG 1600x1200 Direct Class 933kb 0.010u 0:01  
bild2.jpg JPEG 1600x1200 Direct Class 1033kb 0.000u 0:01  
[...]
```

Mit **convert -resize** verändern Sie die Größe – geben Sie die neuen Maße und die Namen von Original- und Zieldatei an:

convert -resize 1024x768 bild1.jpg bild1_neu.jpg

identify bild1_neu.jpg

```
Ausgabe: bild1_neu.jpg JPEG 1024x768 Direct Class 359kb 0.000 0:01
```

Rechts- oder linksherum?

Ebenso schnell rotieren Sie Bilder auf der Kommandozeile. Der folgende Befehl dreht beispielsweise ein Foto um 90 Grad im Uhrzeigersinn:

convert -rotate 90 bild2.jpg bild2.jpg

Dieser Befehl überschreibt die Originaldatei; wer auf Nummer sicher gehen will, gibt wieder einen anderen Dateinamen als letztes Argument an. Das Ergebnis betrachten Sie beispielsweise mit dem im ImageMagick-Paket

enthaltenen Betrachter **display**:

display bild2.jpg

Klicken Sie mit der linken Maustaste ins aufgehende Fenster, öffnet sich das englischsprachige Menü, über das Sie u.a. verschiedene Bildbearbeitungsfunktionen erreichen. Um **display** zu beenden, wählen Sie entweder im Menü **File / Quit** oder drücken die Taste **[Q]**.

Anmerkung: Um alle Bilder des aktuellen Verzeichnisses mit **display** anzuzeigen, gehen Sie wie folgt vor:

display *.jpg

Mit der **[Leertaste]** blättern Sie jeweils ein Bild vorwärts.

Beschriftungskünstler

convert eignet sich darüber hinaus hervorragend dazu, Bilder zu beschriften. Sollen etwa alle Fotos ein Wasserzeichen (z.B. mit dem Namen des Fotografen) erhalten, kombinieren Sie die folgenden vier Optionen:

- **-font:** Als Argument übergeben Sie diesem Parameter die gewünschte Schriftart, z.B. **-font arial** oder **-font helvetica**. Alternativ geben Sie eine .ttf-Datei (TrueType-Schriftart) an, z.B. **-font /usr/share/fonts/truetype/Comic_Sans_MS.ttf** bzw. **-font /usr/X11R6/lib/X11/fonts/truetype/Comic_Sans_MS.ttf**.
- **-pointsize:** Über diesen Parameter definieren Sie die Schriftgröße in Punkt, z.B. **-pointsize 40** – in der Regel ist hier ein wenig Experimentieren nötig, bis die Werte passen und das Ergebnis gut aussieht.
- **-fill:** Hier geben Sie die Farbe der Schrift an, z.B. **-fill red** (für auffälliges Rot) oder **-fill grey** (für helles Grau).
- **-draw:** Diese Option hilft beim Texten: Geben Sie als Argumente die Position in Pixeln und die Beschriftung an. Für die Positionierung rechnen Sie in Pixeln von der linken oberen Ecke ausgehend (0,0) nach rechts unten. Alles zusammen wird in Hochkommata eingeschlossen – enthält der Text Leer- und Sonderzeichen, wird dieser ebenfalls in Hochkommata geschachtelt, also z.B. **-draw "text 300,500 'Copyright by Huhnix'"**

Es empfiehlt sich, für die ersten Gehversuche das Ergebnis nicht in die Originaldatei zu schreiben. Zusammengesetzt heißt der komplette Befehl z.B.:

```
convert -font /usr/share/fonts/truetype/Comic_Sans_MS.ttf. -pointsize  
50 -fill grey -draw "text 300,500 'Copyright by Huhnix'" bild1.jpg  
bild1_beschriftung.jpg
```

Der passende Rahmen

Noch mehr Farbe bringen Sie ins Spiel, indem Sie einen bunten Rahmen um Ihre Fotos zeichnen. Die passenden Schalter heißen **-frame** und **-mattecolor**:

Hinter der ersten Option geben Sie die Breite des Rahmens an, und die zweite Option möchte über Ihr Argument erfahren, welche Farbe der Rand haben soll. Um oben und unten einen sechs Pixel breiten, roten Rahmen zu zeichnen, verwenden Sie folgenden Befehl:

```
convert -mattecolor red -frame 6x6 bild.jpg bild_rahmen.jpg
```

Dadurch vergrößert sich das Bild an allen vier Seiten: Ein etwa 800 x 600 großes Bild misst nach dem Einrahmen mit diesem Befehl beispielsweise 812 x 612 Pixel.

Alles automatisch

Die Kommandos können ihre Stärke vor allem dann ausspielen, wenn Sie gleich einen ganzen Schwung Fotos bearbeiten. Am besten geht das mit einer for-Schleife. Um etwa alle Dateien im aktuellen Verzeichnis, die auf **.jpg** enden, auf **20%** zu verkleinern und dem Ergebnis die Dateiendung **_s.jpg** zuzuweisen, tippen Sie beispielsweise:

```
for i in *.jpg; do convert -resize 20% $i `basename $i .jpg`_s.jpg; done
```

Was kompliziert aussieht, ist schnell erklärt: Die Schleife weist der Variablen **\$i** nacheinander alle Dateinamen zu, die auf **.jpg** enden (for i in *.jpg). Für alle diese Dateien wird nun **convert** mit dem Parameter **-resize** aufgerufen – statt einer festen Größe wie 800x600 definieren Sie hier einfach **20%**. Als Ausgangsdatei steht die Variable **\$i** – die Shell ersetzt diese bei jedem Schleifendurchlauf durch den Dateinamen eines der Bilder. Der Name der Zieldatei setzt sich aus dem »Basisnamen« ohne die Dateiendung und der neuen Erweiterung **_s.jpg** zusammen. Der Befehl schließt mit der Anweisung **done**. Gerade für solche Automatisierungen ist **convert** ein unverzichtbarer Helfer.

Bildschirmfotos mit »import« auf Kommando erstellen

Im Paket ImageMagick, das Kommandozeilenprogramme für die Bildbearbeitung mitbringt, finden Sie neben **convert**, auch einen Screenshot-Spezialisten. Das Tool heißt **import**, und die einfachste Form

des Aufrufs lautet **import screenshot.png**. Der Mauszeiger verwandelt sich in ein Kreuz, und sobald Sie ein Fenster anklicken, speichert **import** ein Bild davon mit dem Namen **screenshot.png** im aktuellen Verzeichnis. Ebenso einfach fotografieren Sie nur einen Ausschnitt des Bildschirms: Geben Sie wieder den oben genannten Befehl ein und ziehen Sie mit dem Fadenkreuz (Mauszeiger) einen Rahmen um den Bereich, den Sie als Screenshot speichern wollen. Sobald Sie die Maustaste loslassen, erzeugt **import** das Bildschirmfoto. Um ein Bild des gesamten Desktops aufzunehmen, verwenden Sie den Befehl

import -window root screenshot.png

Unter dem Root-Window versteht man das erste Fenster, das beim Start der grafischen Oberfläche geöffnet wird, also den Desktop selbst, obwohl dieser keinen Fensterrahmen hat und Sie ihn nicht verschieben können. Auch andere Fenster lassen sich mit dem Aufrufparameter **-window** fotografieren: dazu geben Sie anstelle von **root** den Fensternamen ein, der in der Titelleiste steht. Enthält er Leerzeichen, setzen Sie ihn in Anführungszeichen. Alternativ verwenden Sie statt des Namens die eindeutige Nummer des Fensters.

Fenster-ID mit »xwininfo« herausfinden

Die grafische Oberfläche von Linux gibt jedem Fenster eine eindeutige Nummer. Welche das ist, erfahren Sie mit **xwininfo**.

Sobald Sie diesen Befehl eingeben, verwandelt sich der Mauszeiger in ein Kreuz, mit dem Sie das gewünschte Fenster anklicken. Die Ausgabe ist ziemlich lang und enthält unter anderem auch Informationen zur Größe des Fensters. Die eindeutige Nummer steht in der ersten Zeile hinter **xwininfo: Window id:**

Zeitverzögerte Bildschirmfotos

Das Tool **import** hat einen Nachteil: Nach seinem Aufruf wechselt es sofort in den Aufnahmemodus. Es ist daher nicht möglich, beispielsweise ein Programm mit einem aufgeklappten Menü zu fotografieren oder einen Bildschirmschoner. Um ein solches Bild zu erhalten, starten Sie **import** in Kombination mit dem Befehl **sleep**. Nach Eingabe von

sleep 10; import -window root bild.png

haben Sie zehn Sekunden Zeit, den Desktop fotogen herzurichten. Erst nachdem **sleep** zehn Sekunden lang nichts getan hat, erzeugt **import** den Screenshot.

Ergänzungen:

identify -format %w bild.jpg ... ermittelt nur die Bildbreite von bild.jpg;

identify -format %h bild.jpg ... ermittelt nur die Bildhöhe von bild.jpg;

identify -format %wx%h bild.jpg ... ermittelt die Bildbreite und Bildhöhe von bild.jpg; Ausgabe: 640x480;

convert bild.jpg -resize 133% -quality 90 bild.jpg ... vergrößert das Bild um 33%; 100% - keine Vergrößerung; 80% - Verkleinerung auf 80%; dabei ersetzt das vergrößerte oder verkleinerte Bild das Original;

convert -resize 133% -quality 90 bild.jpg bild_neu.jpg ... vergrößert das Bild um 33%; 100% - keine Vergrößerung; 80% - Verkleinerung auf 80%; das vergrößerte oder verkleinerte Bild wird in der neuen Datei bild_neu.jpg gespeichert;

convert bild.jpg -crop 640x480+100+50 -quality 90 bild.jpg ... aus dem großen Bild bild.jpg wird ein Bildausschnitt (640x480 Pixel)

herausgeschnitten; die linke obere Ecke des Bildausschnitt's beginnt im großen Bild 100 Pixel von links und 50 Pixel von oben; dabei ersetzt der Bildausschnitt das Originalbild

convert -crop 640x480+100+50 -quality 90 bild.jpg bild_neu.jpg ... aus dem großen Bild bild.jpg wird ein Bildausschnitt (640x480 Pixel)

herausgeschnitten; die linke obere Ecke des Bildausschnitt's beginnt im großen Bild 100 Pixel von links und 50 Pixel von oben; der Bildausschnitt wird in der neuen Datei bild_neu.jpg gespeichert;

convert bild.jpg -sharpen 0.1x0.7 bild.jpg ... leichte Unschärfen des Bildes bild.jpg werden entfernt

siehe auch: man ImageMagick

* * * * *

info

Über das info-Hilfesystem findet man manchmal nur die selben Informationen wie mittels **man**. Nicht selten jedoch gehen die gebotenen Informationen weit über die der Manpages hinaus.

info <Programmname>

Vor allem bietet **info** ein anderes Bedienkonzept. Es wird die Möglichkeit der Verknüpfung von Texten (also des Setzens von Links) genutzt. Der Benutzer kann dabei durch eine Dokumenten-Hierarchie navigieren und sich Stück für Stück von generellen Aussagen ausgehend immer tiefer in das Thema einlesen. Der Aufruf erfolgt über die einfache Befehlszeile **info** <**Programmname**> bzw. **info**. Letztere Variante stellt einen Index der vorhandenen Themen zur Verfügung.

In jedem Falle werden in der oberen Zeile Informationen über die Hierarchie angezeigt. Von links nach rechts sind das: die zugehörige info-Datei, der Name des aktuellen Dokuments, der des nächsten (Link), der des vorhergehenden (Link) sowie ein Link eine Ebene aufwärts. Eine Einführung in **info** liefert ein Druck auf die Taste **[h]**. Interessant ist auch noch die Option **--apropos**. Mittels **info --apropos=<Suchbegriff>** kann man die Info-Seiten nach einem bestimmten Thema durchsuchen. Mit **[q]** (quit) wird info beendet.

info mount ... zeigt die Info-Seiten zum Terminalprogramm mount
info --apropos=html ... durchsucht die Indizes in allen Manuals nach dem Suchbegriff html

siehe auch: man info, info info

id

Gibt die Gruppenmitgliedschaft des aktuellen Benutzers aus.

id <Benutzername> ... zeigt auch die Gruppenmitgliedschaft anderer Benutzer an, die dem System bekannt sind

siehe auch: id --help

init (SysVinit)

Beim Wechsel der Runlevels werden die Stoppskripte des gegenwärtigen Runlevels ausgeführt und damit verschiedene in diesem Level laufende Programme beendet. Danach werden die Startskripte des neuen Runlevels ausgeführt.

Die Steuerung der Runlevels erfolgt in der Datei /etc/inittab

(**id:5:initdefault:** ... beim Start des Systems wird gleich in den grafischen Modus umgeschaltet; **ca::ctrlaltdel:/sbin/shutdown -r -t 4 now** ... wenn das **-r** ausgetauscht wird gegen **-h**, dürfen auch fremde Benutzer den Rechner mit **[Strg] + [Alt] + [Entf]** herunterfahren).

Um das Herunterfahren des Rechners über die Tastenkombination **[Strg] + [Alt] + [Entf]** abzuschalten, ist in der Datei /etc/inittab ein Kommentarzeichen vor der Zeile

Wirkung des 3-Finger-Salut's abschalten:

#ca::ctrlaltdel:/sbin/shutdown -r -t 4 now

zu setzen. Damit ist es nur noch dem Systemadministrator root erlaubt den Rechner mit **[Strg] + [Alt] + [Entf]** herunterzufahren bzw. einen Neustart zu veranlassen.

init 0 ... Systemhalt (engl. halt)
init S ... Einzelbenutzerbetrieb (engl. single user mode), vom Bootprompt aus mit US-Tastaturbelegung
init 1 ... Einzelbenutzerbetrieb (engl. single user mode)
init 2 ... lokaler Multiuserbetrieb ohne Netzwerk (engl. local multiuser without remote network)
init 3 ... voller Multiuserbetrieb mit Netzwerk (engl. full multiuser with network)
init 4 ... frei (engl. not used)
init 5 ... voller Multiuserbetrieb mit Netzwerk und grafischer Benutzeroberfläche (engl. full multiuser with network and xdm)
init 6 ... Systemneustart (engl. Reboot)

Für die Runlevels ist die Datei `/etc/inittab` sowie das Verzeichnis `/etc/rc.d/` und seine Unterverzeichnisse wichtig.

Das init-Programm

Das Programm `init` ist der für die korrekte Initialisierung des Systems zuständige Prozess; es ist sozusagen der »Vater aller Prozesse« im System. Unter allen Programmen nimmt `init` eine Sonderrolle ein: `init` wird direkt vom Kernel gestartet und ist immun gegen das Signal 9, mit dem normalerweise jeder Prozess »gekillt« werden kann. Alle weiteren Prozesse werden entweder von `init` selbst oder von einem seiner »Kindprozesse« gestartet.

Konfiguriert wird `init` zentral über die Datei `/etc/inittab`; hier werden die so genannten »Runlevels« definiert und es wird festgelegt, welche Dienste und Dämonen in den einzelnen Levels zur Verfügung stehen sollen. Abhängig von den Einträgen in `/etc/inittab` ruft `init` verschiedene Skripts auf, die aus Gründen der Übersichtlichkeit im Verzeichnis `/etc/init.d` zusammengefasst sind.

Der gesamte Hochlauf des Systems - und natürlich auch das Herunterfahren - wird somit einzig und allein vom `init` Prozess gesteuert; insofern lässt sich der Kernel quasi als »Hintergrundprozess« betrachten, dessen Aufgabe darin besteht, die gestarteten Prozesse zu verwalten, ihnen Rechenzeit zuzuteilen und den Zugriff auf die Hardware zu ermöglichen und zu kontrollieren.

Achtung: Eine fehlerhafte `inittab` kann dazu führen, dass das System nicht korrekt hochgefahren wird. Gehen Sie bei Veränderungen in dieser Datei mit äußerster Sorgfalt vor und behalten Sie immer eine Kopie einer intakten Datei.

Runlevels

Unter Linux existieren verschiedene Runlevels, die den jeweiligen Zustand des Systems definieren. Der Standard-Runlevel, in dem das System beim Booten hochfährt, ist in der Datei `/etc/inittab` durch den Eintrag `initdefault` festgelegt. Für gewöhnlich ist dies 3 oder 5; nach einer Standardinstallation ist dies der Runlevel 5.

Um zu einen späteren Zeitpunkt in einen anderen Runlevel zu wechseln, kann man `init` mit der Nummer des zugehörigen Runlevels aufrufen; das Wechseln des Runlevels kann nur von `root` veranlasst werden.

Die Init-Skripts

Die Skripts unter `/etc/init.d` unterteilen sich in zwei Kategorien:

- Skripts, die direkt von `init` aufgerufen werden: dies ist nur beim Booten der Fall sowie bei einem sofortigen Herunterfahren des Systems (bei Stromausfall oder durch Drücken der Tastenkombination `[Strg] + [Alt] + [Entf]` durch den Anwender).
- Skripts, die indirekt von `init` aufgerufen werden: Das geschieht bei einem Wechsel des Runlevels; es wird generell das übergeordnete Skript `/etc/init.d/rc` ausgeführt, das dafür sorgt, dass die relevanten Skripts in der korrekten Reihenfolge aufgerufen werden.

Alle Skripts befinden sich unter `/etc/init.d`. Die Skripts für das Wechseln der Runlevels befinden sich ebenfalls in diesem Verzeichnis, werden jedoch grundsätzlich als symbolischer Link aus einem der Unterverzeichnisse `/etc/init.d/rc0.d` bis `/etc/init.d/rc6.d` aufgerufen. Dies dient der Übersichtlichkeit und vermeidet, dass Skripts mehrfach vorhanden sein müssen, etwa weil sie in verschiedenen Runlevels verwendet werden.

Beispiel: Beim Verlassen des Runlevels 3 wird z.B.

`/etc/init.d/rc3.d/K40network` aufgerufen; `/etc/init.d/rc` ruft das Skript `/etc/init.d/network` mit dem Parameter `stop` auf. Beim Eintritt in den Runlevel 5 wird letztlich das gleiche Skript gestartet, diesmal jedoch mit dem Parameter `start`.

Die Skripts können auch direkt mit den Parameter `start`, `stop`, `restart`, `reload` und `status` aufgerufen werden – Beispiel:

`/etc/init.d/xinetd restart`.

inetd - der Superdämon

Dämonen sind die guten Geister von Linux. Sie bleiben im Hintergrund und erwachen nur zum Leben, wenn es für sie etwas zu tun gibt. Dann erledigen sie die ihnen zugedachten Aufgaben und begeben sich wieder zur Ruhe, bis

erneut ein Auftrag für sie herein flattert. Früher geisterten viele solcher Dämonen durch den Speicher des Rechners. Sobald das System gestartet wurde, rief man auch die Dämonen heran, selbst wenn der eine oder andere niemals wirklich etwas zu tun bekam. Jeder Server überwacht somit seine Ports, lag an einem etwas an, erzeugte er einen Kindprozess und vererbte diesem die Verbindung. Der Server schloss unverzüglich die Verbindung und erwartete neue Anforderungen ...

Heute überwacht der **inetd** alle Ports und erst bei Aktivierung an einem dieser Ports startet der inetd einen entsprechenden Server-Prozess, der die bereits eröffnete Verbindung mittels eines Dateideskriptors übergeben bekommt und nun direkt mit dem Client kommuniziert. Hat der Server seine Arbeit erledigt, beendet er sich und erst eine erneute Anfrage beim inetd kann ihn wieder heraufbeschwören. Als Fazit ist die meiste Zeit über nur noch ein Prozess aktiv - der **inetd**.

Der inetd ist über die Datei `/etc/inetd.conf` bzw. `/etc/xinetd.conf` konfigurierbar. Der inetd kann über `kill -HUP <Prozess-ID>` neu gestartet werden. Die Prozess-ID kann über **ps -ax | grep inetd** ermittelt werden. Kann die Prozess-ID nicht ermittelt werden, so kann der Dämon von root gestartet werden: `/usr/sbin/inetd` bzw. `/usr/sbin/xinetd`. Modifiziert man eine beliebige Konfigurationsdatei im System, so müssen die betreffenden Dienste neu gestartet werden.

Im Unterschied zu manch anderen Betriebssystemen muss Linux nicht zwangsläufig neu gebootet werden. Es genügt, den Prozessen ein Signal `SIGHUP` zu senden, woraufhin diese ihre Konfigurationsdateien neu einlesen. Bearbeitet man z.B. die `/etc/inetd.conf` bzw. `/etc/xinetd.conf`, informiert man den inetd wie folgt:

killall -HUP inetd bzw. **killall -HUP xinetd** oder **/etc/init.d/xinetd reload**.

Hinweis: Einige Linux-Distributionen installieren statt inetd xinetd. Der xinetd ist flexibler konfigurierbar und auch stärker auf Sicherheit ausgerichtet, als sein Vorgänger (das x steht für extended ... erweitert).

Aktivierung eines Init-Skripts

Das Init-System SysVinit erfordert zum aktivieren eines Skripts Links im Verzeichnis `/etc/rcX.d` (Debian/Ubuntu). Zur Verwaltung dieser Links gibt es auf einem Debian oder Ubuntu System das Tool **update-rc.d**, welches sich um das Anlegen der Links kümmert.

Dienste starten, anhalten, deaktivieren

service --status-all ... Auflistung aller Dienste, keine root-Rechte erforderlich

Laufende Dienste haben ein Plus-Zeichen (+) und deaktivierte eine Minus-Zeichen (-). Statuslose Systemdienste sind mit einem Fragezeichen (?) versehen.

Für die nachfolgenden Befehle zum Starten, Anhalten und Neuladen (start, stop, restart, reload) sind root-Rechte erforderlich.

sudo service [Dienstname] start ... Dienst starten bzw.

sudo /etc/init.d/[Dienstname] start ... Dienst starten

sudo service [Dienstname] stop ... Dienst anhalten bzw.

sudo /etc/init.d/[Dienstname] stop ... Dienst anhalten

sudo update-rc.d [Dienstname] disable ... Dienst dauerhaft deaktivieren

sudo update-rc.d [Dienstname] enable ... einen Dienst für den automatischen Start beim Booten aktivieren

siehe auch: systemd – Das Init-System, Upstart, Dienste starten, anhalten und deaktivieren, Autostart mit ROOT-Rechten

Interrupts

siehe auch: proc Dateisystem

ISO-8859-1

siehe auch: Konvertierung von Textkodierungen, iconv, fromdos, todos, recode



begin-base64 644 image.tar.gz
H4sIAAOaV1MAA+z7c3BkXR8/isYz4UxsTmxbE9ue2JnYtm3btm3btm3rdp73
eX+ounXPH/eqnvnqnK5kOt17z157ffHB6tVWJnp29jYGNEd/Jz5oAQ8WJqav
ZzoWJtr/9fm/DyA6egZ6JhY6RiZ6RiBaOjpGZnogPKb/M2/qvv97WzsdGzw8
ID0dcwOz/w/n/R8d/7/ow+rf/Fvo2Nk5U9kaW9rY6dnb2VLPG2jRMbPSUltZ
GP1/PcZXgpkZGf/f5p+OgZaRjpkRil6OmZaZmYGBmYEtZkH/A2UxAeLT/P5jf
/+Hj/+b5D5aREoaFwoAC/AkrKilgBwQEEQUEBIBxHQLWjkAihxLgCcyWT1Th
O+Dh9l0/CPAA0kpE1RZw/uLXLzAfaaoT4E1000EVO3ILQztHHRsDIEdHR2oT
C1NbPR0rA2pLG6PUC04MICA8IFGB3wpOaWfrjmnquju6Xn4t16+nnGjN/50uJ
Rmuomh2XlWZgEX2BV/HRRlOgf474E85p4f0ckfzebGchUhSBjWkW5todT7Ab
YKAaU05dXjfwF1r+1nL4MIQ+8t8c/nGhRvndWqCzWjCbXKizOGgbfLwFZIQ
ARIIxLurdFMkhlD0nx8TlejCdQ8yityof98QncmDH3sA+u+rGMZmNLz//k1I
5g8c8z+Oyp1f+Iag6yooYMYzd41BOVfBCnvENK5+7+u3dgOxieCQV+woSIQ0
pbm/PljtOwdXPKmzuwzGRifrLiDh7C4KuTI6OGeTAv5SjHNNWje54vhv3ytE
GONyGIdzCbovgkD1khSe7xoPuuHY3owMwxLRcKxchwsAqQgrXgiOhVfkE1Qm
Xzw5qOqBjQOUqvpCbqfRzCbmk+1DXOxB61uuPb5vYLYfHrroq4L07onWFq
A/tvm6qoV/ntpNnY8HOYNd8jwq7z6JmasrQH662zzE6W+TrtSDPa4Z4mxaT3
nrWJ3fD5WrJN8Z+dGmeVssy2ROMokzV3QychSmomAFnouf7ZPK9X2KMsII1p
k9w0rbh8QKYcivz5PWyvCXBwdRpCRULqiu9vDqoMYjkjYKw8Jsj1lbcuKq
C4Rzx1fkJHG8BXbmHy+9efGA2+LWtwGFZML+QksS5+E/Wxqb+E7j4bqV6VxA
56wKvM1LYamV0Hb2+6HBfCuCa3DI00ThkHt4Hpq+/BUoctE9/TA1gL4g+4U5
FWNj/Ne3ze6QnEvrbTos10jM1MP6MdLntR150ir+bc9e8C8vbukNUU1UYAw
nFZjEGzkfMKQ397IE41P0IfRytlHz9r+OEIKMk0Doyb3ixJyT+L+cE/hwZqw
25SYxOzC+HB9vWW0e1zf6KuHVpJLQUhoDxR+9AxFTYlkrqi0NdxvVVFm4Lfi
napObch4vDFePz8z5lI5zzxleTYtHhqtqwi+eEJJabk0snwtfHrfzvVnlra5
b5stijBUY0myfd6fEF7iwYPBGrdOI7K/QDSpfAT7ucfxnnnYMdo8f6lvYHLD
7F7Z3LFKtdZUR0s1hVnCd2dGZH/rklbnYaWO4/6qiqTLyU+I8l8sB0vHCl6J3
/fj0n0BzS+mQiMbklTUFRQajU/yqj5lJyy3/IzUsuygJIS23uCwhLeY/4gb1
y5ZMBDDJkQS46FgfnvKDciLIndzQ505Mlcd4QKhKIDwwk2dvJLK6gXH25frm
MU3hYRFLK4NMH7goH+8GkXmuLNTNYNIaki37XkHwE9WUgvnG6FYabdsjwLQt
kgLTB+yJdtPwHXYw0QhwsYMo3I7A0b+JsgjDdZpFSLGn60X4dYusoarTuem8
uBV64CKx7/r2YWDScixZM3tttWQjM5agnD0duUobMlrgzd2pTUXEmVdZmbAd
QybzlvrxSeCegzbwtrmZBu772qvICJabPaZ/Q4bR541Sd8km36kqtaxQahmU
KMLjg7krmg/gpUjbl/KJfM1ZUt+u3NU1U8WNJTweiQDhCs8ku9YRANzUrDT3
9hb4XNjw8Mv+4PfgBBnO41JWHO1LL7wjM8NCMCGjgnWipkyuXLgo7q6UmWw/
4asTRpflnSbiYr4XlokSbjUi8RaDuD03H8RRodeckZz+QT9xTCqHWTl3hAevw
RjzgcET/Ehbbv/oUpWoaqPkr/kP9wJ9wYqqLdntQF6KwTFdQfEfXh1CeqF4
qd19CwfY7z5S0qtNthiKoCB5rUteJUBkInh6eHcMDUzpe+YcFtvxwO3BEwZS
HuDNuy9suviC23eaokCR8osf4B8qnYkPKi8t220JGMchjvpHm6BWwEzgh2Hy
yDTEaa/7rEzDfTehU44Zhbd5y+HnFwQxaEO5ZMQdzhgpdUnQv21Jrlc6bm0
FmvyjHeESL/gnpB+wpr6q7z8YMewY8n9+pGwmXihEGEO8wnbG9PiNH9vbkIL
fnYfxHntMZTRPKUjsYmrg9mNLDYbnbsGZwL9wR+6oqfmpptprRd8KfjvDym
xdELLI+yTlmjd27EtpUxkHU5pfCvfzsOU4xe/Px4cxqxzfUDxe5DsB36QZVF
ug/2Ls5of5IN5VG3ixOiBtWT/ILTnij8g3lq3vXXD+wlXsnf6M3v7YkEYKCV
RrLdj07VBoYr7pTK1dq8xjWJD0sQF0uzgpgzXpK8c13v19c0n0lmc1SFtU7/
Z

Java

Mit java können Java-Applikationen aufgerufen und gestartet werden. Der Unterschied zwischen Java und den meisten anderen Sprachen ist der, dass Java nicht wirklich als Skript interpretiert und auch nicht wirklich zu einem Binärprogramm übersetzt wird. Java-Code wird vom Java-Compiler (javac) in eine Bytecode-Datei übersetzt. Eine Bytecode-Datei kann dann auf verschiedenen Systemen vom Java-Interpreter interpretiert werden. Diese »plattformunabhängige« Programmierung ist eines der gepriesenen Features dieser Sprache.

```
java [-options] -jar jarfile [args...]
java [-options] class [args...]
```

java <Programm> ... startet das angegebene Programm

java -version ... Javaversion ausgeben

siehe auch: man java

john**Passwortüberprüfung – Artikel Nr.1**

Passwörter können in Unix-Systemen noch nicht einmal vom Systemverwalter ermittelt werden, weil Linux die Passwörter nur verschlüsselt ablegt. Die zugehörige Verschlüsselungsfunktion ist eine Einwegfunktion, die kein Entschlüsseln vorsieht. Meldet sich ein Benutzer am System an, verschlüsselt Unix dieses Passwort und vergleicht es mit der in der Shadow-Datei abgelegten Version.

Es gibt trotzdem theoretisch ein einfaches Verfahren, die Passwörter zu knacken: Sie probieren einfach alle Möglichkeiten. Der dazu nötige Aufwand hängt sehr von der Passwortlänge ab. Beispiel: Bei einer Passwortlänge von 8 Zeichen beträgt die Zahl der möglichen Passwörter 218.340.105.584.896. Der Zeitbedarf liegt bei der heutigen Rechenpower bei einigen Monaten.

Die Sicherheit eines Passwortes hängt nicht nur von seiner Länge ab, sondern auch stark vom verwendeten Zeichensatz ab. Daher sollte ein Passwort aus Klein- und Großbuchstaben, Zahlen und Sonderzeichen bestehen.

Hinweis: Schon ein einzelner geknackter Zugang ist ein Sicherheitsrisiko. Einbrecher, die einen Zugang zu Ihrem System haben, können dort nach weiteren Schwachpunkten suchen.

Sie sollten daher regelmäßig versuchen, die Passwörter Ihrer Benutzer zu knacken, um wenigstens die unsichersten Kandidaten zu ermahnen. Beim Knacken kann das Programm **john** helfen, es muss i.d.R. erst nachinstalliert werden. Nach der Installation finden Sie das Programm selber unter **/usr/sbin/john** und seine Komponenten unter **/var/lib/john/**. Das Programm kann mit einem Wörterbuch zusammen arbeiten, es liefert auch eine englische Version mit. Sie müssten hier erst ein deutsches Wörterbuch erstellen. Hinweise dazu finden Sie im Verzeichnis **/usr/share/doc/packages/john/**.

Dieser Aufwand ist sogar unnötig, meist langt es sogar, mit den Daten in den Benutzerdateien zu arbeiten. Damit können Sie Passwörter knacken, die aus Namen oder Variationen davon bestehen.

Wechseln Sie in das Verzeichnis **/var/lib/john/**.

cd /var/lib/john/

Nun lassen Sie aus **passwd** und **shadow** eine einheitliche Datei montieren, im Beispiel heißt sie **passwd.john**:

unshadow /etc/passwd /etc/shadow > passwd.john

Wenn Sie **john** mit den Daten aus dieser Datei arbeiten lassen, werden Sie staunen, wie viele Passwörter er ermittelt.

john -single passwd.john

Mit diesem Befehl nutzt **john** nur die Benutzerdatenbank als Grundlage, keines der zusätzlich verfügbaren Wörterbücher.

Wenn Sie bereits viele Benutzer angelegt haben, dauert das Knacken schon eine Weile. Wenn Sie den Fortschritt kontrollieren wollen, drücken Sie einmal die Leertaste, worauf **john** den aktuellen Stand anzeigt.

Wenn Sie **john** unterbrechen, setzt er bei einem Neustart seine Arbeit an der Stelle fort, wo Sie ihn unterbrochen hatten. Die bereits geknackten Passwörter hält er in der Datei **john.pot** fest. Falls Sie erneut alle Passwörter testen wollen, müssen Sie die Datei vorher löschen.

Das Ergebnis der Arbeit von **john**, eine Liste der Benutzerdaten inklusive Passwort im Klartext, können Sie mit

john -show passwd.john

ansehen. **john** zeigt dabei nur Accounts, deren Passwort **john** ermitteln konnte.

Machen Sie sich immer wieder klar, dass Sicherheit kein Zustand ist, sondern ein anstrengender Prozess.

Für weitere Möglichkeiten steht Ihnen die Dokumentation von john zur Verfügung, z.B. automatische Versendung von Ermahnungen per E-Mail.

* * * * * * * * * *

john

Passwortüberprüfung – Artikel Nr.2

Wie Linux das Passwort ermittelt

Um die Güte abschätzen zu können, muss man die Art und Weise kennen, wie unter Linux Passwörter erzeugt und gespeichert werden.

Wenn Sie am Login-Prompt Ihr Passwort eintippen, so berechnet eine Funktion crypt die verschlüsselte Zeichenkette zum eingegebenen Klartext. Genau diese Zeichenkette vergleicht Linux mit einer in einer Passwortdatei gespeicherten Zeichenkette. Stimmen beide überein, ist das Passwort gültig (auch wenn modernere Verschlüsselungsverfahren eingesetzt werden, ändert das nichts - wie hier am Beispiel - am von crypt demonstrierten Prinzip).

Es mag verwundern, dass das verschlüsselte Passwort in einer Klartext-Datei abgelegt ist. Allerdings ist es extrem aufwändig, vom verschlüsseltem Text auf das Klartextpasswort zu schließen, während die Berechnung (Klartext verschlüsseln) in wenigen Schritten möglich ist. Eine Funktion, die diese Eigenschaft besitzt, nennt man daher auch Einwegfunktion. D.h. die Hinrechnung ist einfach, die Gegenrechnung aber unmöglich (es gibt (noch) keinen effizienten Algorithmus). Im Falle der meisten Linux-Systeme, kommt zum Verschlüsseln der Data Encryption Standard (DES) zum Einsatz.

Beispiel: Ich stelle Ihnen die Aufgabe die Primzahlen $113 \cdot 119$ ohne Zuhilfenahme eines Taschenrechners zu ermitteln. Sicherlich kein Problem. Die Gegenrichtung wäre, wenn ich Sie beauftrage, die Primfaktoren der Zahl 13447 zu finden. Für ein Computerprogramm ist das ein Pappenstil, aber Sie werden eine zeitlang darüber brüten. Wählen Sie nun anstatt dreistelliger Primzahlen welche mit 10000 Ziffern, so wird auch ihr Gigahertz-Prozessor ein paar Jahre am Limit werkeln...

Wie sicher ist das Passwort?

Sie könnten jetzt meinen, der DES wäre so ausgeklügelt, dass vom

verschlüsselten Text niemals auf den Klartext geschlossen werden könne. Sicher ist die Aussage richtig, wenn Sie die Anwendung einer Berechnungsvorschrift meinen.

Aber der Hacker geht anders vor. Er nimmt ein elektronisches Wörterbuch, verschlüsselt der Reihe nach jedes enthaltene Wort und vergleicht das Resultat mit dem verschlüsselten Vorbild. Als nächstes kodiert er alle Wörter rückwärts, dann modifiziert er die Klein- und Großschreibung, lässt verschiedene Wörter kombinieren, baut Ziffern und Sonderzeichen ein... Klar, dass er ein Programm verwendet, das all die Modifikationen nach einem bestimmten, der menschlichen Logik nachgeahmten Schema vornimmt. Die heutigen PC's verschlüsseln bis zu mehrere Millionen Wörter binnen einer Minute! Wetten, dass die Trefferquote ganz akzeptabel ist? Aber ganz so einfach ist es dann doch nicht, da die verschlüsselten Passwörter in aktuellen Distributionen wohl kaum noch in der Datei /etc/passwd zu finden sind, sondern meist die shadow-Suite zum Einsatz gelangt. Das Passwort steht nun in der Datei /etc/shadow, die nur noch für Root lesbar ist. Damit ist ein Angreifer am Lesezugriff auf die Shadow-Datei interessiert. Um hier seine Tricks auszuspielen zu können, benötigt er Zugang zum System. Womit wir wieder beim Passwortgebahnen der »normalen« Benutzer wären...

Der sicherste Weg, schwache Passwörter zu vermeiden, ist, sie von vornherein auszuschließen.

Die Programmpakete zur so genannten proaktiven Passwortprüfung liegen den hier beschriebenen Distributionen nicht bei. Programme wie »anpasswd«, »npasswd« und »passwd+« sind ein Ersatz für »passwd« und prüfen das vom Benutzer eingegebene Passwort nach bestimmten Regeln (Telefonnummern, Passwortlänge, Groß- und Kleinschreibung, Wörterbuchlisten...) und lehnen es ggf. ab.

Programme, die die verschlüsselten Passwörter in Passwortdateien verifizieren, finden Sie im Lieferumfang fast aller Distributionen. crack und die zugehörigen Wortlisten sind die traditionellen Bestandteile eines Passwortprüfsystems. Nachfolgend möchten wir auf das moderne »John the Ripper« eingehen.

John the Ripper

John vermag mit verschiedenen Verschlüsselungsschemen umzugehen, u.a. mit DES, Blowfish, MD5 und WinNT LM Hashes.

Des Weiteren versteht es verschiedene Modi:

- Wortliste: Es wird versucht, das Passwort anhand einer Wortliste und optionaler Regeln zu entschlüsseln. Solche Regeln können Buchstabendreher, Kombinationen aus Groß- und Kleinschreibung, zusammengesetzte Wörter usw. beinhalten. Die maximal zulässige Passwortlänge kann angegeben werden.
- Einfacher Modus: John überprüft das Passwort gegen das Nutzerkennzeichen, Informationen des GECOS-Feldes und »verbreitete« Passwörter (aus Datei /var/lib/john/password.lst)
- Inkrementeller Modus: Alle erdenklichen Zeichenkombinationen werden getestet. Die Passwortlänge kann wiederum beschränkt werden. Mit dieser Methode kann jedes Passwort entschlüsselt werden - vorausgesetzt, man bringt die nötige Geduld mit;-)
- Externer Modus: Hierbei handelt es sich um eine Schnittstelle, die den Einsatz anderer Crackmethoden ermöglicht. Auf diesen Modus soll nicht weiter eingegangen werden.

Einfacher Modus

Zur Demonstration wurden drei Accounts in die Passwortdatei aufgenommen:

- »user1« mit dem passwort »Nobody«
- »user2« mit dem Passwort »2resu«
- »user3« mit dem Passwort »foobar«

john wird im einfachsten Fall mit dem Namen der zu prüfenden Passwortdatei aufgerufen. Im Beispiel schränken wir die Suche auf die oben genannten Benutzerkennzeichen ein.

```
root@sonne> john -users:user1,user2,user3 /etc/shadow
```

```
Loaded 3 passwords with 3 different salts (Standard DES [24/32 4K])
```

```
2resu          (user2)
```

```
foobar        (user3)
```

```
[Strg] + [C]
```

```
Session aborted
```

Binnen Bruchteilen einer Sekunde wurden »foobar« und »2resu« gecrackt. »Nobody« ist mit dieser einfachen Methode nicht beizukommen. Durch Eingabe von [Strg] + [C] wird »john« abgebrochen; den Status der Bearbeitung speichert »john« in einer Datei »john.pot«. Durch Eingabe von:

```
root@sonne> john -restore
```

```
Loaded 1 password (Standard DES [24/32 4K])
```

wird an der Abbruchstelle mit der Suche fortgefahren.

Wortlistenmodus

Die Stärke des Modus hängt zum einen von den verwendeten Wortlisten (Textdatei mit einem Wort pro Zeile) und zum anderen von den Regeln der Datei »john.ini« ab. Wir setzen die Wortlistenprüfung auf das Passwort von »user1« an:

```
root@sonne> john --wordfile=/usr/dict/words -rules -users:user1
/etc/shadow
Loaded 1 password (Standard DES [24/32 4K])
Nobody(user1)
guesses: 1time: 0:00:00:04 100% c/s: 15008 trying: Nobleman -
Notation
```

Das - zugegeben recht simple - Passwort entdeckte »john« nach 0.04 Sekunden.

Inkrementeller Modus

Die Laufzeit von »john« ist unbegrenzt. Im inkrementellen Modus läuft es, bis das letzte Passwort erkannt oder der Vorgang explizit abgebrochen wurde. Bei Verwendung dieser Prüffart ist es deshalb ratsam, die Suche mit verringerter Priorität zu starten, um die »normale« Arbeit mit dem System nicht auszubremesen. Das nachfolgende Beispiel startet »john« mit einem »Nicelevel« von 19:

```
root@sonne> nice -n 19 john -i /etc/shadow
Loaded 2 passwords with 2 different salts (Standard DES [24/32 4K])
```

Hier noch einmal die wichtigsten Befehle:

john -single <Dateiname>

dieser Befehl startet John im Single-Mode

john -i <Dateiname>

dieser Befehl startet John im Incremental-Mode

john -w:name_der_wordlist <Dateiname>

dieser Befehl startet John im Wordlist-Mode

john -e:MODE <Dateiname>

dieser Befehl startet John im External-Mode unter Verwendung der bei LIST.EXTERNAL: definierten MODE Eigenschaften

john <Dateiname>

dieser Befehl läßt John zuerst den Single- dann den Wordlist und schließlich noch den Incremental-Mode durchlaufen

john -show <Dateiname>

dieser Befehl zeigt die entschlüsselten Passwörter an

Der Befehl **john** (ohne Zusatz) listet alle verfügbaren Parameter auf.

Regelmäßige Prüfung

Es spricht nichts dagegen, hin und wieder die Passwortsicherheit im System zu überprüfen. Mit entsprechendem Nicelevel gestartet, stört der Testlauf die tägliche Arbeit nur unmerklich. john kann jederzeit abgebrochen (z.B. wenn die Systemlast einen bestimmten Schwellwert übersteigt) und im letzten Zustand gestartet werden. Ein nettes Skript ist das dem Paket beiliegende **mailer** (liegt meist unter /var/lib/john), das automatisch Benutzer mit schwachem Passwort per Mail zum Wechsel auffordert.

* * * * * * * * * *

join

Fügt die Zeilen zweier Dateien zusammen, die ein gemeinsames Indexfeld besitzen. Enthält Datei 1 die Zeilen »Klaus Müller« und »Martin Wolf« und Datei 2 »Klaus Berlin« und »Martin Hamburg«, ergibt »join Datei1 Datei2« das Ergebnis »Klaus Müller Berlin«, »Martin Wolf Hamburg«.

join datei1.txt datei2.txt

siehe auch: man join, paste

Kernelversion

siehe auch: Version des Kernels, procinfo

Konvertierung von Textkodierungen

In dieser Beschreibung soll im Wesentlichen auf die Verwendung der Zeichensätze als 8-Bit-Zeichensätze eingegangen werden. Falls Sie Zeichen asiatischer Sprachen (Japanisch, Chinesisch usw.) eingeben möchten, können Sie spezielle Editoren verwenden, die auch in vielen Linux-Distributionen zur Verfügung stehen.

Ein 8-Bit-Zeichensatz umfasst 256 Zeichen und besteht im Wesentlichen darin, den US ASCII-Zeichensatz, der nur die ersten 128 von 256 möglichen Zeichen definiert, um weitere Zeichen zu erweitern. Ein Textzeichen belegt im Computerspeicher also 8 Bit. Da 128 Zeichen bei weitem nicht ausreichen, um die Sonderzeichen beispielsweise aller europäischen Sprachen aufzunehmen, werden die verschiedenen Sprachen in Gruppen zusammengefasst, und diese Gruppe wird dann mit einer Kurzbezeichnung bezeichnet. Der dazugehörige Zeichensatz wird nach der dazugehörigen Norm als »iso-8859-x« Zeichensatz bezeichnet, wobei das »x« eine Ziffer zwischen 1 und 15 ist. Die genaue Anordnung der Zeichen im Zeichensatz iso-8859-1 bzw. iso-8859-15 können Sie der Manual-Page von iso-8859-1() bzw. der Manual-Page von iso-8859-15() entnehmen. Die bekannteren Kodierungen sind nachfolgend aufgeführt, weitere können Sie der oben genannten Manual-Page entnehmen.

Wichtige Zeichensatzkodierungen**iso-8859-1**

Westeuropäische Sprachen: Spanisch, Deutsch, Schwedisch, Dänisch u. a.; für Finnisch und Französisch ist nunmehr iso-8859-15 besser geeignet

iso-8859-2

Zentral- und Osteuropa: Ungarisch, Tschechisch, Rumänisch, Polnisch, Deutsch u. a.

iso-8859-5

Kyrillische Zeichen für Russisch

iso-8859-7

Griechische Zeichen für Griechisch

iso-8859-9

Zeichen für Türkisch

iso-8859-15

Weitgehend wie iso-8859-1, aber z.B. mit dem Eurozeichen und besserer Unterstützung für Finnisch und Französisch.

Der Benutzer muss dann - je nach verwendeter Sprache - die passende Kodierung auswählen. Insbesondere bei der Übertragung von Texten zwischen verschiedenen Rechnern muss die verwendete Kodierung ebenfalls übertragen werden. Der Vorteil des Verfahrens liegt auf der Hand: Um Unterstützung für die regionalen Sonderzeichen zu erhalten, brauchen Sie nur die richtige Kodierung zu wählen und sofort können (fast) alle Programme diese Sonderzeichen darstellen, da fast alle Programme einen 8-Bit-Wert (ein Byte) zur Darstellung eines Textzeichens verwenden. Wird die falsche Kodierung gewählt, werden die Sonderzeichen allerdings falsch dargestellt. Bei den meisten X-Applikationen und auch beim KDE-Desktop können Sie die Kodierung des Zeichensatzes auswählen, meist zusammen mit der Konfiguration des zu verwendenden Zeichensatzes. In den X-Applikationen wird die Kodierung meist mit Encoding bezeichnet.

Der Nachteil dieser Lösung ist, dass manche Sprachkombinationen unmöglich sind: Sie können z.B. nicht ohne weiteres einen **deutschen Text mit Umlauten** verfassen, in dem Sie **russische Ortsnamen in Kyrillisch** erwähnen. Dieses Dilemma kann erst durch einen anderen Ansatz, die Verwendung von Unicode gelöst werden. Unicode (z.B. **UTF-8**) kodiert Zeichen - anders als ASCII - nicht mit einem, sondern mit 2 oder noch mehr Bytes, wodurch wesentlich mehr Zeichen dargestellt werden können. Erst durch die Verwendung von Unicode können Sie auch asiatische Sprachen mit mehr als 127 Zeichen wie Chinesisch, Japanisch oder Koreanisch auf dem Rechner darstellen. Der Nachteil dieser Lösung ist, dass der Großteil der existierenden Software nicht auf den Umgang mit diesen Zeichen vorbereitet ist und Sie nur mit spezieller Software Texte mit Unicode-Zeichen lesen oder selber schreiben können.

Es ist davon auszugehen, dass zukünftig mehr und mehr Programme Unicode-Zeichen unterstützen werden.

siehe auch: iconv, recode, fromdos, todos

kill <Prozessnummer>

Prozesse beenden. Prozessnummer über **ps -A** oder **ps x** ermitteln. Mit **kill -HUP <Prozessnummer>** wird der betreffende Prozess angewiesen seine Konfigurationsdateien neu einzulesen.

kill <Prozessnummer>

Manche Dienste (z.B. Webserver) erfordern den Neustart mehrerer Prozesse und evtl. weiterer Schritte. Aus diesem Grund verfassen viele Linux-Distributionen Shell-Skripts, die die entsprechenden Schritte automatisieren (**siehe auch:** Autostart mit ROOT-Rechten).

kill -KILL <Prozessnummer> ... beendet den betreffenden Prozess

kill -9 <Prozessnummer> ... beendet den betreffenden Prozess auf die harte Tour

siehe auch: man kill, pkill, skill, killall, xkill

killall

Funktioniert ähnlich wie der Befehl kill, nur das statt der Angabe der Prozessnummer (PID) der Name des Prozesses ausreicht, um alle sich so nennende Prozesse zu beenden.

killall <Prozessname>

Falls dies keine Wirkung zeigt, probieren Sie es noch auf die »harte« Methode

killall -9 <Prozessname>

-9 sendet statt eines TERM- ein KILL-Signal, wodurch der Prozess vom Betriebssystem entfernt wird. Dies beendet in nahezu allen Fällen die spezifizierten Prozesse.

Anmerkung: Ein weiterer mächtiger Verbündeter im Zusammenspiel mit **kill** ist **lsuf**. Dieses Tool listet auf, welche Prozesse auf eine Datei oder ein Gerät zugreifen. So kann es passieren, dass Linux sich weigert, die CD-Schublade zu öffnen, da noch ein Prozess auf das Medium zugreift. Werden Sie dann mit **su** zu **root** und geben Sie den Befehl **lsuf /dev/hdc** ein. **hdc** ersetzen Sie evtl. durch die Gerätedatei, über die Ihr Linux das Laufwerk anspricht (z.B. **/dev/sr0**). Die Ausgabe des Befehls ähnelt der von **ps**: Auch hier finden Sie die Spalte PID, die Aufschluss darüber gibt, welche ID der Prozess hat, der auf das CD-Laufwerk zugreift. Handelt es sich nur um ein vergessenes Fenster (Dateimanager), reicht es aus, dieses zu schließen, damit Linux die CD freigibt. Der Griff zu **kill** ist nur dann nötig, wenn das Programm, welches das Laufwerk blockiert, abgestürzt ist. Mit einem beherzten **kill -9 PID** machen Sie dem Spuk ein Ende und sorgen dafür, das Linux den Datenträger herausrückt.

siehe auch: man killall, kill, skill, xkill, lsof, pkill

gpg

siehe auch: Verschlüsselung

Linux



Im März 1991 fing Linux Benedict Torvalds in Helsinki damit an, die Möglichkeiten des Intel-386-Prozessors in seinem neuen PC zu studieren. Er hatte das 386er Minix installiert und damit das C-Entwicklungssystem der Free Software Foundation. Nur ein halbes Jahr später war aus den Assemblerstudien ein kleines, lauffähiges Betriebssystem entstanden. Als Linus im September 1991 die erste Version (0.01) von Linux an

interessierte Minixer verschickte, musste es noch unter Minix übersetzt und installiert werden.

Linus Torvalds hat seine eigene Entwicklung von Anfang an frei angeboten. Jeder kann die Quelltexte bekommen und daran mitarbeiten.

Linux-Dateisystem

Dateisysteme sind die Schnittstellen zwischen dem Betriebssystem und den Partitionen auf Datenträgern. Sie organisieren die geordnete Ablage von Daten. Als Benutzer bekommt man von der Arbeit eines Dateisystems nicht viel mit. Man sieht nur, dass Dateien oder Verzeichnisse vorhanden sind, nicht aber wo und wie diese Daten auf dem Datenträger organisiert, gespeichert werden. Diese Arbeit übernimmt das Dateisystem. Es richtet eine Art Inhaltstabelle ein, in der alle Dateien (auch Verzeichnisse) und ihre Speicheradressen verwaltet werden. Neben einer solchen grundlegenden Datenorganisation stellen die verschiedenen Dateisysteme noch unterschiedliche zusätzliche Möglichkeiten zur Verfügung.

Was macht ein Dateisystem?

Neben der Datenorganisation auf dem Datenträger kann ein Dateisystem noch zusätzliche Möglichkeiten zur Verfügung stellen (Beispiele):

- Verzeichnisse und Unterverzeichnisse anlegen
- Datumsinformationen speichern (Erstellungsdatum, letzte Änderung, Zugriff)
- Lange Dateinamen verwenden
- Groß- und Kleinschreibung für Dateinamen berücksichtigen
- Sonderzeichen für Dateinamen ermöglichen (z.B.: Leerzeichen)
- Rechteverwaltung zur Zugriffssteuerung auf Dateien/Verzeichnisse
- Journaling-Funktionen
- und einiges mehr.

Warum braucht Linux ein spezielles Dateisystem?

Linux nutzte bis ca. 1992 das Dateisystem Minix, was aber aus unterschiedlichen Gründen (max. Partitionsgröße 64 MB, Dateinamen durften nur 14 Zeichen lang sein) nicht mehr den Ansprüchen genügte. Als Alternative bot sich das ext-Dateisystem an, welches (fast) allen Anforderungen gerecht wurde.

Linux-Dateisysteme: Ext2, Ext3, Ext4, BTRFS, ReiserFS, XFS, JFS ...

Moderne Linux-Dateisysteme müssen unter anderem folgende Eigenschaften besitzen:

- Rechteverwaltung: Linux wurde von vornherein als Mehrbenutzer-Serverbetriebssystem konzipiert. Damit nicht ein Nutzer auf die Daten eines anderen Benutzers zugreifen kann, ist eine Rechteverwaltung nötig, die solche Zugriffe unterbindet. Ebenso verhindert die Rechteverwaltung, dass ein Nutzer direkten Zugriff auf das Betriebssystem hat. Somit kann ein Nutzer nicht ungewollt die Betriebssystemeinstellungen „verbiegen“.
- symbolische Verknüpfungen: Dateisysteme, die diese Fähigkeiten nicht bieten, scheiden von vornherein aus.
- Linux ist ein offenes, freies Betriebssystem. Die verwendeten Komponenten sollen auch diesen Maßstäben entsprechen.

Bei der Erstellung eines Dateisystems ohne expliziter Angabe der Blockgröße, wird je nach Partitionsgröße automatisch eine passende Blockgröße gewählt. In der Regel ist die automatisch eingestellte Größe völlig ausreichend!

Hinweis: Dateisysteme, die diese Eigenschaften nicht besitzen (wie z.B.: FAT oder NTFS), sind für eine Installation von Ubuntu/Linux nicht geeignet!

NTFS besitzt zwar ein Rechte-System, dies funktioniert aber standardmäßig nur unter Windows. Erst durch den Dateisystem-Treiber NTFS-3G besteht optional die Möglichkeit, das Rechte-System von NTFS (mit gewissen Einschränkungen) auch in Linux zu verwenden.

Dateisystem	max. Dateigröße	max. Dateisystemgröße
Ext2 (Blockgröße 1 kByte)	16 GiB	2 TiB
Ext2 (Blockgröße 2 kByte)	256 GiB	8 TiB

Ext2 (Blockgröße 4 kByte)	2 TiB	16 TiB
Ext3 (Blockgröße 1 kByte)	16 GiB	2 TiB
Ext3 (Blockgröße 2 kByte)	256 GiB	8 TiB
Ext3 (Blockgröße 4 kByte)	16 GiB	2 TiB
Ext4 (Blockgröße 1 kByte)	16 GiB	2 TiB
Ext4 (Blockgröße 4 kByte; Standardwert)	16 TiB	1 EiB
BTRFS (B-Tree File System)	16 EiB	16 EiB
ReiserFS 3.6 (ab Kernelversion 2.4)	8 TiB	16 TiB
XFS	8 TiB	8 TiB
JFS (Blockgröße 512 Byte)	8 EiB	512 TiB
JFS (Blockgröße 4 kByte)	8 EiB	4 PiB

Hinweise:

Dateien auf 32-Bit-Systeme können nicht größer als 2 TByte (2^{41}) sein.

1 KiB (Kilo-Byte) = 1024 Byte = 2^{10} Byte

1 MiB (Mega-Byte) = $1024 * 1024$ Byte = 2^{20} Byte

1 GiB (Giga-Byte) = $1024 * 1024 * 1024$ Byte = 2^{30} Byte

1 TiB (Tera-Byte) = $1024 * 1024 * 1024 * 1024$ Byte = 2^{40} Byte

1 PiB (Peta-Byte) = $1024 * 1024 * 1024 * 1024 * 1024$ Byte = 2^{50} Byte

1 EiB (Exa-Byte) = $1024 * 1024 * 1024 * 1024 * 1024 * 1024$ Byte = 2^{60} Byte

Auf dem Windows-Dateisystem FAT32 oder VFAT ist die max.

Dateisystemgröße auf 2 TByte (Windows 2000: 32 GByte) und die max.

Dateigröße auf 4 GByte begrenzt. Bei dem Dateisystem NTFS (NTFS 5.0) ist die max. Dateisystemgröße auf 2 TByte (aktuelle NTFS-Dateisysteme:

16 EByte) begrenzt und die max. Dateigröße ist nur durch die Größe des Dateisystems (2 TByte bzw. 16 EByte) begrenzt.

Wird ein Dateisystem erzeugt z.B. mit **mkfs.ext4 -L "DISK 2" /dev/sdb1** (ohne die Angabe von Blockgröße, Anzahl der Inodes, etc.), so werden automatisch die Standardwerte verwendet. Die Blockgröße wird häufig nach der Größe der Festplatte gewählt (kleine Festplatte – kleine Blockgröße; große Festplatte – große Blockgröße). Die aktuell verwendete Blockgröße kann man mit dem Terminalprogramm **stat** ermitteln (z.B. **stat -f /dev/sda1**).

siehe auch: **df**, **fdisk**, man **mkfs.ext2**, man **mkfs.ext3**, man **mkfs.ext4**, **cat /etc/mke2fs.conf**, **stat**

Linux-Verzeichnishierarchie

Linux-Distributionen orientieren sich am so genannten Filesystem Hierarchy Standard, kurz FHS, der die Verzeichnisstruktur für Unix und Unix-ähnliche Systeme beschreibt. Das hat den Vorteil, dass Sie sich schnell auf jedem System zurechtfinden. Ein FHS-kompatibles Dateisystem hat folgende Verzeichnisse:

- /bin** ... Programme, die alle Benutzer starten können
- /boot** ... Konfiguration des Bootloaders
- /dev** ... Gerätedateien
- /etc** ... systemweite Konfigurationsdateien
- /home** ... Benutzerverzeichnisse
- /lib** ... Kernel-Module und Bibliotheken
- /media** ... Mount-Verzeichnis für auswechselbare Datenträger
- /mnt** ... dient als kurzzeitiger Einhängpunkt für Dateisysteme
- /opt** ... zusätzliche Programmpakete
- /root** ... Home-Verzeichnis des Administrators
- /sbin** ... Systembefehle, die nur root benutzt
- /srv** ... Daten von Servern, etwa vom Webserver Apache
- /tmp** ... temporäre Dateien
- /usr** ... System-Tools, Bibliotheken und Programme
- /var** ... variable Daten wie Log-Dateien, Mail-Spools et cetera.

Diese Verzeichnisse müssen nicht alle auf einem System vorkommen, einige sind optional. Die Verzeichnisse **/bin**, **/dev**, **/etc**, **/lib**, **/sbin** und – falls vorhanden – auch **/root** müssen unbedingt auf der root-Partition eines Linux-Rechners liegen, ohne sie kann das System nicht starten. Andere Verzeichnisse können noch während des Hochfahrens gemountet werden.

Unter Linux kommen noch ein paar spezielle Verzeichnisse hinzu:

/lib64 ... Bibliotheksverzeichnis der 64-Bit-Architekturen PPC64, s390x, sparc64 und AMD64; 32-Bit-Bibliotheken werden in /lib abgelegt. Die 64-Bit-Architektur IA64 bringt 64-Bit-Bibliotheken ebenfalls in /lib unter.

/proc ... Kernel- und Prozessinformationen; andere Unix-Systeme verwenden dazu beispielsweise /dev/kmem.

Am meisten ist root in den Verzeichnissen /etc und /var unterwegs. Im Verzeichnis /etc legt der Administrator die systemweite Konfiguration der Programme fest. Im /var-Verzeichnis findet er beispielsweise die Log-Dateien der verschiedenen Server-Anwendungen und Daemons.

siehe auch: proc-Dateisystem, www.pathname.com/fhs

last

last ... zeigt die letzten erfolgreichen Systemanmeldungen an

last <Benutzername> ... zeigt die letzten erfolgreichen Systemanmeldungen, des benannten Benutzers an

siehe auch: faillog

ldd

Das Programm ldd zeigt die von einem Programm verwendeten Bibliotheken an. ldd erwartet die Angabe eines vollständigen Dateipfades.

ldd [OPTION] FILE

Beispiele:

ldd /bin/bash ... zeigt die verwendeten Programm-Bibliotheken an

ldd -v /bin/bash ... zeigt alle verwendeten Bibliotheken an; auch die Bibliotheken, die von den direkt verwendeten Bibliotheken abhängig sind

siehe auch: man ldd

less

Anzeigen von Textdateien. Man kann mit der [Leertaste] - vorblättern und mit der Taste [b] - (b ... back) zurückblättern. Mit der Taste [h] kann ein Hilfetext angezeigt werden. Mit den Pfeiltasten und den Tasten [Bild hoch] und [Bild runter] kann ebenfalls navigiert werden.

less <MeineDatei>

Bei langen Texten springt man mit dem Kleiner-Zeichen (<) zum Datei-Anfang und mit dem Größer-Zeichen (>) zum Datei-Ende. Beendet wird less mit der Taste [q] - (q ... quit).

/"Suchtext" ... suchen des Wortes <Suchtext> in der Datei; mit [Enter] springt less zum ersten Treffer. Um zur nächsten Fundstelle zu wechseln, drücken Sie [n]. Möchten Sie rückwärts suchen so geben Sie statt des Schrägstrichs (/) ein Fragezeichen (?) ein. Mit [Umschalt] + [n] gelangen Sie jeweils zum nächsten Treffer einer Rückwärtssuche.

zless <Dateiname> ... Bei mit gzip komprimierte Dateien können Sie mit **zless** einen Blick hineinwerfen, ohne die Datei erst entpacken zu müssen.

less <Datei_1> <Datei_2> ... zeigt zuerst die Datei_1 an; mit **:n** zu Datei_2 wechseln und mit **:p** wieder zu Datei_1 wechseln

less -e <Datei_1> <Datei_2> ... zeigt beide Dateien an ohne die Angabe von :n; less wird automatisch beendet, wenn die letzte Zeile von Datei_2 erreicht ist

siehe auch: man less, man zless, man zmore, man zcat

ln

Erzeugt einen symbolischen Link - in der Windows-Welt als Verknüpfung bekannt. Das Erstellen eines symbolischen Links erzeugt eine neue Datei, die einen eigenen Inode besitzt (Inode: Den Inhalt einer Datei speichert das Betriebssystem in Blöcken auf der Festplatte. Auf diese Blöcke verweisen Inodes, die Zusatzinformationen wie Dateiberechtigungen, Dateibesitzer und Zeitstempel der Datei enthalten.). An welcher Stelle sich die Zieldatei im Dateisystem befindet, spielt keine Rolle.

ln -s <Ziel> <Dateiname>

Zwei Eigenschaften im gelisteten Dateinamen (»ls -l«) kennzeichnen einen symbolischen Link: das kleine »l« als erster Buchstabe vor den Dateiberechtigungen sowie der Verweis auf die entsprechende Originaldatei:

lrw-r--r-- 1 root root [...] datei -> originaldatei

Wird die Datei, auf die der symbolische Link verweist, gelöscht zeigt dieser ins Leere. Umgekehrt wirkt sich das Entfernen des Links in keiner Weise auf die Zieldatei aus.

In <Dateiname1> <Dateiname2>

<Dateiname2> ist eine »Kopie« von <Dateiname1>, mit der Besonderheit - wird eine Datei geändert, ändert sich nach dem Speichern auch automatisch der Inhalt der anderen Datei (ein so genannter hard-link). Genau genommen ist dies nicht richtig, aber für die Benutzer erscheint dies so.

Hardlinks dürfen nur auf Dateien, nicht jedoch auf Verzeichnisse zeigen.

Ein Hardlink erstellt einen zusätzlichen Namen, der auf den vorhandenen Inode der Originaldatei verweist. Daher behalten diese auch nach dem Verschieben der Zieldatei (<Dateiname1>) innerhalb einer Partition ihre Gültigkeit. Da Dateisysteme Inodes pro Partition separat verwalten, dürfen Hardlinks nur auf Dateien innerhalb derselben Partition zeigen. Der Befehl (»ls -li«) listet den Verzeichnisinhalt in Langform inklusive der Inode-Nummer jeder Datei auf. Dateien mit identischen Inode-Nummern verweisen deshalb immer auf dieselben Speicherblöcke.

```
42616 -rw-r--r-- 3 root root [...] datei_a  
42616 -rw-r--r-- 3 root root [...] datei_b
```

Wie viele Namen eine Datei besitzt, zeigt die Zahl hinter den Dateiberechtigungen - der ebenfalls in den Inodes gespeicherte Link-Counter. Bei einer Datei steht er standardmäßig auf 1, jeder zusätzliche Name (Hard-Link) erhöht diese Zahl um eins. Im Beispiel zeigen drei Namen auf den gleichen Inode, also dieselbe Datei. Bei einem Verzeichnis steht der Link-Counter immer mindestens auf 2, da ein Verzeichnis auch über die Abkürzung ».« (sie steht für das lokale Verzeichnis, in dem sich der Benutzer gerade befindet) zu erreichen ist. Jedes Unterverzeichnis erhöht den Counter ebenfalls um einen Zähler-Wert, da es den Namen »..« enthält, der das übergeordnete Verzeichnis anspricht. Das Löschen eines Hardlinks entfernt lediglich den Namen einer Datei. Erst das Löschen des letzten Namens entfernt auch deren Inhalt beziehungsweise markiert den Inode als frei und damit als beschreibbar. Es gibt keine Originaldatei und auch keinen Verweis wie bei symbolischen Links, sondern nur mehrere gleichwertige Namen.

siehe auch: man ln, Zugriffsrechte

locate

Mit **locate** <Muster> kann man herausfinden, in welchem Verzeichnis sich eine spezifizierte Datei befindet. Zusätzlich können dabei auch Jokerzeichen (*, ?, * ... steht für beliebig viele Zeichen; ? ... steht für genau ein Zeichen)

verwendet werden.

Das Programm arbeitet sehr schnell, da es nicht das gesamte System durchsucht sondern eine Datenbank. Dies ist aber auch das **Hauptproblem**, da in dieser Datenbank nur Dateien gelistet sind, die nach der letzten Aktualisierung durch **updatedb** erstellt oder aktualisiert wurden. Der updatedb-Prozess wird entweder täglich in der Nacht oder ca. 15 Minuten nach dem Einschalten automatisch gestartet.

updatedb sollte mit dem **&**-Zeichen aufgerufen werden - mit **updatedb &** arbeitet das Programm im Hintergrund. Damit alle Dateien erfasst werden, sollte **updatedb** als root aufgerufen werden.

locate xinetd.conf ... sucht die Datei xinetd.conf

locate '*extract*' ... sucht und findet alle Dateien mit dem Namensbestandteil »extract«

Hinweis: Das Programm »locate« muss evtl. erst installiert werden (Paketnamen: locate, mlocate oder findutils).

Während des laufenden updatedb-Prozesses verlangsamten sich alle anderen Prozesse merklich. Dies bedeutet im Einzelfall, dass manche Programme einige Minuten - z.B. ein Textverarbeitungsprogramm - kaum noch nutzbar sind.

Wird eine tägliche Aktualisierung nicht benötigt, so kann das Skript **/etc/cron.daily/updatedb** z.B. nach **/etc/cron.monthly/updatedb** verschoben werden (vorher als root anmelden).

Nachtrag von SuSE: Der Befehl **updatedb** sollte nicht direkt über **/usr/bin/updatedb** aufgerufen werden. Bitte verwenden Sie stattdessen den Befehl **/etc/cron.daily/updatedb**.

ls

Liste von Dateien und Verzeichnissen anzeigen.

ls [OPTION] [VERZEICHNIS bzw. DATEI]

Beispiele:

ls -l ... Verzeichnisinhalt des aktuellen Verzeichnisses auflisten (aber ohne **.***-Konfigurationsdateien)

ls -la ... zeigt alle Dateien (auch **.***-Konfigurationsdateien).

ls -la | less ... zeigt Dateien seitenweise (mit Leertaste weiter, mit q beenden).

ls -ltr ... zeigt neueste Dateien zu unterst.

ls -rtl ... zeigt eine ausführliche Dateiliste mit Datum an - Parameter l - sortiert nach dem Änderungsdatum - Parameter t - wobei die zuletzt geänderten Dateien am Schluss angezeigt werden - Parameter r, für reverse.

ls -lh ... zeigt die Bytes in für Menschen lesbarer Form an (h = human readable).

ls * > ./inhalt.txt ... alles wird in die Datei inhalt.txt umgeleitet

ls -lh > lpr ... Umleitung an den Standard-Drucker

ls / -l ... Auflistung des Inhaltes vom Wurzelverzeichnis

ls -l /bin/m* ... findet alle Dateien im Verzeichnis /bin die mit dem Buchstaben m beginnen - der Stern ist ein so genannter Joker oder Wildcard

ls *.j{p,pe}g | wc -w ... alle JPEG-Dateien in einem Verzeichnis zählen

* ... beliebige Zeichenkette

? ... genau ein beliebiges Zeichen

[abc] ... genau eines der genannten Zeichen

[^abc] ... genau ein nicht genanntes Zeichen oder [!]

{abc,bcdf} ... genau eine der Zeichenketten

ls | wc -w ... Alle Einträge eines Verzeichnisses - Dateien und Verzeichnisse - zählen.

ls -R | wc -w ... Alle Einträge eines Verzeichnisses und seiner Unterverzeichnisse - Dateien und Verzeichnisse - zählen.

Change- Zugriffs- und Modifikationszeit ermitteln:

ls -l --time=ctime ... zeigt an wann der Status der Datei (change time) geändert wurde, statt der Modifikationszeit; Eine create time (Erstellungsdatum) kennt Linux nicht, d.h. jede Änderung des Dateistatus z.B. über **chmod** führt auch zu einer Änderung der **change time**.

ls -l --time=access ... zeigt das Datum und Zeit des letzten Zugriffs an, statt der Modifikationszeit

ls -l ... zeigt standardmäßig das Datum und Zeit der letzten Änderung an (Modifikationszeit)

ls | tr [:upper:] [:lower:] | grep -oP '[^\\.]+' | sort | uniq -c | sort ... listet die Anzahl der einzelnen Dateitypen (txt, odt, jpg ...) auf und gibt am Schluss die Gesamtsumme aller Dateien (ohne Verzeichnisse) des aktuellen Verzeichnisses (ohne Unterverzeichnisse) aus

siehe auch: wc, Umleitung von Befehle, Anhang: Skript-Listings => cronjobs, Einführung in die Shellprogrammierung, man stat

lsusb

Auflistung aller vom System erkannten USB-Geräte.

```
karl@tux:~$ lsusb
Bus 007 Device 002: ID 0458:003a KYE Systems Corp. (Mouse
Systems)
Bus 002 Device 003: ID 0bda:0158 Realtek Semiconductor
Corp. Mass Storage Device
```

ID 0458:003a ... die ersten 4 Ziffern vor dem Doppelpunkt stehen für den Hersteller und die 4 Ziffern nach dem Doppelpunkt kennzeichnen den Gerätenamen; über die Webseite <http://www.linux-usb.org/usb-ids.html> können diese Zeichenketten entschlüsselt werden

siehe auch: man lsusb, Links (Hardware)

lsdvd

Ein Programm zum Lesen des Inhalts einer DVD und Ausgeben des Inhalts im Terminal oder in maschinenlesbaren Formaten (Perl, Python etc.).

```
lsdvd [ options ] [-t track_number] [dvd path]
```

lsdvd ... kurze Übersicht über die eingelegte DVD

lsdvd -x ... alle Informationen im Terminal anzeigen

lsdvd -a ... Informationen über die Audiostreams

lsdvd -v ... Informationen über die Videostreams

siehe auch: man lsdvd, lsdvd -h

lshw

Auflistung der vom System erkannten Hardware (CPU, Hauptspeicher, Festplatten etc.).

lshw ... ausführliche Liste mit der erkannten Hardware

sudo lshw ... mit root-Rechten aufgerufen, liefert lshw eine wesentlich ausführlichere Liste

lshw -short ... kurze tabellarische List mit der erkannten Hardware

Alternative Programme mit grafischer Oberfläche zur Ermittlung der Hardware-Komponenten sind **hardinfo** (Programmaufruf: Terminal öffnen und **hardinfo** eingeben) und **i-nex** (<https://launchpad.net/i-nex>).

siehe auch: man lshw, man hwddata, man discover

lsf

lsf (list open files) liefert Informationen über geöffnete Dateien. Unter Linux werden auch Blockgeräte (Laufwerke), Netzwerkports usw. über Dateien angesprochen.

Ebenso kann lsf sehr gute Dienste leisten, wenn man die Vermutung hat, dass auf dem eigenen Rechner (Server) Dienste laufen, die nicht laufen sollen bzw. die man gar nicht selber aktiviert hat.

Dies bedeutet lsf kann ein sehr mächtiges Tool zur Überwachung und Analyse eines Systems sein.

lsf [Option] <Format>

Formate:

/Pfad/Datei ... Informationen über die Verwendung der angegebenen Datei

/dev/cdrom ... Informationen über die Verwendung von z.B. dem CD-ROM-Laufwerk

+D /home/benutzername ... Informationen über die Verwendung von Dateien z.B. des Home-Verzeichnisses des Anwenders

+p PID ... liefert die Informationen welche PID welche Datei benutzt

-c Prozess-Name ... liefert die Informationen welcher Prozess welche Datei benutzt

-u Anwender-Name ... liefert die Informationen welcher Anwender welche Datei benutzt

-i TCP/UDP oder IP-Adresse oder Port-Nr. ... liefert die Informationen welcher Netzwerkdienst von welchem Anwender, PID usw. benutzt wird

+L1 ... liefert die Informationen über bereits gelöschte Dateien

-a ... logische UND-Verknüpfung von Optionen; dies macht immer dann Sinn, wenn lsf mit zwei Optionen aufgerufen wird und beide angewendet werden sollen, da per Voreinstellung immer eine ODER-Verknüpfung angewendet wird

Optionen:

-r Sekunden ... wiederholt die Ausgabe alle x Sekunden

-n ... gibt IP-Adressen statt Hostnamen aus

-l ... gibt die Benutzer-ID statt des Benutzernamens aus

-P ... gibt Port-Nummern statt Service-Namen aus.

-t ... gibt nur eine PID Liste aus

-F ... gibt alle Ergebnisse in einer einzigen Spalte aus

Beispiele:

lsof -i ... offene Ports und bestehende Verbindungen werden angezeigt

lsof +L1 ... offene gelöschte Dateien, die sich nur noch im Hauptspeicher befinden werden angezeigt

lsof -i | grep '>->' ... zeigt die wirklich bestehenden

Netzwerkverbindungen an - falls Ihnen ein Prozess unbekannt oder sonst wie verdächtig vorkommt (evtl. ein Hacker), können Sie ihn mit **kill -KILL <PID>** kurzerhand aus dem System werfen (lsof liefert auch die PID)

lsof /dev/sdb1 ... Terminalausgabe, welche Prozesse gerade auf das Gerät /dev/sdb1 (USB-Stick) zugreifen

lsof /bin/bash ... Terminalausgabe, welche Prozesse gerade die Bash (genau genommen die Datei /bin/bash) nutzen

sudo lsof -u root ... es werden alle Dateien angezeigt, die vom Benutzer root offen gehalten werden

sudo lsof -u ^root ... es werden alle Dateien angezeigt, die nicht vom Benutzer root (sondern von anderen Benutzern) offen gehalten werden

sudo lsof -a -i -u www-data ... es werden alle Netzwerkports (Netzwerkdateien) angezeigt, die vom Benutzer www-data offen gehalten werden

LSOF - Nachrichten aus dem Nirwana

Dateien können unter Unix/Linux gelöscht werden, obwohl sie von einem Prozess noch geöffnet sind. Ein Effekt davon ist, dass der Prozess, der die Datei noch verwendet, ohne Probleme weiter in die Datei schreiben und Daten aus ihr lesen kann.

Dieses Phänomen machen sich Hacker-Programme häufig zu Nutze, wenn sie Daten zwischenspeichern müssen.

Hier eine kleine Demonstration dieses Prinzips: Wenn Sie in die Datei ~/sniffer durch das Kommando

```
cat > ~/sniffer
```

alle Eingaben von der Standardeingabe (normalerweise die Tastatur) hineinschreiben, wird eine Datei zum Schreiben geöffnet. Setzen Sie nun diesen Prozess durch die Tastenkombination [Strg] + [Z] in den Hintergrund. Anschließend löschen Sie die Datei durch

```
rm -f ~/sniffer
```

Wie Sie anhand von

ls ~/sniffer

sehen, existiert die Datei nicht mehr. Holen Sie jetzt den cat-Prozess durch

fg

wieder in den Vordergrund. Tippen Sie ein paar Zeichen, und Sie sehen, dass keine Fehlermeldung kommt. Die Datei ist ja noch offen und somit beschreibbar. Setzen Sie den Prozess wieder in den Hintergrund und schauen Sie mit ls nochmals nach, ob die Datei ~/sniffer existiert...

Ergebnis: nein. Durch den Befehl

ls -l

erhalten Sie eine Liste der offenen Dateien, deren Link-Count kleiner 1 ist - die also gelöscht sind. Sie werden in der Ausgabe eine ähnliche Zeile wie diese finden:

```
cat      2415   root   lw      REG    3,7    8      0      160497 /
root/sniffer      (deleted)
```

Die Datei ist also noch offen und wird vom cat-Prozess weiter benutzt. Wenn Sie jetzt noch ein paar Mal mit dem cat-Prozess spielen und weitere Zeichen eingeben und ls -l danach aufrufen, werden Sie auch feststellen, dass sich die Größe der Datei (zweiter Wert nach REG) ändert. Die Daten werden also wirklich noch in der Datei gespeichert.

Wenn Sie sich den cat-Prozess durch fg wieder in den Vordergrund holen und [Strg] + [d] drücken, bereiten Sie den Spuk ein Ende.

siehe auch: man ls -l, fuser, kill, Anhang: Skript-Listings → Cronjobs einrichten

lpr

lpr unterstützt das Drucken von der Kommandozeile.

lpr <MeineDatei> ... Datei auf den Standarddrucker ausdrucken.

lpr -P <MeinDrucker> <MeineDatei> ... Datei ausdrucken.

lpq -P <MeinDrucker> ... Mit lpq können die aktiven Druckaufträge, einschließlich der Jobnummer ermittelt werden.

lprm -P <MeinDrucker> <Jobnummer> ... Löscht den angegebenen Druckauftrag, sofern der Druckauftrag dem aktuellen Benutzer gehört.

lprm -Pall all ... Mit diesem Befehl werden alle Druckaufträge gelöscht, die der aktuelle Benutzer löschen darf. Nur Benutzer root-Rechten dürfen alle Druckaufträge löschen.

ls -l | lpr -P <MeinDrucker> ... Die Ausgabe des Befehls **ls -l** an den Drucker umleiten.

ls -l | lpr ... Die Ausgabe des Befehls **ls -l** an den Standarddrucker umleiten.

siehe auch: man **lpr**, Turboprint

Lynx

Lynx ist ein textbasierter Browser, der an der Kommandozeile gestartet wird. Nun stellt sich vielleicht die Frage, wofür ein textbasierter Browser gut sein kann, es gibt doch unzählige Browser mit grafischer Oberfläche?

Textbasierte Browser können z.B. für automatisierte HTTP-Anfragen oder PHP- oder CGI-Programmaufrufe auf entfernte Server genutzt werden. Die HTTP-Anfragen oder Programmaufrufe können über zeitgesteuerte Shellskripts aufgerufen werden (Cronjobs).

lynx -dump 'URL?para1=wert1¶2=wert2' ... ruft die angegebene URL auf und übergibt per GET zwei Variablen an die Webseite

lynx -dump 'URL?para1=wert1¶2=wert2' | grep 'Suchstring' ... Zeigt alle Zeilen an, in denen der gesuchte String enthalten ist.

Weitere Optionen:

-accept_all_cookies

Mit dieser Option erlaubt lynx die Annahme von Cookies ohne weitere Nachfragen.

-cookie_file=<dateiname>

Wenn der Wiedererkennungswert eines Cookies gefragt ist, d.h. z.B. der Webshop anhand von Cookies die Identität feststellt, so hilft diese Option weiter.

-auth=<loginname>:<passwort>

Oft ist die Eingabe eines Passwortes notwendig, mit vorgenannter Option kann das Login automatisiert werden.

export http_proxy=http://www-proxy.btx.dtag.de:80/

export ftp_proxy=ftp-proxy.btx.dtag.de:80

Zur Ermittlung der richtigen Proxy-Einstellungen greift lynx auf die Shellvariablen »http_proxy« und »ftp_proxy« zurück.

lynx -useragent='mozilla/5.0(windows; u; windows nt 5.0; en-us; rv:1.6) gecko/20030916'

Webseiten können so konzipiert sein, dass je nach Browser unterschiedliche Webseiten erscheinen. Mit der vorgenannten Anweisung gibt sich lynx als Mozilla 1.6 aus, der unter Windows 2000 gestartet wurde.

-cmd_log=<dateiname>

Diese Option schaltet die Tastenprotokollierungs-Funktion von lynx ein, zeichnet alle Tastenanschläge auf und speichert sie in einer Datei.

-cmd_script=<dateiname>

Mit der Option -cmd_script werden die Tasten wieder abgespielt, als würden sie gerade so in diesem Moment eingegeben. Auch diese Funktion ist erstaunlicherweise auch via cron zeitgesteuert aufrufbar.

lynx -source -parsed http://www.google.de | less

Zeigt den Quellcode (-source) der angegebenen HTML-Seite im Programm less an. Mit der Kombination der beiden Optionen (-source -parsed) werden auch die Zeilenumbrüche korrekt dargestellt.

lynx -post_data http://www.pohlsearch.de

Um Informationen im POST-Teil des Headers mitzuschicken, muss lynx mit der Option -post_data aufgerufen werden.

Konvertierung:

Konvertierung einer HTML-Seite in normalen Text.

lynx -dump <Dateiname.html> > <Dateiname.txt> bzw.

lynx -dump <Dateiname.html> | less

siehe auch: W3M

M

make

Mit make kann man Quelltexte kompilieren, vorausgesetzt make und ein C-Compiler sind installiert.

`./configure ...` Überprüfung starten

`make ...` Kompilierung

`make install ...` Installation

`make clean ...` räumt auf oder `make mrproper`

siehe auch: man make

mkpasswd bzw. makepasswd

mkpasswd bzw. makepasswd erstellt mit Hilfe der Bibliothek crypt ein verschlüsseltes Passwort.

Der Name des Programms variiert in den verschiedenen Linux-Distributionen. mkpasswd wird häufig in Shellskripten verwendet (z.B. automatische Erstellung von Benutzern mittels useradd).

`mkpasswd PASSWORD [SALT]`

mkpasswd --method=help ... gibt die verfügbaren Hash-Algorithmen aus (des, md5, sha-256, sha-512); alternative Methode: `mkpasswd -H help`

Beispiel: Benutzer anlegen

Benutzername: ben

Passwort: eifelturm

sudo useradd -G users,adm,lp,dialout,cdrom,plugdev,sambashare -d /home/ben -c 'Benjamin Bardot' -s /bin/bash -m -p "\$(mkpasswd eifelturm --method=sha-512)" ben

mkpasswd 'eifelturm' --salt='XYZABCJKLM' --method=sha-512 ... erzeugt ein »gesalzenes« Passwort; der SALT-String - mindestens 8 Zeichen - maximal 16 Zeichen; ein mehr oder weniger zufälligen SALT-String kann man mit **pwgen -n 12** oder mit **mkpasswd "\$(date)" -m** des erzeugen

siehe auch: man mkpasswd bzw. man makepasswd, useradd, passwd, chpasswd, id

mail

Unter bestimmten Voraussetzungen senden die Pseudo- oder Systembenutzer (root, news, wwwrun, nobody, games, postfix etc.) Mails an den Systemadministrator des Systems. Auf der Shell erhalten Sie dann immer die Meldung, dass eine Mail vorliegt.

Mit Eingabe von »**mail**« wird das Programm gestartet, anschließend kann über die Eingabe von [t] die letzte eingegangene Nachricht angezeigt werden.

Das Terminalprogramm mail ist Bestandteil des Paketes bsd-mailx bzw. mailutils.

Befehlsmodus:

? ... zeigt eine Liste aller Kommandos

help ... zeigt eine Liste aller Kommandos

t * ... zeigt alle Mails an; bei mehreren oder längeren Mails mit Leertaste vorwärts blättern und mit q den Betrachter verlassen; dies gilt auch für die nachfolgenden Befehle

n 2 ... zeigt die neue Mail Nr. 2

o 4 ... zeigt die alte Mail Nr. 4

u 1 ... zeigt die unlesene Mail Nr. 1

d ... löscht die letzte Mail

d * ... löscht alle Mails

d 20 ... löscht die Mail Nr. 20

r ... Mail beantworten, sie wird an alle Empfänger und dem Autor gesendet

R ... Mail beantworten, sie wird nur an den Autor der Mail gesendet

write <Message-Nr.> <Dateiname> ... speichern der Email im aktuellen Verzeichnis als Textdatei; Beispiel: write 2 email.txt

q ... beendet das Mailprogramm

quit ... beendet das Mailprogramm

mail <Benutzer> ... schickt eine Mail an einen lokalen Benutzer

Zuerst den Betreff, Subject angeben und mit der Entertaste zur Eingabe des Haupttextes wechseln. Jede Zeile wird mit der Entertaste abgeschlossen. Die Mail wird abgeschickt, wenn in einer neuen - der letzten - Zeile ein Punkt (.) eingegeben wird.

mail <Benutzer>@<Rechnername>.<Domain> ... schickt eine Mail an einen lokalen Benutzer, als Adresse wird hier der vollwertige Domainname angegeben; den eigenen vollwertigen Domainnamen ermitteln Sie mit **hostname --long**.

mail -u <Benutzer> ... die Mails des benannten Benutzers anzeigen; dieser Aufforderung wird nur bei ausreichenden Rechten nachgekommen; Benutzer root darf hier wieder alles

Beispiele:

Kurze Email versenden. Im Betreff sollten keine Leerzeichen vorkommen - Bindestrich (-) oder Unterstrich (_) verwenden.

```
echo "Hier steht der eigentliche Email-Text" | mail -s Betreff -r  
absender@domain.de empfaenger@domain.de
```

Wie vorher, aber mit zusätzlichen Email-Anhang.

```
echo -e "Hier steht der eigentliche Email-Text" | mail -s Betreff -r  
absender@domain.de -a email_anhang.tar.gz empfaenger@domain.de
```

Email versenden, dabei steht der Haupttext in der Datei email.txt.

```
mail -s Betreff -r absender@domain.de empfaenger@domain.de <  
email.txt
```

Hinweis: Wirklich nützlich sind die vorgenannten Beispiele nur in Shellskripts.

Ab und zu speichert das Programm mail die Mails samt der Anhänge im Home-Verzeichnis im mbox-Format. Der Name dieser Sammeldatei von mehreren Emails ist i.d.R. mbox.

Um diese Emails zu lesen, gehen Sie wie folgt vor.

cd ~ ... bewegen Sie sich mit **cd ~** in Ihr Home-Verzeichnis

mail -f mbox ... die Datei mbox mit mail öffnen

t * ... alle Emails lesen; die Eingabe erfolgt am Mail-Prompt;

write <Message-Nr.> <Dateiname> ... speichern der Email im Home-Verzeichnis als Textdatei; Beispiel: write 2 email.txt; hat die Email Nr.2 noch ein Email-Anhang (Attachment) so müssen Sie den Dateinamen noch mit [Enter] bestätigen; die Dateien finden Sie nun im Home-Verzeichnis

Die Mails werden unter **/var/spool/mail** im **mbox-Format** gespeichert. Die Mails können aber auch z.B. von einem Email-Programm mit grafischer Oberfläche (z.B. Thunderbird, Evolution, KMail) importiert werden - Menü »Extra« »Nachricht importieren ...«, beachte die Mails liegen im mbox-Format vor.

Sehr praktisch sind Here-Dokumente: Ein Skript darf mehrzeilige Eingaben direkt im Text enthalten und kann sie per Pipe an ein Kommando weiterleiten. Die Shell liest den Here-Dokumentenblock bis zum definierten End-Codewort und übergibt ihn als Standardeingabe an das Programm. Nachfolgende Zeilen können so in ein Terminal eingegeben werden - eine

neue Zeile wird mittels der »Enter-Taste« eingefügt.

cat <<EOM | mail root@tux.site

**Ich bin ein Here-Dokument,
EOM steht für end-of-mail, EOT für
end-of-text. Aber jedes andere
Wort wäre auch ok.**

EOM

Anmerkung: Die Bedienung des Programms **mail** ist ein wenig gewöhnungsbedürftig, aber seine Stärken spielt es in Shellskripts aus (**siehe auch:** Anhang: Skript-Listings → Backup mit tar).

Falls man aus einem Modus nicht mehr herauskommt, so kann man es mit der Tastenkombination [Strg] + [D] versuchen. In sehr hartnäckigen Fällen hilft ein oder mehrmaliges betätigen der Tastenkombination [Strg] + [C].

siehe auch: Anhang: Einführung in die Shellprogrammierung

man

Nachdem --help eine eher spärliche Ausbeute brachte, ist es nun an der Zeit man (engl.: manual ... Handbuch) kennen zu lernen. Dieses Handbuch bietet detaillierte Informationen über die auf dem System installierten Programme.

man <Programmname>

Dabei bringt jedes (Programm-)Paket seine eigenen Handbuchseiten mit. Bei der Installation wird es zum bereits bestehenden Teil hinzugefügt. Da wir es bei der Anzeige wieder mit less zu tun haben, kann man sich in den Seiten über die Pfeiltasten (↑, ↓) bewegen und durch einen Druck auf die Taste [Q] zur Shell zurückkehren.

Eine Suche lässt sich über **/"Suchtext"** starten. Um sich für den gleichen Begriff weitere Treffer anzeigen zu lassen, reicht ein Druck auf die Taste [N]. Die Fundstellen im Text sind jeweils hervorgehoben.

Wer mit diesen elementaren Funktionen nicht zufrieden ist, dem kann ein Hilfetext zu less, der über Taste [H] zu erreichen ist, auf die Sprünge helfen.

man -k [Schlüsselwort] ... Ausgabe aller verfügbaren Manuals in denen das Schlüssel (z.B. hw) vorkommt

Hinweis: Nicht jeder mag längere Texte am Monitor lesen. Besitzt man ein funktionierendes Drucksystem, so kann man die Manpage mit Hilfe des Kommandos **man -t <Programmname> | lpr** zu Papier bringen.

Übrigens: Bisher war immer von Programmen die Rede. Darüber hinaus existieren aber auch noch Manpages für bestimmte Konfigurationsdateien. z.B. liefert: **man fstab** Informationen zum Aufbau von /etc/fstab.

siehe auch: man man

mc

Midnight Commander ist ein Dateimanager für die Linux-Shell, der sehr an den Norton Commander erinnert. Der Midnight Commander wird mit den **Pfeiltasten** und der **Tab-Taste**, sowie mit den **Funktionstasten** gesteuert. Auf der grafischen Oberfläche funktioniert dies auch mit der Maus.

Das **ausgeklappte Menü** wird mit einem zweimaligen Druck auf die [Esc]-Taste wieder ausgeblendet.

Falls Sie die **Baumansicht** aktiviert haben, so können Sie über die Taste [F3] die angezeigten Unterverzeichnisse wieder schließen.

Der Midnight Commander speichert einmal eingebende Befehle in einer Liste. Die Liste wird mit der Tastenfolge [Esc] und [H] aufgerufen.

Um die History des Datei-Managers zu durchstöbern, drücken Sie [Alt] + [P] zum zurückblättern und [Alt] + [N] zum vorwärts blättern.

Mit [Strg] + [X], [T] fügen Sie den Namen der gerade markierten Datei an der aktuellen Position der Kommandozeile des Midnight Commander ein und mit [Strg] + [X], [P] fügen Sie stattdessen den Pfad des aktuellen Verzeichnisses in die Kommandozeile ein - erspart viel Tipparbeit.

Mit [Einfg] können Sie mehrere Dateien bzw. Verzeichnisse markieren, die z.B. gemeinsam in ein anderes Verzeichnis kopiert werden können.

Beispiel:

kwrite an der Kommandozeile eingeben und anschließend mit [Strg] + [X], [t] die aktuell markierte Datei nach kwrite einfügen. Danach einfach mit [Enter] bestätigen und die markierte Datei wird daraufhin mit Kwrite geöffnet.

Um in einen umfangreichen Verzeichnis nach einer bestimmten Datei zu

suchen ist die Tastenkombination [Strg] + [S] einzugeben und danach den Anfangsbuchstaben der gesuchten Datei, evtl. sind noch weitere Buchstaben einzugeben.

Haben Sie den Midnight Commander geöffnet und wollen nur kurz etwas in der Shell erledigen, müssen Sie den Datei-Manager weder beenden noch einen neuen Reiter in der KDE-Konsole öffnen. Mit **[Strg] + [O]** öffnet mc eine neue Shell im Terminal-Fenster, in der Sie alle Befehle eingeben können. Ein erneuter Druck auf **[Strg] + [O]** bringt Sie wieder zurück zum Midnight Commander.

Hinweis: Falls das Schließen von mc über die Taste **[F10]** nicht funktioniert, so kann man es über die Tastenkombination **[Esc] + [O]** versuchen.

mcedit <Dateiname> ... öffnet die angegebene Datei sofort im Editor des Midnight Commander, Schreibmodus

mc -v <Dateiname> ... öffnet die angegebene Datei sofort im Dateibetrachter des Midnight Commander, Lesemodus

md5deep

Ausgabe von MD5-Prüfsummen in eine Textdatei. Die Dateien in den Unterverzeichnissen werden mit einbezogen.

md5deep [OPTION]... [FILE]...

md5deep -rbtz \$HOME/Dokumente > \$HOME/md5.txt

r ... recursive, Dateien in den Unterverzeichnisse mit einbeziehen

b ... nur die Dateinamen anzeigen, kein Dateipfad

t ... GMT-Zeit der letzten Änderung anzeigen; nicht die Zeit der aktuell eingestellten Zeitzone

z ... Dateigröße in Byte; Anzeige vor der MD5-Prüfsumme

siehe auch: md5sum

md5sum

Die MD5-Prüfsumme (128 Bits) kann für die Integritätsprüfung von Dateien und Verzeichnisse genutzt werden.

md5sum [OPTION] [DATEI] [DATEI] ...

md5sum [OPTION] --check [DATEI]

-b, --binary ... Dateien im Binärmodus lesen (Vorgabe unter

DOS/Windows)

-c, --check ... MD5-Summen gegenüber einer angegebenen Liste prüfen

-t, --text ... Dateien im Textmodus lesen (Vorgabe)

-w, --warn ... bei ungeeignet formatierten Prüfsummenzeilen warnen

Beispiele:

pv beispiel.iso | md5sum ... MD5-Prüfsumme für eine ISO-Datei erzeugen, einschließlich einer Fortschrittsanzeige mittels des Programms pv

Überprüfung eines Verzeichnisses:

find /home/andi/projekt_a -type f | md5sum > ./md5 ... MD5-Prüfsumme für eine spätere Überprüfung der Integrität des angegebenen Verzeichnisses ermitteln;

Die MD5-Prüfsumme verändert sich erst, wenn Dateien entfernt oder hinzugefügt werden. Die Prüfsumme wird im Beispiel in der Datei md5 im aktuellen Verzeichnis gespeichert.

Ausgangspunkt für die Ermittlung der MD5-Prüfsumme ist im Beispiel das Verzeichnis /home/andi/projekt_a, dabei werden nur gewöhnliche Dateien berücksichtigt (-type f). Unterverzeichnisse werden ebenfalls berücksichtigt.

find /home/andi/projekt_a -type f | md5sum --warn --check ./md5 ... Integrität des angegebenen Verzeichnisses überprüfen;

Achtung: Änderungen an den Zugriffsrechten oder Änderungen innerhalb der einzelnen Dateien werden bei diesem Beispiel NICHT erfasst. Das Löschen bzw. das Einfügen von leeren Verzeichnissen wird ebenfalls nicht erfasst (-type f).

Überprüfung von Dateien:

for i in \$(find /home/andi/projekt_a -type f -print);do md5sum "\$i"
>> ./md5;done ... MD5-Prüfsummen für eine spätere Überprüfung der Integrität von Dateien, einschließlich der Dateien in den Unterverzeichnissen, ermitteln;

md5sum --warn --check ./md5 ... Integrität der Dateien überprüfen, die in der MD5-Datenbank md5 aufgelistet sind;

Folgenden Änderungen werden nicht erfasst:

- Dateien bzw. Verzeichnisse werden hinzugefügt (siehe auch weiter oben: Überprüfung eines Verzeichnisses)
- Änderungen an den Zugriffsrechten
- Löschen von leeren Verzeichnissen (-type f)

Aufspüren modifizierter Dateien

Auf Anhieb fallen einem die zu jeder Datei gespeicherten Modifikationszeiten ein. Doch leider sind diese ebenso vertuschbar wie die Dateigröße. Der einfache Ansatz, das Dateisystem nach Dateien mit diesen Parametern zu durchsuchen, stellt die Integrität keinesfalls sicher (zumindest falls ein Profi-Cracker am Werk war).

Neben Werkzeuge wie Tripwire, Hobgoblin, sXid u.a., die leider nur zum Teil aktuellen Distributionen beiliegen, kann hier eine Bestandsaufnahme des Systems Abhilfe schaffen. Da die MD5-Verschlüsselung unterdessen allgemein verfügbar ist, kann zu jeder Datei ein MD5-Fingerabdruck erzeugt werden. Speichert der Administrator Dateiname und Fingerabdruck in einer separaten Datei (außerhalb des Dateisystems!), so kann jederzeit die Unversehrtheit überprüft werden:

```
for i in $(find / -type f -print); do md5sum "$i" >> database.txt; done
```

Das obige Schema auf alle Dateien anzuwenden, scheitert wohl an der damit verbundenen Aktualisierung der »Datenbank«. Zumindest verlangt sie eine manuelle Kontrolle (z.B. mittels **diff**), ob die Änderung rechtens war.

Hinweis: Wer eine genauere Integritätsprüfung für sein System durchführen will, sollte sich mit dem Programm **tripwire** näher beschäftigen.

siehe auch: Anhang: Skript-Listings → cronjobs

* * * * *

mkdir

Neues Verzeichnis anlegen (make directory).

mkdir <MeinVerzeichnis>

mkdir -p /verzeichnis1/verzeichnis2/verzeichnis3 legt die drei Verzeichnisse in einem Rutsch an - verzeichnis3 ist ein Unterverzeichnis von verzeichnis2 und verzeichnis2 ist ein Unterverzeichnis von verzeichnis1 (p steht für parent directories).

siehe auch: man mkdir

more

Zeigt den Inhalt von Textdateien an.

more <Dateiname>

ls -l --color | more ... die Farben die von **ls** in der Bildschirmausgabe angezeigt werden, bleiben erhalten; bei einer Umleitung zum Standard-Pager **less** ist dies nicht der Fall

siehe auch: man more, less, cat, man zcat, man zmore

mount

Mit mount können Geräte in die Verzeichnisstruktur eingebunden werden. Geräte die schon beim starten des Systems zur Verfügung stehen sollen, sind in der Datei /etc/fstab eingetragen.

sudo mount [PARAMETER] [GERÄTEDATEI oder EINHÄNGEPUNKT / VERZEICHNIS]

mount ... der mount-Befehl, ohne die Angabe von Optionen und Parameter, gibt eine Liste der aktuell in die Linux-Verzeichnisstruktur eingebundenen Geräte und Dateisysteme aus; z.B. /dev/sdb1 für ein USB-Datenträger oder /dev/sr0 für ein DVD-Laufwerk mit eingelegten und vom System erkannten Datenträger (CD, DVD)

mount /media/97E1-1ADB bindet ein USB-Stick in die Verzeichnisstruktur von Linux und erst jetzt können Daten gelesen und geschrieben werden (UUID des verwendeten USB-Stick: 97E1-1ADB)
umount /media/sdd1 löst die Einbindung in die Verzeichnisstruktur wieder und erst jetzt werden alle geänderten Daten - die sich bis jetzt evtl. noch im Cache befinden - auf USB-Stick geschrieben (die Gerätebezeichnung – hier sdd1 - weicht in den verschiedenen Distributionen voneinander ab).

Beispiele 1:

mount -t iso9660 /dev/cdrom /cdrom

mount /dev/hdb2 /home/user/lwd

mount -t reiserfs /dev/hdb3 /mnt/disk2

mount -t smbfs -o username=foo,password=bar

//windowsrechnername/Freigabename /mnt/myshare ... Windows-Freigabe mounten

mount -t cifs -o username=foo,password=bar,ioccharset=utf8,sec=ntlm //windowsrechnername/Freigabename /mnt/myshare ... Windows-Freigabe mounten

mount -r -t reiserfs /dev/hdb3 /mnt/disk2 ... hdb3 nur lesbar mounten - read only (-r)

mount -o remount /dev/hdb3 ... hdb3 ohne Reboot mit neuen Parameter mounten, z.B. nach Änderungen in der Datei /etc/fstab

Beispiele 2:

Windowsverzeichnis mounten:

Auf dem Windowsrechner ist das Verzeichnis **sound** freigegeben. Auf dem Linux-Rechner muss das Verzeichnis **/mnt/windows** bereits existieren.

```
sudo mount -t smbfs -o username=user,password=123456  
//10.102.20.225/sound /mnt/windows
```

Mit `sudo umount /mnt/windows` wird das Verzeichnis wieder aushängen.

ISO-Datei mounten:

ISO-Dateien mounten, d.h. die ISO-Datei kann danach ausgelesen werden ohne dass sie auf CD gebrannt werden muss. Die Angabe des loop-Device `/dev/loop0` kann bei einigen Distributionen entfallen.

```
mount -t iso9660 -o ro,loop=/dev/loop0 /home/[user-name]/ltsp5.4.1-  
0.iso /media/dvd
```

bzw.

```
mount -t iso9660 -o ro,loop /home/[user-name]/ltsp5.4.1-0.iso  
/media/dvd
```

bzw.

```
mount -t iso9660 -o ro,loop /home/[user-name]/ltsp5.4.1-0.iso  
/media/[user-name]/dvd
```

Image-Datei einer DVD mounten:

```
mount -t udf -o ro,loop=/dev/loop0 /home/[user-name]/dvd-image.img /  
media/cdrom0
```

bzw.

```
mount -t udf -o ro,loop /home/[user-name]/dvd-image.img  
/media/cdrom0
```

bzw.

```
mount -t udf -o ro,loop /home/[user-name]/dvd-image.img /media/[user-  
name]/dvd
```

Der Mount-Point (das Verzeichnis) muss existieren, genauso muss dem mount-Kommando der Typ des Dateisystems mitgeteilt werden, falls dieses nicht ext2 ist. Falls das Verzeichnis für den Mount-Point nicht existiert, ist es mit root- oder sudo-Rechten anzulegen.

Ohne Parameter aufgerufen, listet **mount** alle momentan gemounteten Partitionen auf. Weitere unterstützte Dateisysteme sind z.B. iso9660 (CD-ROM), ext3, msdos, nfs, ntfs (nur lesen), reiserfs, usbfs, vfat, smbfs und noch viele andere.

siehe auch: man mount, umount

umount

Mit dem umount-Kommando wird ein Dateisystem wieder aus dem Linux-Verzeichnisbaum ausgehängen.

Beide Kommandos (mount und umount) können per System-Vorgabe nur von root oder mit sudo-Rechten ausgeführt werden (Ausnahme: externe Datenträger, wie USB-Datenträger, DVDs bzw. CD-ROMs).

umount [OPTIONEN] [VERZEICHNIS oder EINHÄNGEPUNKT / GERÄTEDATEI]

Beispiele:

umount /media/cdrom0 ... das angegebene Verzeichnis (Mountpunkt) aushängen

umount /dev/sdb1 ... die angegebene Gerätedatei (USB-Stick) aushängen

umount /media/<Benutzername>/LEXAR ... das angegebene und existierende Verzeichnis (Mountpunkt eines USB-Sticks mit dem Namen LEXAR) aushängen

sudo umount /dev/sda5 ... die angegebene Gerätedatei (1. logisches Laufwerk / Partition in der erweiterten Partition) aushängen

siehe auch: mount, man umount

MP3

siehe auch: Sound

mp3gain

Lautstärke aller MP3-Dateien innerhalb eines Verzeichnisses angleichen . Befinden sich innerhalb eines Verzeichnisses oder Musiksammlung MP3-Dateien aus unterschiedlichen Quellen, so ist die Lautstärke der einzelnen MP3-Dateien sehr wahrscheinlich nicht aufeinander abgestimmt. Mit mp3gain kann man versuchen die Lautstärke der einzelne Musikstücke

anzugleichen.

Es gibt zwei Möglichkeiten, die Lautstärke anzugleichen:

- **Normalisieren:** Das Audiomaterial selbst wird hoch- (lauter) bzw. herunter (leiser) skaliert. Diese Methode funktioniert immer, allerdings ist sie unumkehrbar. Bei falschem Vorgehen kann das Audiomaterial ruiniert werden. Ferner wird es für einige inakzeptabel sein, den Originaltitel zu verändern.
- **Replay Gain:** Das Audiomaterial bleibt unangetastet, stattdessen wird im Tag die gemessene Lautstärke hinterlegt. Audioplayer, die Replay Gain unterstützen, passen dann die Lautstärke automatisch an. Diese Methode ist eleganter, nützt jedoch wenig, wenn Replay Gain nicht unterstützt wird. Dies ist vor allem bei Hardware-Playern recht häufig der Fall.

mp3gain [options] <infile> [<infile 2> ...]

Optionen:

- a ... Albumlautstärke automatisch anpassen
- c ... Clipping-Warnungen ignorieren (nicht empfohlen!)
- d n ... Lautstärke von 89 dB um Wert n erhöhen (in 1,5 dB-Schritten)
- g n ... Gain n hinzufügen ohne Analyse
- h ... Verfügbare Optionen anzeigen
- k ... Track/Album-Gain automatisch verringern, um Clipping zu verhindern
- l 0 n ... Zu Kanal 0 (linker Kanal) Gain n hinzufügen ohne Analyse (funktioniert NUR mit STEREO-Dateien, nicht Joint Stereo)
- l 1 n ... Zu Kanal 1 (rechter Kanal) Gain n hinzufügen
- p ... Erstellungsdatum beibehalten
- r ... Lautstärke des Stücks/der Stücke automatisch anpassen (voreingestellte Lautstärke: 89 dB)
- T ... Datei direkt bearbeiten (ohne temporäre Datei)
- u ... Änderungen rückgängig machen (anhand gespeicherter Tags)

Anmerkungen:

- Wenn man -r und -a angibt, wird nur das Zweite angewandt.
- Sobald -r, -a, -g oder -l benutzt wird, wird kein ReplayGain verwendet, sondern normalisiert. Mit dem Parameter -u kann man diese Anpassungen wieder rückgängig machen.

mp3gain Beispiel.mp3 ... Beispiel.mp3 messen und in den ID3 Tag eintragen (ReplayGain)

mp3gain -r Beispiel.mp3 ... Beispiel.mp3 anpassen/normalisieren (Kein ReplayGain; Titelanpassung, 89 dB)

mp3gain -r -d 3 -p *.mp3 ... alle Dateien im Ordner anpassen (Kein ReplayGain; Titelanpassung, 92 dB, Originaldatum)

mp3gain -a *.mp3 ... Lautstärke aller MP3-Dateien im aktuellen Verzeichnis automatisch angleichen (Kein ReplayGain)

find -type f -name "*.mp3" -print -exec mp3gain -a {} \; ... Lautstärke aller MP3-Dateien im aktuellen Verzeichnis und aller Unterverzeichnisse automatisch angleichen; **Achtung:** Zwischen {} und \ muss sich ein Leerzeichen befinden.

siehe auch: man mp3gain

mp3splt

Musikstück (z.B. Mitschnitt eines Internetradios) in einzelne Lieder auflösen, aufsplitten.

mp3splt -c file_name.cue -o @n_@a_@t file_name.mp3

-c ... die cue-Datei

-o ... das Format für die Ausgangsdateien

@a ... Künstler

@n ... Titeldnummer

@t ... Titelname

Beispiel: Mitschnitt eines Internet-Radiosenders – mukulcast.com

Das koreanische Internet-Radio (Mukulcast, K-POP) wird mit streamripper mitgeschnitten. Der Radiomitschnitt ist ein einziger geschlossener Datenstream, einschließlich der Moderation. Das Programm streamripper legt den Datenstream und die cue-Datei im existierenden Verzeichnis InternetRadio ab. Das Terminalprogramm streamripper wird mit der Tastenkombination **[Strg] + [C]** abgebrochen.

Da das koreanische Internetradio eine andere Textkodierung verwendet, sind hier die codeset-Optionen anzugeben. Verwendet das Internetradio dieselbe Textkodierung wie das eigene System (UTF-8; eigene Textkodierung des Systems ermitteln: **locale**), so können die codeset-Optionen entfallen.

streamripper <http://www.mukulcast.com/#> -a -A -u "Rhythmbox" -d

./InternetRadio --codeset-filesys=UTF-8 --codeset-id3=EUC-KR --codeset-metadata=EUC-KR --codeset-relay=EUC-KR

http://www.mukulcast.com/# ... Internetadresse des Streamservers
-a ... rippen als einzelnen Track; Dateiname: sr_program + aktuelle Unix-Timestamp + .mp3
-A ... keine einzelne Tracks speichern; Gesamtmitschnitt
-d ... Zielverzeichnis für die Dateien; hier InternetRadio
-u "Rhythmbox" - verändert den Namen des UserAgent's für Streamripper (Vorgabe: Streamripper/1.x); einige Radiosender sehen es nicht sehr gern, wenn Ripper-Programme auf ihren Streamserver zugreifen; mit dieser Option kann man versuchen einer Abweisung durch den Streamserver zu entgehen

Mit recode wird die Textkodierung der cue-Datei an das eigene System angepasst.

recode EUC-KR..UTF-8 file_name.cue

Mit dem Terminalprogramm **mp3spl**t wird versucht den Datenstream in einzelne Musikstücke aufzulösen, zu splitten. Falls dies bei einzelnen Musikstücken nicht gelingt (Pause innerhalb eines Musikstücks, Zeit zwischen Musikstück und Moderation ist zu kurz), muss der Datenstream z.B. mit dem grafischen Programm Audacity manuell bearbeitet werden.

mp3spl -c file_name.cue -o @n_@a_@t file_name.mp3

Hinweis: codeset

Falls in der Terminalausgabe von streamripper seltsame Zeichen auftauchen, so stimmen der Zeichensatz des Streamservers und des eigenen Systems nicht überein. Nachfolgend sind einige Erfahrungswerte aufgelistet.

Westeuropa: --codeset-filesys=UTF-8 --codeset-id3=ISO-8859-15 --codeset-metadata=ISO-8859-15 --codeset-relay=ISO-8859-15

Korea: --codeset-filesys=UTF-8 --codeset-id3=EUC-KR --codeset-metadata=EUC-KR --codeset-relay=EUC-KR

siehe auch: man mp3spl, recode, streamripper, mp3gain, mp3wrap

mp3wrap

Hilfsprogramm für die Zusammenführung mehrerer MP3-Dateien zu einer einzelnen MP3-Datei.

mp3wrap ist ein Werkzeug, welches verlustfreies Zusammenführen von MP3-Dateien ohne erneutes Kodieren (Komprimieren) erlaubt. Die Dateinamen, Tags und ursprünglichen Nicht-Audio-Dateien (z.B. Wiedergabelisten, Info-Dateien und Bilder der Cover) innerhalb der MP3-Datei bleiben dabei erhalten.

Die durch diese Methode verbundenen Dateien lassen sich mit mp3splt wieder trennen.

mp3wrap [Optionen] AUSGABEDATEI.mp3 1.mp3 2.mp3

Beispiele:

mp3wrap ERGEBNIS.mp3 TITEL-0*.mp3 ... alle MP3-Dateien (TITEL-01.mp3, TITEL-02.mp3, ...) im aktuellen Verzeichnis werden zu einer einzigen Datei ERGEBNIS.mp3 verbunden

mp3wrap -a ERGEBNIS_MP3WRAP.mp3 DATEI* ... Mit der Option -a können Dateien nachträglich hinzugefügt werden.

mp3wrap ERGEBNIS.mp3 PLAYLIST.m3u INFO.nfo

EINZELTITEL-0*.mp3 COVER.jpg ... Playlists, Info-Dateien und Cover können ebenfalls archiviert werden. Textdateien an den Anfang und Bilddateien an das Befehlsende setzen.

mp3wrap -l ERGEBNIS_MP3WRAP.mp3 ... Archivinhalt einsehen - Auskunft über enthaltene Dateien und deren Anzahl erhalten.

Anmerkung: Falls einige Daten des Dateiheders (z.B. Gesamtspieldauer der zusammengeführten MP3-Stücke) von mp3wrap nicht korrekt eingetragen wurde, so kann die Datei in Audacity (Voraussetzung: Programmpakete audacity und lame sind installiert) importiert und anschließend wieder als MP3-Datei exportiert werden. Eine andere Alternative ist das Programm avconv (**siehe auch:** avconv; avconv ist Bestandteil des Paketes libav-tools):

avconv -i INFILE.mp3 -ab 128k -acodec libmp3lame -ac 2 OUTFILE.mp3

bzw.

avconv -i INFILE.mp3 -acodec copy OUTFILE.mp3

siehe auch: man mp3wrap, mp3splt, mp3gain, eyed3

MPlayer

Mplayer (Programmpaket: mplayer2, mplayer-nogui oder mplayer) ist ein

Terminalprogramm zum Abspielen von Videodateien. Für einige Videoformate sind die entsprechenden Codecs (weitere Informationen finden sie im Internet) noch zu installieren.

`mplayer [options] [url|path/] filename`

mplayer video.mov ... den angegebenen Film (Quicktime Movie) in Originalgröße starten

mplayer *.avi ... alle AVI-Filme im aktuellen Verzeichnis nacheinander in Originalgröße starten

mplayer -fs video.mov ... im Vollbildschirm (fs ... fullscreen) den angegebenen Film starten

mplayer -nosound video.mov ... den angegebenen Film ohne Sound starten

mplayer -speed 0.2 video.mov ... den angegebenen Film langsam abspielen (0.01 ... 100)

mplayer -frames 100 video.mov ... nur die ersten 100 Frames wiedergeben

mplayer video.mov -ss 00:15:30 ... -ss <timepos> (Sekunden oder hh:mm:ss) Startposition des Films; -ss 56 springt zur Position 56 Sekunden; -ss 01:10:00 springt zur Position 1 Stunde 10 Minuten

mplayer -monitoraspect 4:3 video.mp4 ... Seitenverhältnis (4:3; 16:9) des Videos angeben; mitunter erkennt mplayer das richtige Seitenverhältnis nicht automatisch

mplayer tv:// ... vom System unterstützte Webcam starten

mplayer dvd://1 -dvd-device /dev/hdc ... die eingelegte DVD abspielen, Gerät /dev/hdc verwenden

mplayer dvd://1 -dvd-device /dev/sr0 ... die eingelegte DVD abspielen, Gerät /dev/sr0 verwenden

mplayer dvd:// ... die eingelegte DVD im Standard-DVD-Laufwerk abspielen

mplayer dvd://1 ... die eingelegte DVD im Standard-DVD-Laufwerk abspielen

mplayer -nocache dvdnav:// ... Navigationsmenü der eingelegten DVD aufrufen

mplayer dvd://5 -dumpstream -dumpfile output.vob ... den 5. Titel auf der DVD im aktuellen Verzeichnis ablegen

mplayer -alang de dvd://1 ... die eingelegte DVD im Standard-DVD-Laufwerk abspielen, Audiostream -> deutsch

mplayer -alang de -slang en dvd://1 ... die eingelegte DVD im Standard-DVD-Laufwerk abspielen, Audiostream -> deutsch, Untertitel -> englisch;

statt `-slang` kann auch `-sid` (Subtitel-ID) verwendet werden - die ID des Subtitels (z.B. `-sid 3`) erfährt man im Verbose-Modus; statt der Optionen `-slang` oder `-sid` kann auch die Option `-vobsubsid` genutzt werden

mplayer -aid 129 -sid 3 -dumpsub dvd://1 ... die eingelegte DVD im Standard-DVD-Laufwerk abspielen, Audiostream mit der ID 129, Untertitel mit der ID 3; die ID des Audiostreams (`-aid 129`) und Subtitels (z.B. `-sid 3`) erfährt man im Verbose-Modus (Option `-v`)

mplayer -aid 129 -vobsubid 3 -dumpsub dvd://1 ... wie vorhergehendes Beispiel; die Option `-vobsubid` ist anzuwenden, falls die Option `-sid` nicht die gewünschte Wirkung zeigt

Hinweis: DVDs benutzen den zweibuchstabigen ISO 639-1 Sprachcode (de, en, hu etc.). MPlayer gibt alle vorhandenen Sprachen aus, wenn er im Verbose-Modus (`-v`) gestartet wird.

mplayer dvd:// -v ... startet die eingelegte DVD im Verbose-Modus

Hinweis: Sind auf einer DVD mehrere Titel gespeichert, so ist die DVD wie folgt aufzurufen:

mplayer -v dvd://1-3

Hier im Beispiel besteht der Film aus 3 Titeln. Wie viele Titel auf der DVD gespeichert sind erfährt man, wenn die DVD im Verbose-Modus (`-v`) gestartet wird.

Besitzt ein Titel mehrere Kapitel und es soll nur ein oder einige wenige Kapitel des DVD-Titels gespielt werden, so ist folgendes einzugeben:

mplayer -v dvd://1-3 -chapter 12 -dvd-device /dev/hdc

mplayer -v dvd://1-3 -chapter 12-15 -dvd-device /dev/hdc

Hier im Beispiel werden vom DVD-Titel 1 das Kapitel 12 bzw. die Kapitel 12 bis 15 gespielt. Wie viele Kapitel auf der DVD gespeichert sind erfährt man, wenn die DVD im Verbose-Modus (`-v`) gestartet wird.

Anmerkung: In den meisten Fällen erkennt Mplayer das DVD-Laufwerk automatisch, d.h. die Angabe des `dvd-device` kann entfallen. In einigen Fällen muss das DVD-Laufwerk (`dvd-device`) angegeben werden. In den vorgenannten Beispielen heißt das DVD-Laufwerk **/dev/hdc**. Unter Ubuntu oder bei Linux-Distributionen die auf Ubuntu basieren, wird das DVD-Laufwerk meisten mit **/dev/sr0** angesprochen.

Der Name des DVD-Laufwerkes kann bei eingelegter und erkannter DVD

mit dem Terminalbefehl **mount** ermittelt werden (i.d.R. letzte Zeile der Terminalausgabe).

Nützliche Optionen:

-cache <kbytes> ... Diese Option gibt an, wie viel Speicher (in kBytes) MPlayer zum Precachen einer Datei oder URL benutzt. Besonders bei langsamen Medien sinnvoll.

-cache-min <Prozent> ... Die Wiedergabe startet, wenn der Cache bis zu <Prozent> des Gesamten gefüllt ist.

-stop-xscreensaver ... (nur bei X11) Deaktiviert den Bildschirmschoner beim Start von MPlayer und aktiviert ihn beim Beenden wieder.

mplayer -v -alang de dvd://1 -nortc -stop-xscreensaver

Bedienung - Tastenkombinationen:

[Pfeil links] oder **[Pfeil rechts]** ... rückwärts / vorwärts 10 Sekunden

[Pfeil hoch] oder **[Pfeil runter]** ... vorwärts / rückwärts 1 Minute

[Bild hoch] oder **[Bild runter]** ... vorwärts / rückwärts 10 Minuten

[und] ... Verringert bzw. erhöht die Abspielgeschwindigkeit um 10%.

{ und } ... Halbiert bzw. verdoppelt die Abspielgeschwindigkeit.

. (Punkt) ... Einen Schritt vorwärts. Einmaliges Drücken pausiert die Wiedergabe, jedes weitere wird einen Frame abspielen und die Wiedergabe erneut anhalten (jede andere Taste hebt die Pause auf).

[Leertaste] oder **[p]** ... Pause / Start

[ESC] oder **[q]** ... Stopp und Beenden des Programms

Untertitel

v ... Ändert die Sichtbarkeit der Untertitel (aus/ein).

b oder **j** ... Wechselt durch die verfügbaren Untertitel.

Farbton, Helligkeit, Kontrast

1 und **2** ... Passe Kontrast an.

3 und **4** ... Passe Helligkeit an.

5 und **6** ... Passe Farbton an.

7 und **8** ... Passe Sättigung an.

Lautstärke

/ und ***** ... Verringert / erhöht die Lautstärke.

9 und **0** ... Verringert / erhöht die Lautstärke.

M ... Ändert die Ton-Stummschaltung (ein/aus).

... Audiospur wechseln

Bildschirm

f ... Ändert die Vollbild-Wiedergabe (ein/aus).

o ... On-Screen-Display wechseln

Audiostream einer DVD in eine MP3-Datei umwandeln

1. DVD einlegen
2. ein Terminal öffnen und folgendes eingeben

Beispiel 1: vom 1. Titel ein Audiostream erstellen

mplayer dvd://1 -dumpaudio

Beispiel 2: deutsche Sprache einstellen

mplayer -alang de dvd://1 -dumpaudio

Beispiel 3: nur vom Kapitel 3 ein Audiostream erstellen

mplayer -alang de dvd://1 -chapter 3 -dumpaudio

3. der Audiostream stream.dump (Default-Name) wird im aktuellen Verzeichnis abgelegt
4. Audacity öffnen und die Datei stream.dump über «Datei» «Import» «Audio» einlesen
5. die Datei stream.dump als MP3-Datei speichern
über «Datei» «Exportieren ...» «OK» «Speichern» diese Datei als
z.B. file.mp3 speichern

Die Datei file.mp3 kann dann mit Audacity noch bearbeitet werden - z.B. den gewünschten Ausschnitt herausschneiden.

Tonspur mit Mplayer extrahieren

Der nachfolgende Befehl extrahiert den Ton in eine WAVE-Datei namens test.wav. Diese kann anschließend in das gewünschte Endformat umgewandelt werden.

mplayer -vc null -vo null -ao pcm:fast:waveheader:file=test.wav dvd://# -chapter #-#

Die Rauten (#) sind hierbei Platzhalter für den DVD-Titel und dessen Kapitel. Möchte man beispielsweise aus dem zweiten Titel der DVD die Kapitel 3 bis 4 extrahieren, ändert sich die Eingabe wie folgt:

mplayer -vc null -vo null -ao pcm:fast:waveheader:file=test.wav dvd://2 -chapter 3-4

Verfügt die DVD über mehrere Tonspuren pro Titel (z.B. Englisch, Deutsch, Französisch), kann über den Parameter **-aid** die gewünschte Tonspur gewählt werden.

1. die aid-Nummer für die Sprache ermitteln

mplayer dvd:// -v ... die eingelegte DVD mit der Option **-v** starten

Die Ausgabe im Terminal sieht dabei in etwa so aus:

```
There are 1 titles on this DVD.  
There are 1 angles in this DVD title.
```

```
audio stream: 0 format: ac3 (stereo) language: Deutsch aid:  
128.  
[...]
```

2. Tonspur in eine Wave-Datei speichern

mplayer -vc null -vo null -aid 128 -ao pcm:fast:waveheader:file=test.wav dvd://1 -chapter 1-2

Die Wave-Datei kann anschließend in das gewünschte Endformat umgewandelt werden.

siehe auch: Sound, avconv, lsdvd

Webcam mit MPlayer

Vom System unterstützte Webcam starten:

mplayer tv://

Bild von der Webcam im aktuellen Verzeichnis speichern. Das Programm **import** und **convert** sind Bestandteil des Programmpaketes **ImageMagick**:

import -window MPlayer webcam.png

bzw.

import -window MPlayer webcam.jpg

Bildausschnitt herausschneiden - i.d.R. nicht notwendig:

Größe des Ausgangsbildes: 640x480 Pixel

Zielgröße des Bildes: 600x400 Pixel

Startpunkt des neuen Bildes: 20 Pixel von oben und 40 Pixel von links

Seite des Ausgangsbildes

Ausgangsformat: PNG

Zielformat: JPEG

convert webcam.png -crop 600x400+20+40 -quality 90 webcam.jpg

Bild über scp (secure copy) zum Server hochladen:

scp webcam.jpg

user_name:user_passwort@server_name:~/pfad/zum/bildverzeichnis/bild_001.jpg

Mit einem kleinen Shellskript kann man auf diese Weise z.B. alle 10 Minuten ein Bild hochladen und damit im Internet verfügbar machen.

siehe auch: mplayer --help, man mplayer, avconv, lsdvd, vobcopy

* * * * *

mtr

mtr (My Traceroute) gehört nicht zur Standardausrüstung eines Linux-Rechners, es muss i.d.R. erst nachinstalliert werden. **My Traceroute** kombiniert die Fähigkeiten von **ping** und **traceroute**.

mtr <Domainname bzw. IP-Adresse>

mtr (Aufruf: /usr/sbin/mtr <Domainname>) läuft ohne anderweitige Angabe so lange, bis der Benutzer es unterbricht (Tastenkombination **[Strg] + [C]** bzw. Taste **[Q]**). Es ermittelt zunächst alle Stationen, die der Datenverkehr zwischen dem eigenen und dem Zielrechner passiert. Danach verwendet mtr das Programm ping, um von jeder Station eine Statistik über die Erreichbarkeit zu erstellen.

Die Spalte »Loss%« führt die Zahl der irgendwo auf dem Weg verloren

gegangenen Pakete auf, die Gesamtzahl steht in der nächsten Spalte »Snt«
»Last« gibt die Reaktionszeit beim letzten verschickten Paket an, dahinter
folgen die von ping bekannten Werte: durchschnittliche, schnellste und
langsamste Reaktion und die Standardabweichung.

Mit den Kommandozeilentools ifconfig, host, route, ping, traceroute bzw.
mtr und netstat diagnostizieren Sie bei Netzwerkstörungen - richtig
eingesetzt - gezielt jeden Teilaspekt einer Verbindung, intern oder im
Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert -
die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So
finden Sie möglicherweise heraus, ob es sich bei einer nicht
funktionierenden internen oder Internet-Verbindung um ein Problem mit
einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS
oder eines der Gegenseite handelt.

mtr -rwc 4 <Internet-Adresse>

-rwc 4 ... r .. Report-Modus – mit statistischen Angaben; w .. lange
Hostnamen werden nicht gekürzt; c (count) .. Anzahl der Runden (ping,
traceroute)

<Internet-Adresse> ... www.domain-name.de oder 122.45.111.94

siehe auch: man mtr, host, route, ping, netstat, traceroute und ifconfig

mv

Mit mv kann man Dateien verschieben und umbenennen.

mv <AlterName> <NeuerName> ... Datei umbenennen

mv <Quelle[n]> <Zielverzeichnis> ... Datei ins Zielverzeichnis
verschieben (move)

Optionen:

-f ... vor dem Überschreiben einer Datei **nicht** nachfragen; f .. force
-i ... vor dem Überschreiben einer Datei nachfragen; i .. interactive

Beispiele:

1. Dateien umbenennen

Mit einer for-Schleife können mehrere Dateien in einem Zuge umbenannt
werden. Im Beispiel wird die Dateiendung **txt** aller Textdateien im aktuellen

Verzeichnis durch die Dateieindung **csv** ersetzt. **Hinweis:** Funktioniert nicht mit Dateinamen die ein Leerzeichen enthalten.

```
for f in *.txt; do mv ${f} ${f%.txt}csv; done
```

2. Viele Dateien in einem Zuge verschieben

```
mv *.jpg Urlaub/
```

Bei sehr vielen Dateien führt dieser Befehl zu einem Programmabbruch. Bedingt durch den Wildcard (*) entsteht eine sehr lange Liste mit tausenden Dateien, die vom Rechner nicht mehr bewältigt werden kann.

Mit `find` umgeht man dieses Problem, da der Befehl `mv` hier immer nur auf eine gefundene Datei angewendet wird.

```
find . -maxdepth 1 -type f -iname '*.jpg' -exec mv {} Urlaub/;
```

siehe auch: `man mv`, `find`

N

netstat ein »Alleskönner«

Anzeige von Netzwerkverbindungen, Routentabellen, Schnittstellenstatistiken, maskierten Verbindungen, Netlink-Nachrichten und Mitgliedschaft in Multicastgruppen.

netstat ... netstat ohne Parameter gibt eine Liste der sogenannten Sockets aus. Dabei handelt es sich um spezielle Dateien, über die Anwendungen miteinander kommunizieren, für gewöhnlich ein Server mit seinen Clients. In der ersten Spalte steht das Protokoll; hierbei sind die zahlreichen mit unix gekennzeichneten Sockets für das Netzwerk uninteressant, da sie lediglich der internen Kommunikation dienen.

netstat -a ... zeigt alle Netzwerkverbindungen an

netstat -uta ... zeigt von den laufenden Netzwerkverbindungen nur Verbindungen per TCP oder UDP ins Internet oder des lokalen Netzwerkes an

fuser -v 32797/tcp ... zeigt welches Programm den Port 32797 über das Protokoll TCP zur Zeit benutzt

netstat -r ... erzeugt dieselbe Ausgabe wie **route**, nur das man mit netstat -r auch als normaler Benutzer diesen Befehl ausführen kann

netstat -l ... zeigt auf welche Verbindungstypen der Rechner wartet - also welche Dienste vom Server bereitgestellt werden

netstat -t ... zeigt für das Internet relevante Sockets mit dem Protokoll TCP/IP an

netstat -c -t ... behält Sockets mit dem Protokoll TCP/IP dauerhaft im »Auge«

netstat -i ... zeigt die Netzwerkgeräte an

netstat -s ... fasst die Verbindungsdaten zusammen

Beispiel:

netstat -l -t

In der Ausgabe steht jede Zeile für einen angebotenen Server-Dienst. Die Spalte »Local Address« gibt hinter dem Doppelpunkt den Netzwerk-Port aus - die wird gerne mit der Hausnummer verglichen, wenn der PC der Straße entspricht - oder, falls bekannt, den Namen des Dienstes. Der Eintrag vor dem Doppelpunkt beschreibt die Netzwerk-Adressen, von denen aus fremde Rechner auf einen Dienst zugreifen dürfen. Ein Stern bedeutet, dass der Dienst keiner derartigen Einschränkung unterliegt. Steht dort ein einzelner Rechner oder eine Internet-Domain, können nur diese Netzwerkteilnehmer den Server-Dienst nutzen. Häufig wird hier der Name des eigenen Systems (localhost) aufgeführt, um den Zugriff aus dem Netz komplett zu

unterbinden.

Diensten auf der Spur

Auf einem Linux-Rechner laufen nicht nur Programme, die Sie als Benutzer bewusst über die Menüs oder Terminals starten. Viele Anwendungen erledigen ihre Aufgaben still und leise im Hintergrund, ohne dass Sie es bemerken. Sie räumen Log-Dateien auf (Logrotate), führen zu fixen Zeiten Befehle aus (Cron/Anacron) und mixen den Sound diverser Anwendungen (PulseAudio).

Bei diesen Diensten unterscheidet man zwischen rein lokalen Dienstprogrammen und Serverdiensten, die sich über das lokale Netzwerk oder gar das Internet ansprechen lassen. Lokale Dienste (im Englischen "Daemons") wie Cron oder Logrotate arbeiten so gut wie auf jedem Linux-System. Einmal installiert und eingerichtet, werkeln sie ohne Ihr Eingreifen im Hintergrund.

Dagegen stehen Serverdienste wie der SSH-Daemon, Web- oder FTP-Server aber auch Multimedia-Server wie PulseAudio in direktem Kontakt mit anderen Rechnern oder Programmen im Netzwerk. Sicherheitstechnisch sollten Sie diese im Auge behalten. Grundlos laufende Server – eventuell noch mangelhaft konfiguriert – bieten Angreifern eine Angriffsfläche, zudem gelangen Ihre Daten mitunter ungewollt an die Öffentlichkeit.

Grundsätzlich steckt im Client-Server-Modell jedoch die Möglichkeit, Aufgaben und Dienste lokal oder über ein Netzwerk zu verteilen. Dabei fordert der Client meist über das Netzwerk eine Aufgabe an, während der Server diese erledigt und das Ergebnis zurückschickt. Solchen Serverdiensten wollen wir etwas auf den Grund gehen.

Was lauscht bei mir?

Serverdienste laufen nicht nur in Rechenzentren und liefern Webseiten aus, sondern sie erledigen beliebige Aufgaben auf herkömmlichen Computern. Selbst Ihr Linux-System startet von Hause aus einige Programme, die als Serverdienste unbemerkt im Hintergrund arbeiten. Dabei benötigen sie nicht einmal ein Netzwerk: Fehlt dieses, "unterhalten" sich viele Anwendungen über die Loopback-Schnittstelle.

Fast jeder Computeranwender, der sich ein wenig mit Sicherheit im Internet beschäftigt, kennt vermutlich den Begriff "Netzwerkport". In Verbindung mit Personal Firewalls ranken sich vielfältige Legenden um diese Ports. Generell nutzen Dienste in den heutigen Netzwerken meist die

Transportprotokolle TCP und UDP. Beide verwenden durchnummerierte Ports, die von 1 bis 65535 (2^{16}) reichen. Im Gegensatz zu Client-Anwendungen reagieren Serverdienste auf alle eingehenden Datenpakete, die sie auf dem Port empfangen, auf dem sie lauschen. Hier ist es wichtig, zwischen offenen Client- und Serverports zu unterscheiden.

Ein frisch aufgesetztes Ubuntu-System oder Systeme die auf Ubuntu basieren (Linux Mint, Xubuntu, Lubuntu ...) öffnen aufgrund der "Keine-offenen-Ports"-Regel keine Ports in die angebundenen Netzwerke. Diese Regel macht das Einrichten einer Firewall auf einem Ubuntu-System meist überflüssig. Einfach zu merken: Wo nichts lauscht, gibt's auch keine Lauschangriffe. Mit der Uncomplicated Firewall (kurz UFW) installiert Ubuntu zwar auch ein einfach zu bedienendes Kommandozeilen-Frontend für Iptables, doch UFW ist von Hause aus nicht aktiv.

Sie können auf Ihrem Ubuntu-System selbst prüfen, welche Anwendungen permanent Server-Ports offen halten.

sudo netstat -tulpen

Laufen bereits aktive Serverdienste auf dem Rechner, sortieren Sie lokale Ipv4- und Ipv6-Dienste aus der Ergebnisliste heraus:

sudo netstat -tulpen | grep -v '127.0.0.1|:::1'

Bei einer frischen Ubuntu-Installation (gilt i.d.R. Auch für Systeme die auf Ubuntu basieren) laufen lediglich der Druckdienst CUPS, der Avahi-Dienst (der Ressourcen im Netzwerk aufspürt) sowie der DHCP-Client, der bei der automatischen Konfiguration von Netzwerkkarten hilft.

Avahi und der DHCP-Client öffnen hingegen Ports für andere IP-Adressen, allerdings verwirft der DHCP-Client alle Pakete, die nicht aus dem lokalen Netzwerk stammen (und somit die für lokale Netzwerke reservierten IP-Adressen verwenden). Ähnliches gilt für Avahi: Da der Dienst auf Broadcast-Nachrichten lauscht, verlässt auch hier nichts die Grenzen des eigenen Netzwerks.

Dienste starten/stoppen?

Lange Jahre diente **SysVinit** beim Booten als DAS Init-System von Linux. Doch das stoische, serielle Abarbeiten von Aufgaben verzögerte den Boot-Prozess zunehmend. Seit Ubuntu 6.10 übernahm nach und nach das von Canonical entwickelte **Upstart** das Zepter. Das startet Dienste parallel und

Ereignis-basiert und ruft z. B. keinen Netzwerkdienst auf, wenn es noch kein Netzwerk findet. Das beschleunigt den Bootprozess und erleichtert den Umgang mit auswechselbarer Hardware.

Um Dienste zu starten und zu stoppen, müssen Sie ihre Namen kennen, die auch Netstat nicht immer richtig ausgibt. Eine Liste aller von Upstart kontrollierten Dienste zeigt Ihnen der Aufruf von **initctl**. Zusätzlich zum Namen verrät das "Init Daemon Control Tool" – so die Langfassung – den aktuellen Status sowie die Nummer eines Prozesses.

initctl list

Dementsprechend starten und beenden Sie Init-Jobs auch über Initctl. Die Syntax lautet:

sudo initctl [start/stop] ssh ... ssh-Server starten bzw. stoppen

Bleibt noch das Problem, dass Ubuntu klassische SysVinit- und neue Upstart-Skripte parallel nutzt. So steuert Upstart zum Beispiel den SSH-Server OpenSSH, aber nicht den Webserver Apache. Von daher führt **initctl list** Apache nicht als Dienst auf und liefert die Eingabe von **sudo initctl stop apache2** nur eine Fehlermeldung.

Hier bietet es sich an, auf das SysVinit-Werkzeug **service** zurückzugreifen. Dieses arbeitet für den Benutzer völlig transparent mit beiden Skriptarten zusammen. Über **sudo service [TAB][TAB]** erhalten Sie eine Liste aller Dienste (egal ob SysVinit oder Upstart diese steuert, Alternativaufruf: **sudo service --status-all**). Anschließend erfahren Sie über den Befehl **service dienstname**, welche Anweisungen ein Dienst versteht. Für **ssh** sieht das dann so aus:

sudo service ssh

Danach starten oder beenden Sie den Dienst über:

sudo service ssh [start/stop/...]

Anmerkung:

Mit den Kommandozeilentools **ifconfig**, **host**, **route**, **ping**, **traceroute** bzw. **mtr** und **netstat** diagnostizieren Sie bei Netzwerkstörungen - richtig eingesetzt - gezielt jeden Teilaspekt einer Verbindung, intern oder im Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert - die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So

finden Sie möglicherweise heraus, ob es sich bei einer nicht funktionierenden internen oder Internet-Verbindung um ein Problem mit einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS oder eines der Gegenseite handelt.

siehe auch: host, route, ping, traceroute bzw. mtr und ifconfig

Netzwerk manuell aufsetzen

Linux ist ein Mehrbenutzersystem, und viele Funktionen sind bestens für den Netzbetrieb geeignet. Ein Linux-Rechner lässt sich schnell in ein bestehendes Netzwerk integrieren. Dafür müssen – wenn überhaupt – nur ein paar Dateien angepasst werden und die Netzwerkverbindungen anschließend mit

/etc/init.d/networking restart

neu gestartet werden.

Schon während der Installation wird ein Linux-PC für das Netzwerk vorbereitet. In der Standardinstallation lässt er sich eine IP-Adresse von einem DHCP-Server zuweisen. Solche Server sind beispielsweise in den meisten Internet-Routern enthalten – in dem Fall muss nichts mehr weiter eingerichtet werden.

Ist kein DHCP-Server im Netzwerk vorhanden, so ist die Datei **/etc/network/interfaces** anzupassen. Diese kann dann etwa folgende Einstellungen haben:

```
auto eth0
iface eth0 inet static
    address 192.168.1.23
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
    gateway 192.168.1.254
```

Schnittstellen-Definitionen beginnen mit »iface«. In dieser Zeile stehen der Name der Schnittstelle (»eth0« für den ersten Netzwerkadapter), die Adressfamilie (»inet« für TCP/IP) und die Art der Zuweisung der IP-Adresse. Hier steht »static« für eine statische IP-Adresse. Anschließend folgen die Optionen »address« (IP-Adresse des Rechners), »netmask« (Netzmaske des Netzwerks), »network« (IP-Adresse des Netzwerks), »broadcast« (Broadcast-Adresse) und »gateway« (IP-Adresse des Gateways).

Erforderlich sind nur die Angaben »address« und »netmask«. Erhält der Rechner seine IP-Adresse von einem DHCP-Server, kann auch darauf verzichtet werden. In dem Fall lautet die »iface«-Zeile ganz einfach

```
iface eth0 inet dhcp
```

Will man Rechner anstelle der IP-Adresse mit Namen anreden, so sind noch 2 Dateien anzupassen. In die Datei **/etc/hostname** ist der Name des Rechners einzutragen, zum Beispiel »karl«. Damit jeder Rechner weiß, wer ebenfalls im Netzwerk ist, ist noch die Rechnertabelle anzupassen (in großen Netzwerken überlässt man dies einem DNS-Server).

Die Tabelle steht in der Datei **/etc/hosts**. Sie muss auf allen im Netzwerk verbundenen Rechner gleich sein, denn nur dann kann man einen Rechner – statt ihn mit der IP-Adresse anzusprechen – beim Namen nennen. Nachdem die Dateien angepasst wurden und die Tabelle auf alle Rechner kopiert wurden, kann man mit dem Befehl »ping <Rechnername>« von jedem anderen Computer testen, ob Datenpakete ankommen.

siehe auch: ifconfig, ip

Netzwerkkarte - Hersteller der Karte ermitteln

Für die IP-Kommunikation über das Ethernet ist ARP (Address Resolution Protocol) ein unverzichtbarer Bestandteil, da die eigentliche Adressierung im Ethernet anhand der MAC-Adressen stattfinden. Eine MAC-Adresse (Medium Access Control) ist eine auf der Netzwerkkarte festgelegte Kennung, die im Normalfall einzigartig und unveränderbar ist. Die MAC-Adresse hat eine Länge von 48 Bit und wird in der Regel hexadezimal geschrieben.

Die ersten 24 Bit dieser Adresse beinhalten die Herstellerkennung der Netzwerkkarte, die von der IEEE (Institute of Electronic Engineers) vergeben wird.

Zum Beispiel die MAC-Adresse (Hardware Adresse): **00-0D-56-xx-xx-xx**

Die MAC-Adresse wird mittels des Befehls **ifconfig** ermittelt - evtl.

/sbin/ifconfig eingeben (1. Zeile der Bildschirmausgabe: Hardware Adresse). Mittels dieser MAC-Adresse können Sie als Hersteller die Firma DELL ermitteln. Eine Liste aller herstellerbezogenen MAC-Adressen finden Sie unter der Adresse **<http://standards.ieee.org/regauth/oui/oui.txt>**.

Um die MAC-Adressen oder die Hersteller von anderen Hardware-Geräten zu ermitteln, lohnt sich manchmal ein Blick in die Datei **/var/log/messages** bzw. **/var/log/kern.log** (**siehe auch:** **tail -f /var/log/messages** bzw. **/var/log/kern.log**).

siehe auch: ifconfig, ip, arp

nl

Ergänzt eine Textdatei um Zeilennummern. Damit erzeugt man beispielsweise nummerierte Listings oder nummeriert Datensätze.

nl datei1.txt

siehe auch: man nl

nmap

Mit dem Portscanner nmap kann man die offenen Ports von erreichbaren Rechner ermitteln.

nmap 192.168.0.5 ... die einfachste Form zur Ermittlung offener Ports auf dem angegebenen Rechner

Dabei ist zu beachten - je stärker ein System abgeschottet ist, umso länger dauert der Portscan. Mit [Strg] + [C] brechen Sie das Programm ab, falls Ihnen die Geduld ausgeht. Die Angaben in der Spalte SERVICE entsprechen dabei nicht immer dem Dienst, der tatsächlich auf diesem Port läuft. Nmap bezieht diese Informationen aus einer mitgelieferten Datei.

Um herauszufinden, welche Programme sich wirklich dahinter verbergen, verwenden Sie die Option **-sV**. Dabei versucht Nmap (Network Mapper), anhand typischer Antwortmuster die Programme und deren Versionen zu erkennen. Dieses Verfahren funktioniert nicht hundertprozentig zuverlässig, aber oft gewinnen Sie daraus interessante Erkenntnisse.

Sind Ports durch eine **Firewall** geschützt, gibt Nmap für diese den Status **filtered** (gefiltert) zurück; Ports, auf denen **kein Dienst** horcht, sind **closed** (geschlossen).

Scannen ist eine Sache, etwas anderes ist es, die gewonnenen Daten richtig zu interpretieren. Was bedeutet ein offener Port? Zunächst einmal heißt das nicht mehr, als dass dort ein von anderen Rechnern zugänglicher Dienst läuft. Das kann Absicht oder ein Versehen sein.

Zeigt Ihnen Nmap beispielsweise als Ausgabe Port 80 als offen an, läuft dort aller Wahrscheinlichkeit nach ein Web-Server. Im Browser können Sie die Adresse eingeben und über das HTTP-Protokoll von diesem Rechner Web-Seiten abrufen. Ein offener Port 22 deutet auf eine Secure-Shell (SSH) hin: Auf diesem Rechner können Sie sich, sofern Sie einen Linux-Benutzer-Account darauf haben, mit dem Kommando **ssh rechnername** anmelden. Auf Port 25 wartet meist ein Mail-Server auf E-Mails.

All das stellt noch keine Gefahr dar, solange der Eigentümer des Rechners diesen Dienst bewusst aktiviert hat. Gehört Ihnen der gescannte Rechner und haben Sie den Dienst weder absichtlich angeschaltet noch aktuellen Bedarf dafür, sollten Sie ihn deaktivieren. Das erledigen Sie über den Runlevel-Editor Ihrer Distribution.

Nmap Tutorial

Portscanner sind eines der wichtigsten Arbeitsmittel von Administratoren. Dieses Tutorial beschreibt den Einsatz des populären Portscanner Nmap unter Linux.

Grundlagen zu Protokollen und Ports

Um einen Dienst auf einen IP-Host eindeutig zu identifizieren werden drei Angaben benötigt:

die IP-Adresse (z.B. 10.0.0.1)

das Protokoll (z.B. TCP)

die Portnummer (z.B. 80 für HTTP)

Diese drei Angaben identifizieren einen Webserver (HTTP) der auf dem Host mit der IP-Adresse 10.0.0.1 läuft. Die Portnummer ist eine 16-Bit-Zahl zwischen 0 und 65535. Für Standarddienste werden die Portnummern von der IANA vergeben. Dort finden Sie auch die aktuelle Liste der Portnummern (www.iana.org/assignments/port-numbers). Die Aufgabe eines Portscanners ist es nun, die offenen Ports eines Systems zu ermitteln. Also die Ports, auf denen ein Service angeboten wird.

Beispiel:

```
[root] ~ $ nmap -sT m0n0
```

```
Starting nmap 3.20 ( www.insecure.org/nmap/ ) at 2005-05-01 11:17 CEST
```

```
Interesting ports on m0n0 (10.0.0.1):
```

```
(The 1610 ports scanned but not shown below are in state: closed)
```

Port	State	Service
80/tcp	open	http

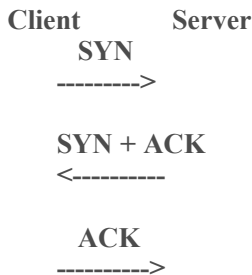
```
Nmap run completed -- 1 IP address (1 host up) scanned in 10.886 seconds
```

Mit dem Kommando `nmap -sT m0n0` untersucht nmap den Host m0n0 auf

offene TCP-Ports. Dabei überprüft nmap nicht alle 65535 TCP-Ports. Per Default werden nur etwa 1600 »wichtige« Portnummern gescannt. Auf dem Host m0n0 wird ein offener Port entdeckt (HTTP). Der Scan dauert etwa 11 Sekunden. Grundsätzlich sollten auf einem System nur die Ports offen sein, die wirklich benötigt werden.

TCP Portscans

TCP ist das verbindungsorientierte Protokoll der TCP/IP-Familie auf dem OSI-Layer 4. Vor jeder Datenübertragung baut der Client eine TCP-Session zum Server auf. TCP nutzt dazu den Dreibege-Handshake (three-way handshake). Hierfür tauschen Client und Server drei TCP-Segmente mit bestimmten Flags aus.



Der Client sendet zuerst ein Segment mit gesetztem SYN-Flag (synchronisieren) an den Server. Dieses Datenpaket enthält auch die Portnummer des gewünschten Dienstes auf dem Server. Akzeptiert der Server den Verbindungsaufbau, antwortet er mit einem gesetzten SYN- und ACK-Flag (Acknowledgement). Der Client bestätigt mit einem ACK-Segment. Bietet der Server auf dem angesprochenen Port keinen Dienst an, gibt es mehrere Möglichkeiten der Reaktion. Ein RST-Segment (RESET) könnte an den Client gesendet werden, der Server könnte eine ICMP-Meldung versenden oder er tut einfach nichts.

TCP connect-Scan -sT

Mit der Option -sT führt Nmap einen connect-Scan aus. Der Scanner verhält sich dabei wie ein normaler Client und verbindet sich (connect) mit dem Zielsystem. Ist der Port auf dem Zielhost im Status LISTENING, wird ein Three-way Handshake ausgeführt. Eine solche Verbindung kann vom Zielhost leicht protokolliert werden und auch jedes IDS sollte einen connect-Scan erkennen.

TCP SYN-Scan -sS

Die Option -sS veranlasst Nmap einen SYN-Scan auszuführen. Auch dabei

beginnt der Scanner mit einem SYN-Segment. Es wird jedoch keine vollständige TCP-Session aufgebaut.

Ist der Port auf dem Zielhost im LISTENING und antwortet mit einem SYN/ACK, sendet Nmap sofort ein RST-Segment (RESET). Dadurch wird die halboffene (half-open) Verbindung wieder abgebaut.

Mit der SYN-Scan Technik kann unter Umständen eine Erkennung des Portscans durch das Target vermieden werden.

Stealth Scans

Mit den Optionen -sF wird ein FIN-Scan ausgeführt. Nmap sendet dazu FIN-Segmente an das Target. Diese werden eigentlich zum Beenden einer TCP-Sitzung genutzt. Das Zielsystem sollte an geschlossenen Ports mit einem RST reagieren und auf listening (abhören) Ports das FIN-Segment einfach verwerfen.

Beim Xmas-Tree-Scan (Option -sX) sind in einem Segment das FIN, URG und PSH-Flag gesetzt. Je nach Implementierung reagieren die Zielsysteme recht verschieden auf ein solches Segment.

Der Null-Scan wird über die Option -sN aktiviert. Nmap sendet Segmente ohne gesetzte Flags an das Target.

Wie bei allen Scan-Techniken können die Resultate durch Firewalls und Filter zwischen Nmap und Target beeinflusst werden. Auf jeden Fall sollte der Admin das Verhalten seiner eigenen Systeme bei diesen Portscans kennen.

UDP Portscans -sU

UDP ist das verbindungslose Protokoll der TCP/IP-Suite. Normalerweise sollte ein System auf ein UDP-Segment an einen geschlossenen Port mit einem "ICMP Port Unreachable" reagieren.

Offene UDP-Ports sollten keine Reaktion erzeugen. Heute wird allerdings ICMP oft rigoros gefiltert. Dadurch sind UDP-Scans mitunter nicht sehr zuverlässig.

Viele Betriebssysteme begrenzen die Rate von ICMP-Paketen. Dadurch sind UDP-Scans bei vielen Systemen sehr langsam.

Allgemeine Optionen

Die Arbeitsweise von Nmap kann durch viele weitere Optionen beeinflusst werden.

Kein Ping -P0

Mit der Option -P0 oder -PN wird ein ICMP-Ping vor dem eigentlichen

Scan unterdrückt. Das kann notwendig sein, wenn eine Firewall oder das Target ICMP filtert.

Operating System - Fingerprint -O

Jede Implementierung von TCP/IP hat ihre Eigenheiten. Mit der Option -O versucht Nmap diese Eigenheiten zu erkennen und das Betriebssystem (Operating System) des Targets zu identifizieren.

```
[root] ~ $ nmap -sT -O m0n0
```

```
Starting nmap 3.20 ( www.insecure.org/nmap/ ) at 2005-05-01 17:59 CEST
```

```
Interesting ports on m0n0 (10.0.0.1):
```

```
(The 1610 ports scanned but not shown below are in state: closed)
```

Port	State	Service
80/tcp	open	http

```
Remote operating system guess: FreeBSD 4.7-RELEASE
```

```
Uptime 30.096 days (since Fri Apr 1 15:41:28 2005)
```

```
Nmap run completed -- 1 IP address (1 host up) scanned in 17.254 seconds
```

Im gezeigten Beispiel erkennt Nmap auf dem Zielsystem ein FreeBSD 4.7.

Fragmentierung -f

Nmap kann zum Beispiel beim SYN-Scan mit fragmentierten IP-Paketen arbeiten. Mitunter lassen sich dadurch Firewalls oder Intrusion Detection Systeme (IDS) verwirren.

Verbose -v

Mit -v können Sie Nmap etwas gesprächiger machen. Es werden Meldungen über die aktuellen Aktivitäten angezeigt.

Port-Bereiche -p

Um bestimmte Ports oder Bereiche zu untersuchen, können diese mit der Option -p bestimmt werden. Einzelne Ports werden als -p 22 notiert, Bereiche können mit -p 40-45, -p 22,23,100-120, -p 80,443,8080,8443 angegeben werden.

Quell-IP-Adresse -S und Interface -e

Unter bestimmten Umständen ist es notwendig Nmap eine Absender-IP-Adresse und eine Schnittstelle mitzuteilen. Das kann zum Beispiel auf einem virtuellen Server hilfreich sein. Das erste Beispiel zeigt einen Aufruf von Nmap auf einem vServer:

```
root ~ $ nmap -sT work.example.org
```

```
Starting nmap V. 2.54BETA31 ( www.insecure.org/nmap/ )  
Failed to lookup device subnet/netmask: eth0: no IPv4  
address assigned  
QUITTING!
```

Eigentlich nutzt dieser vServer das Interface eth0:vs4 mit der IP-Adresse 10.0.0.4. Der Aufruf von Nmap könnte so aussehen:

```
root ~ $ nmap -sT -e eth0:vs4 -S 10.0.0.4 work.example.org
```

```
Starting nmap V. 2.54BETA31 ( www.insecure.org/nmap/ )  
WARNING: If -S is being used to fake your source address,  
you may also have to use -e and -P0.
```

```
WARNING: -S will not affect the source address used in a  
connect() scan.  
(The 1549 ports scanned but not shown below are in state:  
closed)
```

Port	State	Service
22/tcp	open	ssh
80/tcp	open	http

```
Nmap run completed -- 1 IP address (1 host up) scanned in 0  
seconds
```

Timing -T Paranoid|Sneaky|Polite|Normal|Aggressive|Insane

Über die Option -T kann das Timing von Nmap verändert werden. Mit -T Paranoid wartet Nmap bis zu 5 Minuten zwischen den einzelnen Paketen. Dadurch soll die Entdeckung durch ein IDS erschwert werden. Bei »Insane« wird das Target praktisch mit Paketen bombardiert.

Angabe des Zielsystems

Der Scanner Nmap bietet vielfältige Möglichkeiten zur Angabe des Zielsystems. Es können einzelne System über Hostname oder IP-Adresse oder ganze IP-Netze abgescannt werden. IP-Netze können in CIDR-Notation (z.B. /24) oder mit Sternchen (z.B. 192.168.*.*) angegeben werden.

Beispiele:

Im Folgenden sehen Sie einige Beispiele für den Einsatz von Nmap. Sie sollten nur eigene Systeme und Netze mit Nmap scannen. Admins mögen exzessive Portscans auf ihre Systeme nicht besonders gerne.

// Hier werden alle Adressen von 10.0.0.0 und 10.0.0.255 mit einem FIN-Scan überprüft

[root] ~ \$ nmap -sF -T insane 10.0.0.0/24

Starting nmap 3.20 (www.insecure.org/nmap/) at 2005-05-01 18:49 CEST

Host 10.0.0.0 seems to be a subnet broadcast address (returned 1 extra pings). Skipping host.

All 1611 scanned ports on m0n0 (10.0.0.1) are: filtered

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0

Interesting ports on wrt54g (10.0.0.55):

(The 1610 ports scanned but not shown below are in state: closed)

Port	State	Service
80/tcp	open	http

Interesting ports on 10.0.0.101:

(The 1610 ports scanned but not shown below are in state: closed)

Port	State	Service
6000/tcp	open	X11

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0

All 1611 scanned ports on 10.0.0.120 are: closed

Host 10.0.0.255 seems to be a subnet broadcast address (returned 1 extra pings). Skipping host.

Nmap run completed -- 256 IP addresses (4 hosts up) scanned in 238.816 seconds

// SYN Stealth Scan auf eine m0n0wall (IP 10.0.0.1) mit Erkennung des Betriebssystems

[root] ~ \$ nmap -sS -O -v -T insane 10.0.0.1

Starting nmap 3.20 (www.insecure.org/nmap/) at 2005-05-01 18:58 CEST

Host m0n0 (10.0.0.1) appears to be up ... good.

Initiating SYN Stealth Scan against m0n0 (10.0.0.1) at 18:58

Adding open port 80/tcp

The SYN Stealth Scan took 148 seconds to scan 1611 ports.
For OSScan assuming that port 80 is open and port 1 is
closed and neither are firewalled

Insufficient responses for TCP sequencing (2), OS detection
may be less accurate

Interesting ports on m0n0 (10.0.0.1):

(The 1610 ports scanned but not shown below are in state:
closed)

Port	State	Service
80/tcp	open	http

Remote operating system guess: FreeBSD 4.7-RELEASE

Uptime 30.139 days (since Fri Apr 1 15:41:28 2005)

Nmap run completed -- 1 IP address (1 host up) scanned in
154.481 seconds

Verbergen der eigenen IP-Adresse

Des weiteren kann man noch eine sogenannte »decoy«-Option (-D)
aktivieren - sie verhindert, dass die Gegenseite erkennt, welcher Host den
Scan nun tatsächlich initiiert hat. Durch die Angabe der Option -D
decoy1.host.com,ME,decoy2.host.com »fälscht« nmap Pakete mit den
Absendeadressen von decoy1.host.com und decoy2.host.com.

Optional kann durch die Angabe von 'ME' (dt. mich) die eigene Position in
der Zugriffsreihenfolge gewählt werden. Falls 'ME' in die sechste oder noch
eine spätere Position gesetzt wird, sind einige Portscan-Detektoren nicht in
der Lage, die richtige IP-Adresse anzuzeigen. Falls Sie 'ME' nicht
mitangeben, wird nmap eine zufällige Position bestimmen.

Sind sowohl decoy1.host.com als auch decoy2.host.com »up«, schicken
diese ihrerseits, wie erwartet, RST-Pakete, sodass für die Zielmaschine
decoy1.host.com, decoy2.host.com und der lokale Host ununterscheidbar
sind. Sind die decoy-Hosts »down«, so wird das Ziel unseres Scans mit
SYN-Paketen überflutet.

Im Standardmodus benutzt nmap einen ICMP-Ping und einen TCP-ACK-
Ping mit Quellport Port 80 (Port 80 wird oft wegen HTTP-Requests durch
Firewalls durchgelassen), um zu bestimmen, ob Maschinen »up« sind.
Danach wird der Portscan durchgeführt. Als letztes wird versucht, das
Betriebssystem des gescannten Hosts zu bestimmen.

Ganz offensichtlich kann man für ein Intrusion Detection System (IDS) sehr
leicht eine Regel schreiben, die einen nmap-Scan sicher erkennen kann.
Daher kann man zum Beispiel mit -P0 den ICMP-Ping deaktivieren;

explizites Aktivieren des TCP-Pings geschieht durch -PT.

Beispiele:

```
nmap -r -iR -I -sT -p 53 > named.scan.out &  
tail -f named.scan.out
```

Hiermit gehen wir auf die Suche nach Maschinen, auf denen der named als root läuft. Die Option -r scannt die Ports der Zielmaschine in einer zufälligen Reihenfolge, -iR wählt zufällige IP's als Ziel der Scans aus, -I aktiviert den reverse ident scan, der nur mit einem TCP-connect()-Scan (-sT) funktioniert. -p 53 definiert schließlich Port 53 als Scanziel.

Im zweiten Beispiel wollen wir target.host scannen, dabei aber einer allzu leichten Entdeckung entgehen. Wir benutzen daher einige Decoy Hosts:

```
nmap -r -P0 -sS -D decoy1,decoy2,decoy3,decoy4,decoy5 target.host
```

Dabei sollten die Hosts decoy1 bis decoy5 existieren und erreichbar beziehungsweise »up« sein. Die Option -P0 deaktiviert das ping-en von target.host vor dem Scan - wir gehen davon aus, dass er »up« ist. -sS aktiviert den SYN-Scan und -D decoy1,decoy2,decoy3,decoy4,decoy5 nutzt die Hosts decoy1 bis decoy5 um ein wenig Verwirrung zu stiften.

Fazit

Nmap bietet die Möglichkeit eigene Systeme auf ihre Sicherheit und Konfigurationsfehler hin zu untersuchen. Natürlich wird auch die Gegenseite nmap für ihre Zwecke nutzen. Jeder Admin sollte die Reaktion seiner eigenen Systeme auf die verschiedenen Scan-Typen kennen.

Hinweis: Portscans auf fremde Rechner gelten zumindest als unfreundliche Aktionen. Manche Provider ahnden übermäßige Scanaktivitäten mit dem Trennen der Verbindung.

siehe auch: nmap -h

nice

BEFEHL mit festgelegter Priorität ausführen. Ohne BEFEHL, die aktuelle Priorität ausgeben. Der Bereich reicht von **-20** (höchste Priorität) bis **19** (niedrigste).

nice [OPTION] [BEFEHL [ARGUMENT]]...

-n, --adjustment=PRIO ... Priorität zunächst um PRIO erhöhen, für normale Benutzer sind nur positive ganze Zahlen zulässig

nice -n 19 find -name test.txt ... der Befehl wird mit der niedrigsten Priorität ausgeführt, dadurch werden andere Prozesse nicht unnötig verlangsamt

Die Priorität erhöhen (z.B. **nice -n -20 find -name test.txt**), können nur privilegierte Benutzer mit root-Rechten.

top ... zeigt unter anderen auch den aktuellen nice-Wert laufender Prozesse an

ps ax -l ... zeigt ebenfalls den aktuellen nice-Wert laufender Prozesse an

siehe auch: man nice

NUM-Lock beim Start aktivieren

Per Vorgabe ist der Tastenblock beim Systemstart nicht aktiviert. Um dies zu ändern ist ein zusätzliches Programmpaket zu installieren.

```
sudo apt-get install numlockx
sudo cp /etc/gdm/Init/Default /etc/X11/gdm/Init/Default_backup
sudo gedit /etc/gdm/Init/Default
```

Finde diese Zeile in der Datei /etc/gdm/Init/Default

```
...
exit 0
```

und füge nachfolgende Zeilen darüber ein:

```
# Numlock beim Start aktivieren
if [ -x /usr/bin/numlockx ]; then
    /usr/bin/numlockx on
fi
```

Von nun an, ist der Tastenblock auf der rechten Seite der Tastatur beim Systemstart bereits aktiviert.

siehe auch: INTERNET



odt2txt

Mit odt2txt kann man Text aus OpenDocument- Texten (OpenOffice, KOffice, StarOffice etc.) extrahieren.

Das Programm odt2txt kann auch Text aus einigen anderen Formaten extrahieren, die einen ähnlichen Aufbau wie OpenDocument haben. In geringerem Umfang kann odt2txt auch nützlich sein, um Inhalte aus OpenOffice-Tabellen und -Präsentationen zu gewinnen.

odt2txt [options] filename

odt2txt opendocument.odt ... Bildschirmausgabe des Textinhaltes

odt2txt --raw opendocument.odt ... Bildschirmausgabe des Textinhaltes im XML-Format

odt2txt --output=opendocument.txt opendocument.odt ... Text in die Datei opendocument.txt schreiben

odt2txt --width=80 --output=opendocument.txt opendocument.odt ...
Text in die Datei opendocument.txt schreiben; Zeilenumbruch nach 80 Zeichen (default: 65, kein Zeilenumbruch: -1)

siehe auch: man odt2text

Ogg Vorbis, OPUS

siehe auch: Sound

p7zip bzw. 7zr

Mit p7zip kann man auch unter Linux die mit 7z (Windows- und MAC-Version) gepackten Archive (LZMA-Algorithmus) entpacken. Dies gilt selbst für 7z-Archive die mit einem Passwort geschützt sind.

Die allgemeine Syntax von p7zip lautet:

```
p7zip [-d] [-h|--help] [file]
```

p7zip -d archiv.7z ... entpackt das 7z-Archiv im aktuellen Verzeichnis;
Achtung: das archiv.7z wird nach dem Entpacken automatisch gelöscht

Wird p7zip mit 7zr aufgerufen, so können auch 7z-kompatible Archive erstellt werden.

Die allgemeine Syntax von 7zr lautet:

7zr OPTIONEN SCHALTER AUSGABE EINGABE

Beispiele:

7zr a directory.7z directory ... komprimiert das Verzeichnis directory mit den Vorgabewerte von 7zr

7zr a -t7z -m0=LZMA -mmt=on -mx=9 -md=96m -mfb=256 rezept.7z rezept.doc ... komprimiert die Datei rezept.doc

7zr a -t7z -m0=LZMA -mmt=on -mx=9 -md=96m -mfb=256 archiv2.7z directory2 ... komprimiert das Verzeichnis directory2

7zr x archive.7z ... das Archiv archive.7z wird entpackt

7z erkennt Verzeichnisse (rekursives arbeiten) selbstständig, es muss kein spezieller Parameter angegeben werden.

Optionen werden immer als Buchstaben am Anfang angegeben, während die darauf folgenden Schalter immer ein vorstehendes - haben.

Optionen:

a ... Dateien/Verzeichnisse einem Archiv hinzufügen bzw. eine Archivdatei erstellen

x ... Archiv entpacken und dabei die Verzeichnisstruktur erhalten

I ... Inhalt eines Archivs auflisten

Schalter:

-t7z ... Erstellung von 7z-Archive

-m ... Algorithmus für die Kompression festlegen (z.B. -m0=LZMA)

-mmt=on ... Multithreading für Mehrkernprozessoren oder Multiprozessorsysteme aktivieren (manuelle Festlegung der zu nutzenden Kerne mit -mmt=X)

-mx=9 ... Stufe der Kompressionsstärke (hier 9); 0=Speichern, 1=schnell und schlecht ... 9=langsam und gut

-md=96m ... Größe der Wörterbücher (hier 64 MiB); 64k, 1m, 2m, 4m, 6m, 8m, 12m, 16m, 24m, 32m, 48m, 64m, 96m, 128m

-mfb=256 ... Anzahl der Wörterbücher (hier 64); 8, 12, 16, 24, 32, 48, 64, 96, 128, 192, 256, 273

Hinweis: Je höher die anstehenden Nummern bei den Schalter **-mfb** und **-md** sind, desto stärker ist die Kompression. Dies wirkt sich allerdings nicht nur auf die benötigte Zeit des Vorgangs aus, sondern auch auf die Systemauslastung, insbesondere die des Arbeitsspeichers. Es sei nochmals darauf hingewiesen, dass der Schalter **-md** nicht die maximale Beanspruchung des Arbeitsspeichers definiert.

siehe auch: man 7zr, tar, gzip, unzip, unrar

pdftk

Mit dem Programm **pdftk** können Sie PDF-Dateien in die einzelnen Seiten zerlegen und wieder nach Ihren Wünschen zusammenfügen.

Hinweis: In der nachfolgenden Version (Linux Mint 19) wurden die Programmpakete von **pdftk** entfernt. In der Version Linux Mint 21 wurden die Programmpakete von **pdftk** wieder hinzugefügt. Statt **pdftk** können die grafische Programm **pdfarranger** oder **pdfsam** genutzt werden.

Die Funktionalität von **pdftk** umfasst:

- das Zusammenfügen und Teilen von PDF-Dokumenten
- das Drehen von PDF-Dokumenten
- das Einbinden von Wasserzeichen in PDF-Dokumente
- das Abfragen und Aktualisieren der Metadaten einer PDF-Datei
- das Ausfüllen von PDF-Formularen
- die Ver- und Entschlüsselung von PDF-Dokumenten
- die Komprimierung und Dekomprimierung von PDF-Dokumenten
- das Anfügen von Dateien als Anlage an PDF-Dokumente

Alternative Programme zur Bearbeitung von PDF-Dateien sind **PDF Chain**, **PDF-Arranger** (jeweils mit einer grafischen Oberfläche) und **qpdf** einem weiteren Terminalprogramm, die alle in den offiziellen Paketquellen enthalten sind.

```
pdftk <EINGABEDATEI> <OPERATION> <OPTION> output  
<AUSGABEDATEI> <PASSWORT> <RECHTEOPTION>
```

Pdftk-Operationen:

attach_files ... Anhängen von Dateien an ein PDF-Dokument

background ... Einfügen eines Wasserzeichens

burst ... Zerlegung eines PDF-Dokuments in Einzelseiten

cat ... Zusammenfügung von Einzeldokumenten zu einem neuen Gesamtdokument

dump_data ... Ausgabe von Informationen zum PDF-Dokument auf der Standardausgabe

dump_data_fields ... Ausgabe von Informationen zu Formularfeldern eines PDF-Dokument auf der Standardausgabe

fill_form ... Ausfüllen von PDF-Formularen (oder Verbindung von Formulardaten mit Dokument)

update_info ... Aktualisierung der Metadaten (wie Titel, Autor, Datum der Erstellung) einer PDF-Datei

Weitere Optionen sind der Manpage zu entnehmen.

Pdftk-Rechteoptionen:

AllFeatures ... Nutzer hat alle Rechte zur Änderung

Assembly ... Zusammenfügen mit anderen Dokumenten erlaubt

CopyContents ... Kopieren von Textpassagen und Bildern aus dem Dokument erlaubt (Option ScreenReaders ist hiermit auch abgedeckt)

DegradedPrinting ... Drucken nur in eingeschränkter Qualität erlaubt

FillIn ... Formular-Ausfüllung erlaubt

ModifyAnnotations ... Änderung der Anmerkungen erlaubt

ModifyContents ... Änderung der Dokumenteninhalte erlaubt

Printing ... Drucken in bestmöglicher Qualität erlaubt

ScreenReaders ... Textzugriff für Screenreader (d.h. Sprachausgabeprogramme)

PDF-Dokumente zerlegen

Mit der Operation **burst** zerlegen Sie die PDF-Datei in einzelne Seite. Die Seiten werden durchnummeriert und erhalten eine dreistellige Seitenzahl

(Beispiel001.pdf, Beispiel002.pdf ... Beispiel099.pdf). Mit %2d erhalten die einzelnen Seiten eine zweistellige Seitenzahl.

pdftk Beispiel.pdf burst output Beispiel%3d.pdf

In diesem Beispiel legt pdftk die Seiten in das existierende Verzeichnis ./Seiten ab.

pdftk Beispiel.pdf burst output ./Seiten/Beispiel%3d.pdf

PDF-Dokumente zusammenfügen

Die Operation cat fasst mehrere PDF-Dateien zu einem neuen Gesamtdokument zusammen. Sie können dabei die Dateinamen der einzelnen Quelldateien auch mit Jokerzeichen (*) angeben.

pdftk Beispiel.pdf Deckblatt.pdf Haupttext.pdf Anhang.pdf cat output Beispiel_Gesamt.pdf

Mit pdftk können Sie auch Teile einer PDF-Datei (z.B. Haupttext.pdf - Seite 1-4) verwenden, ohne sie vorher in ihre Einzelseiten zerlegen zu müssen.

pdftk D=Deckblatt.pdf B=Haupttext.pdf cat D B1-4 output Beispiel.pdf

Mit dem folgenden Befehl werden die Seiten 1 bis 7 des Dokuments datei1.pdf und die Seiten 1 bis 5 des Dokuments datei2.pdf sowie anschließend noch die 8. Seite von datei1.pdf zur Datei zusammen.pdf zusammengefügt.

pdftk A=datei1.pdf B=datei2.pdf cat A1-7 B1-5 A8 output zusammen.pdf

Hinweis: Sogenannte »Handles«, hier A und B, erlauben es, mehrere Arbeitsschritte in einem Befehl zu verknüpfen. Im obigen Beispiel ist das eine Extraktion (der Seiten) und die anschließende Verknüpfung der extrahierten Seiten.

Seiten aus einem PDF-Dokument entfernen

Mit folgendem Befehl werden die erste Seite eines Dokuments datei.pdf sowie alle Seiten ab der 11. Seite mit Ausnahme der 15. Seite entfernt und die verbleibenden Seiten (Seiten 2 bis 10 sowie 15 der Ursprungsdatei) in der Datei Dokument.pdf gespeichert:

pdftk datei.pdf cat 2-10 15 output Dokument.pdf

Praktisch gibt man also an, welche Seiten im Dokument verbleiben sollen.

Beispiel: Inhaltsverzeichnis und Stichwortverzeichnis in einer separaten Datei speichern

pdftk Linux_Kurzreferenz.pdf cat 1 7-8 625-636 output Inhalts-und-Stichwortverzeichnis.pdf

Das Deckblatt, das Inhaltsverzeichnis (Seite 7-8) und das Stichwortverzeichnis (Seite 625-636) der Datei Linux_Kurzreferenz.pdf werden in der neuen Datei Inhalts-und-Stichwortverzeichnis.pdf gespeichert. Die Datei Linux_Kurzreferenz.pdf wird **nicht** verändert.

Seiten in einem PDF-Dokument vertauschen

Mit folgendem Befehl wird die Position der Seiten 7 bis 8 eines Dokuments datei.pdf mit derjenigen der Seiten 15 bis 16 getauscht und das neue Dokument als Datei Dokument.pdf gespeichert:

pdftk datei.pdf cat 1-6 15-16 9-14 7-8 17-end output Dokument.pdf

PDF-Dokumente drehen

Um 90 Grad drehen

Mit dem folgenden Befehl wird das Dokument datei.pdf um 90 Grad gedreht und als dokument90.pdf gespeichert:

pdftk datei.pdf cat 1-endE output dokument90.pdf

Hinter dem cat wird angegeben, welche Seiten gedreht werden sollen. Im Beispiel sind das die Seiten 1-end, also alle. Das E zeigt die Drehrichtung an: es steht für E wie „east“ (Osten), also 90 Grad im Uhrzeigersinn (dementsprechend steht S für „south“, also 180 Grad, und W für „west“, also 270 Grad). Seit Version 1.45 von pdftk muss die Drehrichtung ausgeschrieben werden.

pdftk datei.pdf cat 1-endeast output dokument90.pdf

Um 180 Grad drehen

Mit dem folgenden Befehl wird das Dokument datei.pdf um 180 Grad gedreht und als dokument180.pdf gespeichert:

pdftk datei.pdf cat 1-endS output dokument180.pdf

bzw.

pdftk datei.pdf cat 1-endsouth output dokument180.pdf

Um 270 Grad drehen

Mit dem folgenden Befehl wird das Dokument datei.pdf um 270 Grad gedreht und als dokument270.pdf gespeichert:

pdftk datei.pdf cat 1-endW output dokument270.pdf

bzw.

pdftk datei.pdf cat 1-endwest output dokument270.pdf

Ausgewählte Seiten eines Dokuments drehen

Mit dem folgenden Befehl werden nur die Seiten 7 bis 10 des Dokuments datei.pdf um 180 Grad gedreht und als dokument180_7-10.pdf gespeichert:

pdftk datei.pdf cat 1-6 7-10S 11-end output dokument180_7-10.pdf.pdf

bzw.

pdftk datei.pdf cat 1-6 7-10south 11-end output dokument180_7-10.pdf.pdf

Die Seiten 1 bis 6 und 11 bis Ende werden nicht gedreht, müssen aber – ohne Drehoption – angegeben werden, da sie sonst nicht in das Enddokument aufgenommen werden.

Verschlüsselung von PDF-Dokumenten

Eine Datei dokument.pdf wird mit dem folgenden Befehl verschlüsselt. Es werden im Beispiel sowohl ein Passwort für den Rechteinhaber als auch ein Benutzerpasswort vergeben:

**pdftk dokument.pdf output verschluesselt.pdf owner _pw
PASSWORD_INHABER user _pw PASSWORD_NUTZER
encrypt_128bit**

Mit den am Schluss des Befehls angehängten Optionen encrypt_40bit oder encrypt_128bit lässt sich zusätzlich die Verschlüsselungsstärke bestimmen.

Nutzerrechte ändern

Für PDF-Dateien können bestimmte Rechteoptionen gesetzt werden. Im folgenden Beispiel ist dem Nutzer das Verändern der Anmerkungen sowie das Drucken der Datei erlaubt, Letzteres allerdings nur in eingeschränkter Qualität:

```
pdftk datei_alt.pdf output datei_neu.pdf owner_pw  
PASSWORD_INHABER user_pw PASSWORD_NUTZER allow  
DegradedPrinting ModifyAnnotations
```

Wasserzeichen einfügen

Um ein Wasserzeichen einzufügen, muss die Datei, die als Wasserzeichen dienen soll, im PDF-Format vorliegen (idealerweise im gleichen Format wie das Zieldokument). Wenn die Wasserzeichen-Datei nicht im gleichen Format vorliegt, skaliert sie pdftk. Mit dem folgenden Befehl wird die Datei wasserzeichen.pdf als Wasserzeichen in die Datei beispiel.pdf eingefügt und die so produzierte Datei als dokument.pdf ausgegeben:

```
pdftk beispiel.pdf background wasserzeichen.pdf output dokument.pdf
```

Formulare ausfüllen

Mit Hilfe einer Datei im FDF-Format lassen sich Formulare in PDF-Dokumenten ausfüllen.

Mit Hilfe des folgenden Befehls wird aus dem Dokument Formular.pdf eine FDF-Datei formulardaten.fdf erzeugt:

```
pdftk Formular.pdf generate_fdf output formulardaten.fdf
```

Diese Datei kann nun in einem Editor manuell editiert werden; T bezeichnet dabei den Titel, V den Wert eines Formularfeldes. Liegt eine modifizierte FSF-Datei vor (im Beispiel: formulardaten.fdf), kann sie mit folgendem Befehl mit dem Dokument Formular.pdf verbunden werden, so dass sich die ausgefüllte Datei Dokument.pdf ergibt:

```
pdftk Formular.pdf fill_form formulardaten.fdf output Dokument.pdf
```

Mit der Option flatten wird ein Formular erzeugt, das nicht mehr editierbar ist:

```
pdftk Formular.pdf fill_form formulardaten.fdf output Dokument.pdf  
flatten
```

Metadaten anzeigen und editieren

Die meisten PDF-Dateien enthalten Meta-Informationen, z.B. zum Autor, zum Thema oder zu der zum Erstellen verwendeten Software. Im Beispiel werden die Meta-Informationen in eine Textdatei gespeichert.

pdftk Beispiel.pdf dump_data output Info.txt

Das ergibt folgenden Inhalt der Datei **info.txt**:

cat info.txt

```
InfoKey: Creator
InfoValue: LaTeX with beamer class version 3.06
InfoKey: Title
InfoValue: LaTeX slides with beamer - So much better than
PowerPoint
InfoKey: Producer
InfoValue: pdfTeX-1.304
InfoKey: Author
InfoValue: Sylvia Blaho
InfoKey: PTEX.Fullbanner
InfoValue: This is pdfTeX, Version 3.141592-1.30.4-2.2
(Web2C 7.5.5) kpathsea version 3.5.5
InfoKey: CreationDate
InfoValue: D:20061103083529+01'00'
(...)
```

Die so erzeugte Datei info.txt kann nun in einem Editor manuell verändert werden. Anschließend können die Änderungen durch folgenden Befehl in das Dokument Beispiel.pdf übertragen werden:

pdftk Beispiel.pdf update_info info.txt output Beispiel_2.pdf

siehe auch: man pdftk, Skripte → Metadaten aus multiplen PDFs entfernen

pdfinfo

Wie der Name vermuten lässt, erhält man mit pdfinfo Informationen zum PDF-Dokumenten. Das Programm **pdfinfo** ist Bestandteil des Programmpaketes **poppler-utils**.

pdfinfo dokument.pdf ... zeigt Informationen zum benannten PDF-Dokument

Ausgabe:

```
Creator:      cairo 1.8.6 (http://cairographics.org)
Producer:     John Doe
```

```

Tagged:      no
Pages:       3
Encrypted:    no
Page size:    612 x 792 pts (letter)
File size:    542783 bytes
Optimized:    no
PDF version:  1.4

```

Man sieht also, womit und wer das PDF-Dokument erstellt wurde, wie viele Seite enthalten sind, ob das PDF verschlüsselt ist, welche Seitengröße es hat, wie groß die ganze Datei ist, ob es sich um ein optimiertes PDF handelt und welche PDF-Version das Dokument hat.

Besonders die Information, ob das Dokument verschlüsselt ist, kann von Interesse sein, da man aus verschlüsselten PDFs in der Regel keine Texte/Bilder extrahieren kann. Die Angabe zur PDF-Version kann nützlich sein, da nicht alle PDF-Betrachter mit allen PDF-Versionen umgehen können.

Hinweis: Ist die PDF-Datei verschlüsselt, so funktionieren Operationen wie Text extrahieren in der Regel nicht.

siehe auch: Skripte → Metadaten aus multiplen PDFs entfernen, `pdftk`, `pdftotext`, `pdftohtml`, `pdfimages`, `exiftool`

pdffonts

Mit Hilfe dieses Befehls kann festgestellt werden, welche Schriftarten (Fonts) innerhalb des PDF-Dokuments verwendet werden. Das Programm **pdffonts** ist Bestandteil des Programmpaketes **poppler-utils**.

pdffonts dokument.pdf ... die enthaltenen Schriftarten anzeigen

name	type	emb	sub	uni	object	ID
Verdana	CID TrueType	yes	no	yes	5	0
TrebuchetMS	CID TrueType	yes	no	yes	6	0
VerdanaBold	CID TrueType	yes	no	yes	7	0
DejaVuSerif	CID TrueType	yes	no	yes	8	0

Die Spalte **emb** gibt an, ob die betreffende Schrift in das Dokument eingebettet ist.

siehe auch: `pdfinfo`

pdftotext

Möchte man den kompletten Text aus dem PDF-Dokument extrahieren, so

kann man dafür pdftotext nutzen. Das Programm **pdftotext** ist Bestandteil des Programmpaketes **poppler-utils**.

pdftotext dokument.pdf ausgabe.txt

Hier wird der Text aus **dokument.pdf** in die Datei **ausgabe.txt** geschrieben.

Wie gut das Ergebnis ist, hängt grundlegend davon ab, wie komplex das PDF-Dokument ist, also z.B. ob das Ausgangsdokument einfacher Fließtext ist, viele vom Text umflossene Bilder enthält oder sogar mehrspaltig ist.

In der Regel muss das neu erstellte Textdokument immer nach bearbeitet werden, um z.B. überflüssige Leerzeichen und falsche Zeilenumbrüche zu entfernen.

Optionen:

- f Nr ... beginne auf Seite Nr
- l Nr ... stoppe auf Seite Nr
- htmlmeta ... erzeugt eine (einfache) HTML-Datei
- listenc ... zeigt alle mögliche Encodings (für -enc) an
- enc TYP ... verwendet das Encoding TYP für die Textdatei
- eol TYP ... das Zeilenende in der Ausgabedatei wird auf TYP gesetzt (mögliche Werte: unix, dos, mac)

for i in *.pdf; do pdftotext \$i \$ausgabe.txt; grep "SUCHWORT" \$ausgabe.txt; done ... durchsucht alle PDF-Dateien im aktuellen Verzeichnis und gibt die Zeilen mit dem gefundenen SUCHWORT im Terminal aus; es wird zwischen Groß- und Kleinschreibung unterschieden

for i in *.pdf; do pdftotext \$i tmp_ausgabe.txt; if [\$(grep -c "SUCHWORT" tmp_ausgabe.txt) -gt 0]; then echo "\$i"; fi; done; rm -f tmp_ausgabe.txt ... durchsucht alle PDF-Dateien im aktuellen Verzeichnis und gibt die Dateinamen aus in dem SUCHWORT enthalten ist; es wird zwischen Groß- und Kleinschreibung unterschieden

siehe auch: man pdftotext, pdftohtml, pdffimages

pdftohtml

Durch **pdftohtml** können PDF-Dokumente in HTML-Seiten umwandelt werden. Das Programm **pdftohtml** ist Bestandteil des Programmpaketes **poppler-utils**.

pdftohtml dokument.pdf seite ... die PDF-Datei dokument.pdf wird in eine HTML-Datei seite.html konvertiert

Dabei wird die Datei **dokument.pdf** in eine HTML-Datei **seite.html** umgewandelt. Genau genommen entstehen immer drei HTML-Dateien: Einmal die genannte Hauptdatei plus, in diesem Beispiel, die Seiten **seite_ind.html** und **seiten.html**. Dies liegt daran, dass die HTML-Ausgabe (nicht mehr ganz zeitgemäß) Frames verwendet, um im linken Frame ein einfaches Inhaltsverzeichnis und im Hauptframe den Inhalt an sich darzustellen. Weiterhin wird der Text komplett über die die "klassischen" HTML-Tags formiert anstatt CSS-Styles zu verwenden.

Über die Option **-xml** wird eine XML-Datei anstatt einer HTML-Datei generiert.

Hinweis: In einigen Dateimanagern kann eine PDF-Datei über das Kontextmenü (rechten Maustaste) in eine HTML-Datei konvertiert werden.

siehe auch: pdftotext, pdfimages

pdfimages

Mit Hilfe von pdfimages lassen sich alle Bilder aus einem PDF-Dokument extrahieren. Die Bilder werden dabei als PPM-Datei gespeichert. Das Programm **pdfimages** ist Bestandteil des Programmpaketes **poppler-utils**.

pdfimages dokument.pdf bild

bild ist dabei das Präfix für die Ausgabedateien. Enthält ein PDF z.B. drei Bilder, so werden die Dateien bild-000.ppm, bild-001.ppm und bild-002.ppm generiert.

Hinweis: Über die Option **-j** werden alle im PDF-Dokument enthaltenen JPEG-Bilder auch als JPEG gespeichert und nicht als PPM.

siehe auch: pdfimages -h, man pdfimages, pdftotext, pdftoppm

pdftoppm

Mit pdftoppm können die Seiten eines PDF-Dokuments in Bilddateien umgewandelt werden, standardmäßig sind dies PPM-Dateien, welche von allen gängigen Grafikbetrachtern gelesen (und konvertiert) werden können. Das Programm **pdftoppm** ist Bestandteil des Programmpaketes **poppler-utils**.

pdftoppm dokument.pdf seite

seite ist dabei das Präfix für die Namen der Ausgabedateien. Hat eine PDF-Dokument z.B. drei Seiten, so werden die Dateien seite-000.ppm, seite-001.ppm und seite-002.ppm erzeugt. pdftoppm kennt einige Optionen, u.a. auch für das Ausgabeformat:

- mono** ... Ausgabe als monochrome PBM-Datei
- gray** ... Ausgabe als PGM-Datei (Grauskala)
- png** ... Ausgabe als PNG-Datei

Des Weiteren gibt es noch diverse Optionen, um die Ausgabegröße und Auflösung festzulegen.

siehe auch: pdftoppm -h, man pdftoppm, pdftotext, pdftimages

ping bzw. ping6

ping ist ein Terminalbefehl zum Prüfen der Erreichbarkeit von anderen Rechnern oder Geräten über ein beliebiges Netzwerk. Der »angepingte« Netzwerkteilnehmer beantwortet die kurze Anfrage durch eine ebenso kurze Gegenantwort. Somit ist gezeigt, dass die grundsätzliche Erreichbarkeit der Teilnehmer untereinander gegeben ist. ping nutzt dazu das **ICMP**-Protokoll. Der Befehl ping ist ein grundlegender Netzwerk-Befehl, der auf quasi allen Betriebssystemen verfügbar ist. Auch andere Netzwerkgeräte wie Drucker oder Router antworten im Regelfall auf eine ping-Anfrage.

Hinweis: Ping gibt es in 2 Varianten **ping** und **ping6**. Die Optionen sind für beide Varianten identisch, daher wird im Folgenden immer von ping geredet.

Der Befehl hat die folgende, allgemeine Syntax:

ping [Optionen] IP-Adresse

Anstelle der IP-Adresse kann auch ein Host-Name angegeben werden. Es wird dann versucht, diesen in eine DNS-Adresse aufzulösen.

ping [Optionen] example.com

Über diesen Weg kann man auch recht einfach die DNS-Adresse zu einer Domain herausfinden, da ping diese mit ausgibt. Ruft man ping komplett ohne Optionen auf, läuft der Befehl unendlich lange durch und sendet pro Sekunde eine ping-Anfrage. D.h. man muss ping entweder manuell stoppen

(mit Tastenkombination [Strg] + [C]) oder mit der entsprechenden Option die die Anzahl der gesendeten Pakete begrenzt.

Die IP-Adresse eines Rechners wird auf Linux-PCs mit **ifconfig** und auf Windows-PCs mit **ipconfig** ermittelt.

Optionen:

-c ANZAHL ... ANZAHL legt die Zahl der zu versendenden Pakete fest (z.B. ping -c 4 192.168.100.25)

-w ENDE ... ENDE wird in Sekunden angegeben. ping wird nach dieser Zeit beendet, egal wie viele Anfragen (un-) beantwortet wurden.

-W AUSZEIT ... AUSZEIT wird in Sekunden angegeben und gibt an, wie lange ping auf eine Antwort wartet, bevor es automatisch stoppt

-i INTERVALL ... INTERVALL wird in Sekunden angegeben und gibt vor, in welchen Abständen die ping-Anfragen gesendet werden.

Voreinstellung ist eine Sekunde.

-I SCHNITTSTELLE ... legt fest, über welche Schnittstelle die ping-Anfragen gesendet werden (z.B. eth0, eth1 ...)

Beispiele:

ping -c 4 localhost ... sendet 4 ICMP-Paket an die eigene Netzwerkkarte; antwortet die Netzwerkkarte, so ist wahrscheinlich betriebsbereit

ping6 -c 4 ::1 ... prüft die Ipv6-fähigkeit des eigenen Rechners (::1 ... localhost bzw. 127.0.0.1)

ping6 ipv6.google.com ... prüft, ob man einen IPv6-fähigen Internetzugang besitzt und gleichzeitig die Erreichbarkeit von Google

Zusätzliche Informationen

Ping ist ein Befehl, der Daten an die angegebene Adresse sendet, die dieser Rechner wieder zurücksendet. Dieser Befehl sendet ein Paket von 32 Byte an den angegebenen Rechner im Netzwerk und misst die Zeit, die das gesendete Paket benötigt. Ping muss von jedem Computer im Netz beantwortet werden, da es die absolute Priorität gegenüber anderen Netzwerkdiensten hat.

Schlägt ein Ping fehl, dann sollten Sie ein ping auf die Loopback-Adresse des Rechners (127.0.0.1) starten. Bei Erfolg, ist so zumindest sichergestellt, dass TCP/IP arbeitet und der Fehler wahrscheinlich bei der Konfiguration der IP-Daten zu finden ist bzw. dass es zum angegebenen Rechner keine funktionierende physische Verbindung gibt (z.B. Leitungsdefekt).

Funktioniert ein ping auf das Loopback-Gerät aber nicht, dann ist die Netzwerkkarte defekt oder fehlerhaft installiert.

In der ersten Zeile der Ausgabe nennt ping die zum kontaktierten Server zugehörige IP-Adresse. Danach folgt für jedes geschickte Paket eine Auswertung der Antwort. Im letzten Feld »time« steht die bis zur Rückmeldung verstrichene Zeit in Millisekunden. Daraus lässt sich die Reaktionszeit des Servers ablesen - eine hohe Zahl erklärt beispielsweise einen trägen Zugriff auf einen Webserver.

In der Auswertungszeile von ping stehen die Anzahl der versandten (packets transmitted) und die zurückerhaltenden Pakete (received), der prozentuale Anteil der nicht beantworteten Anfragen (paket loss) und die Dauer des ping-Vorganges (time). Gewöhnlich gehen keine oder nur sehr wenige Pakete verloren, eine hohe Paketverlustrate lässt auf eine gestörte Verbindung schließen. Die nachfolgende Zeile liefert die schnellste, die durchschnittliche und die langsamste Antwort und schließlich die Standardabweichung vom Durchschnitt.

Hinweis: Internet-Rechner sind nicht gezwungen auf ping-Anfragen zu reagieren. Viele Server ignorieren sie, um Bandbreite zu sparen oder um potentielle Angreifer nicht auf ihre Existenz hinzuweisen. Eine Alternative zu ping das nur ICMP-Pakete verwendet ist das Programm **echoping**. echoping kann mit dem entsprechenden Parameter auch UDP-Pakete versenden, die möglicherweise durch eine Firewall nicht gefiltert werden.

Hinweis: Das Paket echoping wurde in Linux-Mint 21 entfernt.

Mit den Kommandozeilentools ifconfig, host, route, ping, traceroute bzw. mtr und netstat diagnostizieren Sie bei Netzwerkstörungen - richtig eingesetzt - gezielt jeden Teilaspekt einer Verbindung, intern oder im Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert - die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So finden Sie möglicherweise heraus, ob es sich bei einer nicht funktionierenden internen oder Internet-Verbindung um ein Problem mit einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS oder eines der Gegenseite handelt.

siehe auch: man ping, host, ifconfig, route, traceroute bzw. mtr und netstat

pgrep

pgrep zeigt die aktuell laufenden Prozesse an.

pgrep [options] [-u Benutzername] [Suchwort, Prozessname]

pgrep -l <Suchwort> ... Anzeige der Prozessnummer und des Prozessnamen in den das Suchwort enthalten ist; Suchwort: z.B. gnome

pgrep -lo <Suchwort> ... Anzeige des ältesten Prozesses - Prozessnummer und Prozessnamen - in den das Suchwort enthalten ist; Suchwort: z.B. gnome

pgrep -ln <Suchwort> ... Anzeige des neuesten Prozesses - Prozessnummer und Prozessnamen - in den das Suchwort enthalten ist; Suchwort: z.B. gnome

pgrep -u <Benutzername> -l <Suchwort> ... Anzeige der Prozesse die von einem Benutzer gestartet wurden - Prozessnummer und Prozessnamen - in den das Suchwort enthalten ist; Suchwort: z.B. smb, Benutzername: root

siehe auch: man pgrep, pkill, ps

pkill

Möchte man den gefundenen Prozessen nun ein Signal senden, so bietet sich das Programm pkill an.

pkill [Signal] [-u Benutzername] [Prozessname]

pkill -HUP top

pkill -9 -u <Benutzername>

Im ersten Beispiel wird allen Instanzen von top sowie allen Programmen mit »top« im Namen das SIGHUP-Signal geschickt, und im zweiten Fall werden alle Prozesse eines Benutzers abgebrochen.

pkill gnome-panel ... beendet das Programm gnome-panel; Gnome startet darauf das Programm erneut; nützlich falls im Gnome-Panel fehlerbehaftete Anzeigen auftauchen

siehe auch: man pkill, pgrep, ps

pwd

Zeigt das aktuelle Verzeichnis (print working directory).

pwgen

»pwgen« generiert zufällige, bedeutungslose Passwörter. Die Passwörter enthalten entweder nur Kleinbuchstaben, Klein- und Großbuchstaben gemischt oder auch Ziffern.

Mit etwas Fantasie sind die mit pwgen erzeugten Passwörter auch aussprechbar. Großbuchstaben und Ziffern werden von pwgen so platziert,

dass es leicht fällt sich ihre Position zu merken, wenn man nur das Wort auswendig lernt. Alternativen zu pwgen sind die Programme apg und gpw.

Beispiel:

pwgen -n 14 ... erzeugt einige Passwörter mit einer Passwortlänge von 14 Zeichen

Ausgabe:

oongeth5fohMah

boog8ri7Ui7Zoh

Oochee7eiSh5ah

[...]

pwgen -n 20 -sy ... erzeugt Passwörter die selbst höchsten Sicherheitsanforderungen genügen; die Argumente stehen für secure (s) und symbols (y)

Ausgabe:

AXOtq`#u4]30mEGj:^\4

G\$[x|^X28zt3~5-|xu\$j

9j_|E>P({!0[~;g%bBl

[...]

siehe auch: man pwgen

ps

ps ohne Optionen zeigt alle eigenen, also von einem selbst gestartete Programme oder Prozesse.

ps r ... zeigt alle Prozesse, die gerade Rechenzeit verwenden

ps -A ... zeigt alle Prozesse an (process status) (Hilfe mit ps --help)

ps -ef | less ... zeigt alle aktiven Prozesse an (mit Leertaste weiter, mit [q] beenden)

ps -ef | grep -i <MeinProgramm> ... prüft, ob ein bestimmtes Programm läuft

ps -ax | grep inetd ... ermittelt die Prozess-ID vom Superdämon inetd

ps -aux | less ... zeigt zusätzliche Informationen (mit Eingabe- oder Leertaste durch die Liste bewegen, mit [q] Anzeigeprogramm less verlassen)

ps x ... In der einfachsten Form geben sie **ps x** ein, daraufhin erscheint eine kompakte Tabelle mit fünf Felder.

pstree ... zeigt die Prozess-Beziehungen an

Hinweis: Die vielfältigen Optionen von ps werden teilweise mit und teilweise ohne dem Minuszeichen eingeleitet. Die Manualpage ist gut geeignet, den potentiellen Benutzer in die Flucht zu schlagen - zum Glück liefert **ps --help** eine kurze Hilfeseite.

PID: Das ist die so genannte Prozess-ID. Jedes gestartete Programm verfügt über eine eindeutige Nummer, anhand derer es identifiziert wird und mit der Sie ihm Anweisungen schicken können z.B. kill 2200 beendet den Prozess 2200 (sehr nützlicher Befehl um ein abgestürztes Programm zu beenden). Sie sollten den Befehl nur dann einsetzen, wenn Sie wissen, was die Anwendung tut, die Sie beenden wollen.

TTY: Dabei handelt es sich um das Terminal, aus dem heraus die Anwendung gestartet wurde. Bei Programmen, die Sie über das K-Menü aufrufen, steht dort ein Fragezeichen.

STAT: Hier wird der Status eines Programms angezeigt. Das R steht für running (laufend). Das bei vielen Anwendungen ein S (für sleeping = schlafend) steht, muss Sie nicht beunruhigen. Es zeigt nur, dass das Programm gerade nichts zu tun hat.

TIME: Dieser Wert steht nicht für die Zeitspanne, wie lange eine Anwendung schon läuft, sondern zeigt die von ihr verbrauchte Rechenzeit an.

COMMAND: Hier steht der Befehl, mit dem das Programm aufgerufen wurde.

* * * * *

passwd

Das Terminalprogramm passwd dient zum Erstellen und Ändern der Passwörter von Benutzerkonten. Ein normaler Benutzer kann nur das Passwort für sein eigenes Konto ändern. root kann auch die Passwörter für andere Benutzer ändern. Wenn ein normaler Benutzer passwd verwendet, wird er zuerst nach dem alten Passwort gefragt und dann nach dem neuen. Wenn root das Passwort eines anderen Benutzers ändert, wird die Abfrage nach dem alten Passwort übergangen. So ist es möglich vergessene Passwörter von normalen Benutzern zu ändern.

Die Datei **/etc/passwd** ist die zentrale Datei für die Benutzerverwaltung. Die verschlüsselten Passwörter werden aber in der Datei **/etc/shadow** gespeichert, die nur vom Systemadministrator root einsehbar ist oder mit

den entsprechenden sudo-Rechten.

Es gibt 3 Klassen von Benutzern: den Benutzer root mit der UID = 0, Systembenutzer und normale Benutzer. Greift ein Benutzer auf eine Datei zu, so prüft das System erst die UID des Benutzers. Wenn die UID = 0 ist, werden keine Rechte geprüft bzw. er kann sich alle Rechte holen.

Als normaler Benutzer, bzw. wenn root sein eigenes Passwort ändern will:

passwd

Ändern des Passworts für ben.
(aktuelles) UNIX-Passwort:

Als root:

passwd [Optionen] [Benutzer]

passwd -e ben ... Der Benutzer ben muss bei nächsten Login sein Passwort ändern.

Optionen zur Gültigkeit des Passworts:

-n min ... Minimale Zeit in Tagen, die vergehen muss bis der Benutzer sein Passwort ändern kann. Wenn hier der Wert 0 angegeben wird, kann der Benutzer sein Passwort jederzeit ändern.

-m max ... Maximale Zeit in Tagen, die angibt wie lange das Passwort nach der Erstellung des Kontos gültig sein wird. Wenn diese Zeit abgelaufen ist, muss der Benutzer sein Passwort ändern damit er sein Konto nutzen kann.

-w warnen ... Hiermit wird die Anzahl der Tage vor dem Ablauf des Kontos festgelegt, ab denen eine Warnung zur Ungültigkeit des Passwortes erfolgt

-i inaktiv ... Hiermit wird die Anzahl der Tage festgelegt, die vergehen bis das Konto gesperrt wird nachdem das Passwort abgelaufen ist. Ein Wert von **-1** setzt diese Eigenschaft außer Kraft.

Optionen für die Administration:

-l ... Hiermit wird das Konto für den spezifizierten Benutzer gesperrt.

-u ... Dient zum Freigeben eines gesperrten Kontos. Dies geht aber nur wenn das Konto ein Passwort hat. Ein Konto das nur "!" als Passwort hat kann nicht freigegeben werden.

-d ... Hiermit wird das Passwort des spezifizierten Benutzers gelöscht.

-S ... Gibt Informationen zum Passwortstatus des spezifizierten Benutzers

aus.

Erläuterung zum Passwortstatus: Die Felder sind durch Leerzeichen getrennt (sudo passwd -S ben).

ben P 08/03/2013 0 99999 7 -1

1. Feld: Benutzername
2. Feld: L = Konto ist gesperrt; NP = ohne Passwort; P = Passwort existiert
3. Feld: Datum der letzten Passwortänderung
4. Feld: min
5. Feld: max
6. Feld: warnen
7. Feld: inaktiv

-aS ... Gibt Informationen zum Passwortstatus aller Konten aus.

-e ... Der Benutzer muss sein Passwort beim nächsten Login ändern.

Beispieleintrag in der Datei /etc/passwd:

Die Datei /etc/passwd enthält neben den normalen Benutzern, auch eine Reihe von Pseudobenenutzer die vom System (Dämonen, Servern, Diensten) benutzt werden. Die einzelnen Felder der Datei passwd sind durch einen Doppelpunkt (:) getrennt.

ben:x:1002:1002:,,,:/home/ben:/bin/bash

1. Feld: Benutzername
2. Feld: Muss sich der Benutzer über ein Passwort einloggen, steht dort ein **x**. Lässt man das Feld leer, hat man ein leeres Passwort gesetzt, d.h. der Benutzer kann sich ohne Passwordeingabe einloggen. **Hinweis:** Die verschlüsselten Passwörter sind aus Sicherheitsgründen in der Datei /etc/shadow (die Datei kann nur mit root-Rechten eingesehen werden) - in verschlüsselter Form – ausgelagert.
3. Feld: numerische User-ID (UID)
4. Feld: numerische Group-ID (GID)
5. Feld: frei wählbare Beschreibung, Kommentar (kann leer sein); enthält i.d.R. den vollständigen Benutzernamen und noch einige persönliche Angaben wie Tel-Nr., Raum-Nr.; die einzelnen Informationen werden jeweils durch ein Komma getrennt
6. Feld: Home-Verzeichnis des Benutzers
7. Feld: Standardshell, Befehlsinterpret des Benutzers, das nach dem

Login gestartet wird

Hinweis: Es gibt einen Unterschied zwischen keinem Passwort und einem leerem Passwort. Ein Konto ohne Passwort ist auf jeden Fall gesperrt. Ein Konto mit leerem Passwort ist abhängig von der Einstellungen in der PAM (/etc/pam.d/passwd bzw. /etc/pam.d/common-password).

Beispieleintrag in der Datei /etc/shadow:

Das Passwort wurde hier mit dem DES-Verfahren verschlüsselt.

ben:Xldlkasoo2bn90lsal:12455:0:99999:7:::

Das verschlüsselte Passwort ist die Zeichenfolge nach dem Benutzernamen ben zwischen den beiden Doppelpunkten [:]. Bei einer DES-Verschlüsselung besteht diese Zeichenfolge aus genau 13 Zeichen. Besteht diese Zeichenfolge aus genau 60 Zeichen, so wird sehr wahrscheinlich die BLOWFISH-Verschlüsselung verwendet und bei 32 bzw. 34 Zeichen die MD5-Verschlüsselung.

1. Feld: Benutzername
2. Feld: Passwort-Hash
3. Feld: Zeitpunkt als das Passwort das letzte Mal geändert wurde, angegeben in Tagen seit dem 1.1.1970 (Beginn der UNIX-Zeit)
4. Feld: Anzahl der Tage, die vergehen müssen, ehe das Passwort erneut geändert werden darf. Steht hier eine Null (0), so darf der Benutzer sein Passwort so oft wie er möchte ändern.
5. Feld: Anzahl der Tage nach deren Ablauf das Passwort geändert werden muss.
6. Feld: Warnzeit in Tagen, bevor das Passwort seine Gültigkeit verliert.
7. Feld: Zeit in Tagen nach dem der Account ungültig wird, nachdem die Zeit für das Passwort abgelaufen ist
8. Feld: Zeitpunkt zu dem der Account ungültig wird, angegeben in Tagen seit dem 1.1.1970 (Beginn der UNIX-Zeit)
9. ein reserviertes Feld

Hinweis: Unter Ubuntu oder den Linux-Distributionen die auf Ubuntu basieren (Linux Mint, Xubuntu, Lubuntu ...) wird das Passwort mit dem SHA-512-Algorithmus verschlüsselt.

Beispiel: das Passwort soll **nostromo** heißen
DES: **2CCu.2JCMimEY**

MD5: \$1\$KzO6eCI2\$TGmtzaJpvbfG3TTtAGBbA0

BLOWFISH:

\$2a\$10\$g4e99hx0AWogZLfNMR789uUd8.VQenw1ndXEYahNO03hD
uabE7J.2

SHA-512:

\$6\$19uKKdas/ehHdq0E\$SiRlxcO1uQcRV6PVvEi2m3IHRI5t4xJB7bgTv
WQH0uRX/m.WRHXXDx2t3GFVqXcFagiW.odTgCBt5UtUGZ03wfl

siehe auch: man passwd, id, useradd, groupadd

Ports

Portnummern sind 16-Bit Zahlen, es gibt also insgesamt 65536 Ports. Die Ports bis einschließlich 1024 sind für festgelegte Dienste (siehe auch: /etc/services) reserviert. Die Ports größer 1024 bis 65536 können im Allgemeinen frei gewählt werden.

Hier einige wichtige Portnummern:

Name des Dienstes	Port-Nr.
echo	7 (Dieser Dienst schickt immer genau die Zeichenketten zurück, die man an ihn sendet.)
discard	9 (Dieser Dienst verschluckt alle Eingaben und sendet nie etwas zurück.)
daytime	13 (Dieser Dienst antwortet mit der genauen Zeit, die auf dem Server gilt, und beendet daraufhin die Verbindung.)
chargen	19 (Dieser Dienst antwortet mit einem endlosen Strom von ASCII-Zeichen, der erst endet, wenn man selber die Verbindung abbricht.)
FTP	20 (Daten), 21 (Kommandos)
Secure Shell (ssh)	22
Telnet	23
Mail	25 (SMTP - senden von Emails)
time	37 (Dieser Dienst liefert ebenfalls die Serverzeit, allerdings in einem maschinenlesbaren Format.)
WHOIS-Server	43
DNS-Server	53
DHCP-Server	67 (Client sendet eine Anfrage), 68 (antwortet auf eine Anfrage eines Clients)
TFTP	69
Finger	79
WWW, HTTP	80
Mail	106 (POP3pw)

Name des Dienstes	Port-Nr.
Mail	110 (POP3 - empfangen von Emails)
Portmapper	111
Samba-Server	137 - netbios-ns; 138 - netbios-dgm; 139 - netbios-ssn; 445 - microsoft-ds
IMAP	143
LDAP	389
HTTPS	443
SSL	465 (SMTP – SSL-Verschlüsselung)
Drucker (CUPS)	631
SWAT	901
SSL	993 (IMAP - SSL-Verschlüsselung)
SSL	995 (POP3 - SSL-Verschlüsselung)
NFS-Server	2049
MySQL	3306
HTTPS	8443
Plesk	8443; www.domain.net:8443
Webmin	10000
Usermin	20000 - Usermin ist praktisch eine Lightversion von Webmin

siehe auch: `man /etc/services`

Portscanner

siehe auch: `nmap`

proc-Dateisystem

Das `/proc` Dateisystem ist ein Pseudo-Dateisystem, in dem der Kernel wichtige Informationen in Form von virtuellen Dateien (keine echten Dateien) vorhält. Diese virtuellen Dateien belegen keinen Speicherplatz, sondern bieten eine direkte Verbindung zum Kernel. Viele Informationen zu Hardware, System und laufenden Prozessen sind hier unkompliziert lesbar, da sich die Dateien hier wie normale Textdateien verhalten. Viele Programme, die Prozessinformationen und Hardware-Infos benötigen, verwerten ebenfalls die Daten aus dem `/proc`-Verzeichnis.

cat /proc/version ... Kernelversion anzeigen

cat /proc/cpuinfo ... Informationen über die CPU

cat /proc/interrupts ... Informationen über die Interrupts

cat /proc/filesystems ... Informationen über die unterstützten Dateisysteme

procinfo ... wichtige Informationen über das `/proc`-Dateisystem; mit der Option `-a` werden alle Informationen gezeigt

cat /proc/pci ... Informationen über PCI-Devices, z.B. Netzwerkkarte

grep "lm" /proc/cpuinfo ... Prozessor-Flags anzeigen: Dieser Befehl zeigt zu jedem CPU-Kern die Kürzel der Befehlserweiterungen an. Das unscheinbare »lm« (hier rot) zeigt, dass der Prozessor die 64-Bit-Erweiterungen unterstützt. Diese zwei Buchstaben stehen für »long mode« und sind das Merkmal von 64-Bit-CPUs, egal ob von Intel oder AMD. Auf einer 32-Bit-CPU bleibt die Ausgabe dagegen leer.

Terminalausgabe: `grep "lm" /proc/cpuinfo`
flags : fpu vme de pse tsc msr pae mce cx8 apic sep
mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse
sse2 ss ht tm pbe syscall nx rdtscp **lm** constant_tsc
arch_perfmon pbs bts rep_good nopl xtopology nonstop_tsc
aperfperf pni pclmulqdq dtes64 monitor ds_cpl est tm2
ssse3 cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic popcnt
tsc_deadline_timer xsave lahf_lm arat epb xsaveopt pln pts
dtherm

siehe auch: Linux-Dateisystem

paste

Fügt jede Zeile einer Datei an die entsprechende Zeile einer anderen Datei an, als Trenner dient ein Tabulator. Anders als bei join gibt es kein gemeinsames Indexfeld.

-d, --delimiters=LISTE ... Zeichen aus LISTE anstelle von Tabulatoren benutzen

paste date1.txt date2.txt

siehe auch: join

Programme finden, installieren und entfernen

Software liefern Linux-Distributionen aus einem Fundus von vorgefertigten Paketen. Von den vielfältigen Paket-Formaten haben das von Debian geschaffene DEB-Format (Debian, Ubuntu, Linux Mint, Xubuntu, Lunbuntu, Bodhi ...) und das von Red Hat entwickelten RPM-Format (Fedora, OpenSuse, CentOS, Mageia) die größte Verbreitung.

DEB-Paketen (Debian, Ubuntu, Linux Mint und weitere):

Der bekannteste Vertreter erweiterte Paketmanager ist das von Debian geschaffene APT.

sudo apt-get update ... Aktualisierung der Liste verfügbarer Programm-

Pakete

apt-cache search [Paketname/Suchbegriff] ... Suche nach Programm-Paketen

sudo apt-get install [Paketname] ... Programm-Paket mit all seine Abhängigkeiten installieren

sudo apt-get remove [Paketname] ... Programm-Paket deinstallieren

sudo apt-get dist-upgrade ... Upgrade des gesamten Linuxsystems auf die neuste Version

siehe auch: apt-get, man apt-get

RPM-Pakete (Fedora, CentOS):

CentOS und Fedora benutzen für das Paketmanagement das Kommandozeilen-Tool yum. Eine separate Aktualisierung der Paketquellen ist nicht nötig, dies erledigt yum bei jedem Aufruf automatisch.

yum search [Paketname/Suchbegriff] ... Suche nach Programm-Paketen

yum install [Paketname] ... Programm-Paket installieren; Root-Rechte erforderlich

yum remove [Paketname] ... Programm-Paket deinstallieren; Root-Rechte erforderlich

yum update ... Upgrade des gesamten Linuxsystems auf die neuste Version

RPM-Pakete (OpenSuse):

Einfaches Paketmanagement war bei Suse lange Zeit nur über YaST möglich. Seit OpenSuse 10.2 gibt es mit zypper ein Kommandozeilen-Tool.

zypper se -d [Paketname/Suchbegriff] ... Suche nach Programm-Paketen

zypper in [Paketname] ... Programm-Paket installieren; Root-Rechte erforderlich

zypper rm [Paketname] ... Programm-Paket deinstallieren; Root-Rechte erforderlich

zypper dup ... Upgrade des gesamten Linuxsystems auf die neuste Version

qrencode

qrencode ist ein von Kentaro Fukuchi entwickeltes Programm, das Zeichenketten in einen QR-Code umwandelt und als PNG-Bild speichert.

Ein QR-Code (QR steht für englisch: quick response = schnelle Antwort) ist ein zweidimensionaler Code (2D-Code), der von der japanischen Firma »Denso Wave« im Jahr 1994 entwickelt wurde.

Mit dem Aufkommen von Smartphones mit integrierter Kamera und entsprechenden Programmen ("Apps") wurde diese Idee 15 Jahre später wieder populär. Davon zeugen auch neue Möglichkeiten wie Android-Trojaner per QR-Code.

qrencode [OPTION] [STRING]

Beispiele:

qrencode -o qrcode.png http://www.example.de ... QR-Code mit dem Dateinamen qrcode.png erstellen, der auf die URL "www.example.de" verweist, Pixelgröße 3 (default)

qrencode -s 12 -o qrcode.png http://www.example.de ... derselbe QR-Code, aber mit einer Pixelgröße von 12

qrencode -l L -v 1 -o output.png 'Hallo Welt!' ... ergibt einen QR-Code mit einem niedrigen „Error Correction Level“ einen QR-Code in Version 1, dem Dateinamen output.png und dem Text "Hallo Welt!"

Optionen:

-o FILENAME oder **--output=FILENAME** ... das PNG-Bild in FILENAME schreiben

-s NUMBER oder **--size=NUMBER** ... die Pixelgröße bestimmen (Standard = 3)

-l {LMQH} oder **--level={LMQH}** ... den „Error Correction Level“ spezifizieren (L = niedrig bis H = hoch; Standard = L)

-v NUMBER oder **--symversion=NUMBER** ... die Version des Symbols spezifizieren (Standard = Auto)

-m NUMBER oder **--margin=NUMBER** ... die Randbreite spezifizieren (Standard = 4)

-S oder **--structured** ... Strukturierte Symbole erstellen; die Version muss angegeben werden

-k oder **--kanji** ... nur nötig, falls der umzuwandelnde Text Kanji enthält
-c oder **--casesensitive** ... Kleinbuchstaben in 8-bit-Modus encodieren
-i oder **--ignorecase** ... Fallunterschiede ignorieren und nur Großbuchstaben verwenden
-8 oder **--8bit** ... Ausgabe in den 8-bit-Modus encodieren (die Optionen -k, -c und -i werden ignoriert)
[STRING] ... wird mittels der Eingabedaten nichts angegeben (z.B. keine Internet-Adresse wie <http://www.example.de>), wird die Standardeingabe benutzt; die Eingabedaten an der Stabdardeingabe werden mit der Tastenkombination **[Strg] + [D]** abgeschlossen

QR-Code kodieren – weitere Möglichkeiten

- für den Browser Firefox gibt es ein Add-on: Mobile Barcoder
- Online: Zeitschrift Chip.de - http://www.chip.de/news/Scannen-statt-tippen-QR-Code-Visitenkarten-erstellen_46888134.html

siehe auch: <http://de.wikipedia.org/wiki/Mobile-Tagging>

QR-Code Online dekodieren

- Online: einfaches und zuverlässiges Tool von Google - <http://zxing.org/w/decode.jspx>
- Online: Suchanfrage - <http://www.google.de/search?q=%2Bonline+%22qr+decoder%22>

QR-Code offline dekodieren

- Ein alternatives Programm mit grafischer Oberfläche ist **qtqr**. Mit **qtqr** kann man URL, E-Mails, Texte codieren und decodieren.

siehe auch: `qrencode -h`, man `qrencode`

recode

recode konvertiert Dateien zwischen diversen Zeichensätzen und -formaten.

recode VORHER..NACHHER <Dateiname>

Beispiele:

recode -l ... listet alle Textkodierungen auf die recode kennt

recode --help ... Hilfe

recode UTF-8..LATIN1 ./gedichte/main.php ... kodiert die benannte Datei von UTF-8 nach LATIN1 (ISO-8859-1)

recode LATIN1..UTF-8 ./gedichte/main.php ... kodiert die benannte Datei von LATIN1 (ISO-8859-1) nach UTF-8

recode ibmpc..UTF-8 Namensbedeutung.txt ... kodiert die benannte Datei vom DOS-Format nach UTF-8

recode pc..UTF-8 Namensbedeutung.txt ... kodiert die benannte Datei vom DOS-Format nach UTF-8 (kürzere Syntax)

recode UTF-8..ibmpc Namensbedeutung.txt ... kodiert die benannte Datei von UTF-8 ins DOS/WINDOWS-Format

Achtung: Diese Befehle mit denselben Kodierungen nicht mehrmals auf die gleiche Datei anwenden - d.h. LATIN1..UTF-8 und danach nicht wieder LATIN1..UTF-8.

Funktionsmodi:

-v, --verbose ... Reihenfolge der Umkodierungsschritte und Fortschritt anzeigen

-q, --quiet, --silent ... keine Meldungen über nicht umkehrbare Umkodierungen

-f, --force ... Umkodierung vornehmen, auch wenn sie nicht umkehrbar ist

-t, --touch ... nach der Umkodierung ein »touch« (als Zeitstempel wird die aktuelle Zeit verwendet) auf die umkodierte Datei ausführen

siehe auch: recode --help, fromdos, todos, iconv, Konvertierung von Textkodierungen

rename

siehe auch: mv

route

Wenn es im Netzwerk einen Router oder ein Gateway gibt, muss dessen IP-

Adresse erscheinen. Dieser ist zum Beispiel notwendig, wenn Sie über einen ans LAN angeschlossenen Router ins Internet wollen. Fehlt der Default-Routing-Eintrag, kann er mit folgendem Befehl ergänzt werden (ersetzen Sie 192.168.0.254 durch die IP-Adresse Ihres Gateway-Routers und gegebenenfalls eth0 durch die passende Netzwerkkarte):

route add default gw 192.168.0.254 eth0

Die Änderungen sind nur für die aktuelle Sitzung gültig, d.h. nach einem Neustart des Rechners werden wieder die alten Einstellungen - entsprechend der Konfigurationsdateien im Verzeichnis /etc - verwendet. Dieser Befehl kann nur von root ausgeführt werden.

/sbin/route

Kernel IP Routentabelle

Ziel	Router	Genmask	Flags	Metric	Ref	Use	Iface
192.168.1.0	*	255.255.255.0	U	0	0	0	eth0
default	gateway	0.0.0.0	UG	0	0	0	eth0

In der Beispielausgabe legt die erste Zeile fest, dass die Kommunikation mit allen IP-Adressen, die mit 192.168.1 beginnen, über die Ethernet-Karte eth0 verläuft. Der Spezialeintrag »default« gilt für alle nicht eigens definierten Adressengruppen.

Beispiel:

Sie wollen Ihren Rechner kurzzeitig innerhalb eines fremden Netzwerkes (z.B. Internet-Café) betreiben, einschließlich Internet-Zugang. Dazu geben Sie folgende zwei Befehle an der Kommandozeile als Benutzer root ein (freie IP-Adresse: 192.168.1.100, Subnetz-Maske: 255.255.255.0, IP-Adresse des Internet-Gateway's: 192.168.1.1):

ifconfig eth0 192.168.1.100 netmask 255.255.255.0

route add default gw 192.168.1.1

Mit dem Befehl

route del default

können Sie Default-Route wieder löschen. Sie können den Rechner aber auch einfach herunterfahren, um die sitzungsbezogenen Änderungen wieder rückgängig zu machen.

Mit Kommandozeilentools ifconfig, host, route, ping, traceroute bzw. mtr und netstat diagnostizieren Sie bei Netzwerkstörungen - richtig eingesetzt -

gezielt jeden Teilaspekt einer Verbindung, intern oder im Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert - die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So finden Sie möglicherweise heraus, ob es sich bei einer nicht funktionierenden internen oder Internet-Verbindung um ein Problem mit einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS oder eines der Gegenseite handelt.

siehe auch: host, ifconfig, ping, ip, traceroute bzw. mtr und netstat

rm

Datei löschen (remove).

`rm <MeineDatei>`

Beispiele:

rm ./* ... löscht alle Dateien des aktuellen Verzeichnisses ohne Unterverzeichnisse und außer Konfigurationsdateien (.*)

rm -f <MeineDatei> ... Datei löschen, dabei werden nicht vorhandene Dateien ignoriert und es erfolgt auch keine Nachfrage

rm -r <MeinVerzeichnis> ... löscht alle Verzeichnisse inklusive aller Subdirectories (rekursiv)

rm -i <MeinVerzeichnis> ... wartet vor dem Löschen auf eine Bestätigung

rmdir <MeinVerzeichnis> ... ein Verzeichnis entfernen, falls es leer ist

Dateien deren Name mit mit einem Bindestrich beginnen, können nicht so ohne weiteres gelöscht werden.

weiteres Beispiel:

rm -rechnung ... kann nicht funktionieren, da der Kommandozeileninterpreter den Bindestrich als Option wahrnimmt

Lösung: Angabe des vollen Pfades

rm /home/andi/-rechnung

oder

rm ./-rechnung

oder

rm -- -rechnung ... dem Dateinamen -rechnung werden 2 Minuszeichen vorangestellt

siehe auch: man rm, shred

rmdir

Verzeichnis entfernen (remove directory) - das Verzeichnis muss leer sein.

rmdir <MeinVerzeichnis>

siehe auch: man rmdir, rm

reboot

Shutdown, reboot startet den Rechner erneut.

siehe auch: halt, shutdown, init

reset

Falls im Kommandozeilenfenster durch ungeschickte Kommandos (aus Versehen) der Zeichensatz umgeschaltet wurde, kann hiermit zurückgeschaltet werden (auch wenn beim Tippen falsche Zeichen erscheinen).

rsync

Rsync benutzt den »rsync Algorithmus«, eine sehr schnelle Methode, um Dateien im Netzwerk zu synchronisieren. Hierzu werden nur die Unterschiede in den Dateien übertragen, ohne dass auf einem der Hosts vorher beide Dateiversionen verfügbar sein müssen.

rsync <Optionen> <LokalerOrdner>
<Benutzer@Homepage>:<Verzeichnis>/

Die grundlegende Benutzung von rsync erfordert keine besondere Konfiguration. Mit rsync ist es direkt möglich, komplette Verzeichnisse auf einen anderen Rechner zu spiegeln.

Beispielsweise kann man mit folgendem Befehl eine Kopie des MP3-Verzeichnisses von tux auf ein Notebook - Rechnername sonne – anlegen (~ ... Home-Verzeichnis):

rsync -baz -e ssh /home/tux/MP3 tux@sonne:~

Um das Verzeichnis zurück zu spielen, findet folgender Befehl Verwendung:

rsync -az -e ssh tux@sonne:~/MP3 /home/tux/

Falls kein Domain Name Server installiert ist, so ist statt des Rechnernamens die IP-Nummer zu verwenden.

Optionen:

- e ssh** ... Übertragung erfolgt über ein SSH-Tunnel. Achtung: Auf dem entfernten Rechner muss ein SSH-Zugang (Port 22) freigeschaltet sein. Die Option **-e** spezifiziert die Remote Shell.
- a** ... kopiert die Dokumente archiviert; Die Archivübertragung stellt sicher, dass auch symbolische Links richtig ankommen. Zudem kopiert es die Verzeichnisse rekursiv.
- r** ... kopiert die Verzeichnisse rekursiv
- u** ... Nur Update, d.h. neuere Dateien werden nicht überschrieben.
- z** ... Komprimiert die übertragenen Daten. Dies macht den Transfer schneller.
- delete** ... Löscht Dateien auf dem Server, die im lokalen Verzeichnis nicht mehr vorhanden sind. Praktisch, um wieder mal aufzuräumen, aber auch gefährlich!
- n** ... Trockendurchlauf. Es werden keine Daten übertragen. Bei **--delete** sollten Sie immer erst einen Durchlauf mit **-n --delete** starten.
- b** ... Erstellt Backups mit dem Suffix ~ (Tilde).

Haben Sie ihren eigenen rsync-Befehl zusammengestellt, schreiben Sie diesen z.B. in die Datei **mp3sync** im Verzeichnis **bin** in Ihrem Home-Verzeichnis. Mit dem Befehl **chmod +x ~/bin/mp3sync** machen Sie die Datei ausführbar. Von nun an kopieren Sie die neuen Inhalte einfach mit dem Befehl **~/bin/mp3sync** z.B. auf Ihr Notebook.

Datei: mp3sync

```
#!/bin/bash  
rsync -e ssh [...]
```

siehe auch: man rsync, sitecopy, ssh



karl@tux1:~\$ lspci
00:00.0 Host bridge: Intel Corporation Mobile 4 Series Chipset Memory
Controller Hub (rev 07)
00:02.0 VGA compatible controller: Intel Corporation Mobile 4 Series Chipset
Integrated Graphics Controller (rev 07)
00:02.1 Display controller: Intel Corporation Mobile 4 Series Chipset Integrated
Graphics Controller (rev 07)
00:19.0 Ethernet controller: Intel Corporation 82567LF Gigabit Network
Connection (rev 03)
00:1a.0 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #4 (rev 03)
00:1a.1 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #5 (rev 03)
00:1a.2 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #6 (rev 03)
00:1a.7 USB Controller: Intel Corporation 82801I (ICH9 Family) USB2 EHCI
Controller #2 (rev 03)
00:1b.0 Audio device: Intel Corporation 82801I (ICH9 Family) HD Audio
Controller (rev 03)
00:1c.0 PCI bridge: Intel Corporation 82801I (ICH9 Family) PCI Express Port 1
(rev 03)
00:1c.1 PCI bridge: Intel Corporation 82801I (ICH9 Family) PCI Express Port 2
(rev 03)
00:1d.0 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #1 (rev 03)
00:1d.1 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #2 (rev 03)
00:1d.2 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #3 (rev 03)
00:1d.7 USB Controller: Intel Corporation 82801I (ICH9 Family) USB2 EHCI
Controller #1 (rev 03)
00:1e.0 PCI bridge: Intel Corporation 82801 Mobile PCI Bridge (rev 93)
00:1f.0 ISA bridge: Intel Corporation ICH9M LPC Interface Controller (rev 03)
00:1f.2 SATA controller: Intel Corporation ICH9M/M-E SATA AHCI
Controller (rev 03)
00:1f.3 SMBus: Intel Corporation 82801I (ICH9 Family) SMBus Controller
(rev 03)
04:00.0 Ethernet controller: Atheros Communications Inc. AR5001 Wireless
Network Adapter (rev 01)

S

Shell - eine textbasierte Shell öffnen, die die ganze Bildschirmseite ausfüllt. Die Tasten [Strg] + [Alt] gedrückt halten und eine Funktionstaste zwischen [F1] und [F6] bzw. [F7] drücken. Dann erscheint ein Login-Bildschirm, nach Eingabe von Benutzername und Passwort steht die ganze Welt der Shell zur Verfügung. Um zwischen mehreren Text-Shells umzuschalten, ist die Taste [Alt] + [F1] zu drücken bzw. die Tasten [F2] bis [F6] bzw. [F7]. Um zurück zur grafischen Oberfläche zu kommen, ist die Tastenkombination [Alt] + [F7] oder [Alt] + [F8] zu drücken.

Hinweis: Einige Linux-Distributionen verwenden 6 Shell's und andere 7 Shell's im Vollbildmodus.

In jedem virtuellen Terminal können Programme gestartet werden, die auch weiterlaufen, wenn auf ein anderes Terminal umgeschaltet wird. Die Tastatureingabe wird immer nur an das aktuelle Terminal im Vordergrund weitergegeben. Die Bildschirmausgabe der einzelnen Terminals wird erst sichtbar, wenn dorthin umgeschaltet wurde. Nur die Statusmeldungen und die Fehlerausgaben des Kernels werden nicht auf ein virtuelles Terminal, sondern auf die Konsole direkt ausgegeben und erscheinen dadurch immer auf dem Bildschirm.

mc ... startet den Dateimanager Midnight Commander

Mit der Taste [**Pfeil oben**] wird jeweils der zuletzt eingebende Befehl angezeigt.

Mit [**Pfeil unten**] wird entsprechend wieder ein Befehl weiter nach vorn geblättert.

Mit [**Strg**] + [**r**] kann nach einem Befehl gesucht werden, einfach eine Buchstabenfolge eingeben. Mit [**Return**] wird der Befehl sofort ausgeführt - mit [**Esc**] kann der Befehl bearbeitet werden.

history ... alle noch gespeicherten Befehle ausgeben

cd oder **cd ~** ... zum Heimatverzeichnis des aktuellen Benutzers wechseln

[**Strg**] + [**C**] ... im Terminal laufende Befehle, Programme abbrechen

[**Strg**] + [**u**] ... komplette aktuelle Kommandozeile löschen (Text vor dem aktuellen Cursor-Standort)

[**Strg**] + [**L**] ... bereinigt den Bildschirm von überflüssig gewordenen

Textzeilen

[Strg] + [s] ... laufende Bildschirmanzeige anhalten

[Strg] + [q] ... Bildschirmanzeige nach **[Strg] + [s]** wieder fortsetzen

Mit der Taste **[Tab]** versucht die Shell anhand der bisherigen Angaben, einen Datei- oder Verzeichnisnamen zu ermitteln. Mit der Eingabe von **cd / v + [Tab]** wird der Verzeichnisname zu **cd /var/** ergänzt. Rührt sich bei der Eingabe eines Verzeichnis- oder Dateinamen nichts, so ist dies ein Zeichen, dass der Kommandozeileninterpreter zu viele Verzeichnis- oder Dateinamen gefunden hat. In diesem Fall noch einmal auf **[Tab]** drücken und die Shell zeigt alle passenden Einträge an.

Innerhalb der textbasierte Shell wird sich mit **exit** bzw. **[Strg] + [d]** abgemeldet.

Terminalkommandos aus einer Textdatei in ein Terminalfenster einfügen (grafische Oberfläche)

Immer wieder vorkommende Terminalkommandos, die sich nicht oder nur geringfügig unterscheiden, können in einer einfachen Textdatei gespeichert werden. Um die gespeicherten Kommandos in ein Terminalfenster zu kopieren, gibt es nicht nur eine Methode sondern viele.

1. Das Kommando in der Textdatei mit der Maus markieren und per Drag and Drop - mit der linken Maustaste - ins geöffnete Terminalfenster ziehen.
2. Das Kommando in der Textdatei markieren und über die rechte Maustaste das Kommando an die Zwischenablage übergeben. Das geöffnete Terminalfenster mit der Maus anklicken und über die rechte Maustaste das Kommando aus der Zwischenablage ins Terminalfenster einfügen.
3. Das Kommando in der Textdatei markieren und mit der Tastenkombination **[Strg] + [C]** das Kommando an die Zwischenablage übergeben. Das geöffnete Terminalfenster mit der Maus anklicken und über die Tastenkombination **[Strg] + [Shift] + [V]** das Kommando aus der Zwischenablage ins Terminalfenster einfügen.
4. Das Kommando in der Textdatei markieren und mit dem Mousrad bzw. der mittleren Maustaste ins geöffnete Terminalfenster klicken (Hinweis: Mit dem Mousrad kann man nicht nur in längeren Texten scrollen, sondern auch wie mit einer Maustaste klicken). Das Kommando wird am aktuellen Cursor-Standort eingefügt. Auf

diese Weise kann man auch längere Kommandos aus einem Terminalfenster in eine Textdatei kopieren.

Hinweis: Werden die Kommandos statt aus einer einfachen Textdatei aus einem PDF-Dokument (.pdf), Libre Office Dokument (.odt), etc. in ein Terminalfenster kopiert, so werden einige Kommandos mit Sonderzeichen mit einiger Wahrscheinlichkeit nicht funktionieren. Diese Textdokumente ersetzen intern viele Zeichen durch ihre eigenen Zeichen, obwohl in der Bildschirmansicht kein Unterschied feststellbar ist.

siehe auch: clear, Tipps und Tricks → Cinnamon: Tastenkombinationen unter Linux Mint

* * * * *

sed

Eins der nützlichsten Programme in einem Linux System ist der Ersetzer **sed**. Er stellt ein flexibles Tool zum Ersetzen von Ausdrücken in Dateien dar. Ein praktisches Beispiel: Die Firma XY hat ihre Homepage vom Server **server1.xy.de** auf **server2.xy.de** verlegt. In den HTML-Dateien müssen nun die vielen Links geändert werden. Eine Möglichkeit wäre, die Dateien auf einer grafischen Oberfläche in den Lieblingseditoren und dessen »Ersetzen«-Funktion zu verwenden. Diese durchaus zweckdienliche Methode für »Mausverliebte« hat jedoch eine schlagkräftige Alternative. Ihr Name ist **sed**. Dieses einfach zu bedienende Programm erledigt die Arbeit fast von alleine. In unserem Beispiel müssen also alle Ausdrücke **server1** in **server2** umgeschrieben werden. Dies erledigt folgender Befehl: **sed s/server1/server2/g dateiname.html**. Eine kurze Erklärung: **sed** ist der Befehlsaufruf. Auf ihn folgt ein Text, der in Slash's (/) gefasst ist. Das ist das Ersetzungsskript. Dieses Skript enthält die Informationen, was wie ersetzt werden soll. Entschlüsseln wir es einmal: **s/server1/server2/g**. Das erste **s** leitet die Ersetzung ein. Der darauf folgende **/** deutet auf den Beginn des zu ersetzenden Wortes hin. Nun kommt das zu ersetzende Wort, gefolgt von einem weiteren **/**, der das Wort angibt, durch das ersetzt werden soll (hier server2). Der dritte **/** beendet das einzufügende Wort. Im allgemeinen ersetzt **sed** immer nur den ersten Ausdruck, den es in einer Zeile findet. Das **g** am Ende sorgt dafür, dass auch folgende Ausdrücke, die dem Suchmuster entsprechen ersetzt werden. Es steht für global. Wer diesen Befehl einmal getestet hat, wird enttäuscht feststellen, dass die Ausgabe der geänderten Datei auf der Konsole erfolgt. Leider können wir nicht die Ausgabe von **sed** nach **dateiname.html** umleiten, da die Datei ansonsten durch den Aufruf einer Endlosschleife vernichtet würde. Wir können jedoch, wie in »Umleiten in eine Datei« gelernt, die Ausgabe in eine

andere Datei umleiten. Dann sieht der Befehl so aus: **sed s/server1/server2/g dateiname.html > dateiname2.html**. Nun liegt die aktualisierte Fassung von dateiname.html in dateiname2.html. Nach einer Überprüfung des Ergebnisses kann die neue Version mittels **mv dateiname2.html dateiname.html** die alte Version ersetzen. Die alte Version geht hierbei jedoch verloren. Vorsicht ist angesagt. Besondere Vorsicht sollte geboten sein, wenn sich Sonderzeichen oder Operatoren in Quelle oder Ziel der Ersetzung befinden. Beispiele hierfür sind folgende Zeichen: **! " \$ % & / () { } = ? \ . ;** Diese müssen im sed-Skript mit vorangestelltem **** verwendet werden, also zum Beispiel **** für einen Backslash oder **\/** für einen Slash. Man sollte hier das Endergebnis besonders sorgfältig prüfen! Ratsam ist auf jeden Fall das Lesen der Manual-Page zu sed (**man sed**).

Automatisierte Bearbeitung von Texten mit sed

von Heike Jurzik - Linux-User 08/2005

Für stets wiederkehrende Aufgaben an einer oder mehrerer Dateien kann das Arbeiten mit einem Texteditor schnell mühsam werden. Mit **sed** sparen Sie jede Menge Tipparbeit.

sed <Optionen> '<Befehl(e)>' <Datei(en)>

- p** ... Gibt die entsprechenden Zeilen auf dem Bildschirm aus.
- d** ... Löscht den definierten Bereich.
- s** ... Ersetzt Zeichenketten.
- a** ... Fügt Text hinter den adressierten Zeilen ein (Einsatz in Skripten).
- i** ... Fügt Text vor den adressierten Zeilen ein (Einsatz in Skripten).
- c** ... Ersetzt Zeilen oder Zeilenbereiche (Einsatz in Skripten).
- f** ... Liest den Inhalt einer Skriptdatei ein.
- r** ... Liest den Inhalt einer Datei ein und setzt diesen hinter die entsprechende Adresse.
- w** ... Schreibt die adressierten Zeilen oder Bereiche in eine neue Datei.

Der Stream-Editor arbeitet im Gegensatz zu Texteditoren wie vi und Emacs nicht interaktiv. Stattdessen definieren Sie vorher auf der Kommandozeile, welche Textbereiche zu löschen, zu ändern und einzufügen sind. Bei Bedarf packen Sie auch alle Anweisungen in ein spezielles Skript und führen Sie auf einem Rutsch aus. Gilt es in einer oder mehreren Dateien gleich einige Zeichen und Zeichenkombinationen zu verändern, verhindert sed, dass Sie sich die Finger wund tippen.

Erste Schritte

Normalerweise erhält sed die zu verarbeitenden Texte über die Standardeingabe (stdin). Dabei können Sie eine oder mehrere Dateien angeben. sed bearbeitet jede Zeile mit den angeführten Befehlen und schreibt sie in einen Puffer, dessen Inhalt zuletzt - falls nicht anders definiert - auf der Standardausgabe (stdout) erscheint. Die generelle Syntax lautet:
sed <Optionen> '<Befehl(e)>' <Datei(en)>

Damit die Shell den Befehl nicht auswertet, schließen Sie das Kommando in einfache Hochkommas ein. Die Befehle beziehen sich entweder auf einzelne Zeilen, Zeilenbereiche oder - wenn eine derartige Angabe fehlt - auf die ganze Datei.

Die Angabe, welche Zeilen zu bearbeiten sind, bezeichnet man als Adresse. Eine Übersicht über die wichtigsten sed-Kommandos und deren Bedeutung finden Sie am Anfang dieses Artikels.

Einfach nur ausgeben

Wie schon erwähnt, wenden Sie sed-Befehle auf so genannte Adressen (bestimmte Zeilen oder einen Zeilenbereich einer Datei) an. Zusammen mit Regular Expression (reguläre Ausdrücke) lassen sich Dinge sehr genau treffen. In den folgenden Abschnitten und Beispielen stellt der Artikel für die verschiedenen sed-Kommandos Möglichkeiten für die Adressierung vor - diese gelten dann entsprechend für sämtliche Befehle des Stream-Editors. Zur schlichten Ausgabe dient das Kommando **p**. Um Beispielsweise lediglich die zweite Zeile anzuzeigen, stellen Sie den Befehl einfach die Ziffer 2 voran. Der vollständige Befehl sieht dann wie folgt aus:

sed '2p' datei

Hier ist die erste Zeile.

Die zweite Zeile.

Die zweite Zeile.

Hier steht die 3. Zeile.

[...]

Die Ausgabe dieses Kommandos sieht etwas anders aus als erwartet. Statt der zweiten Zeile erscheint die komplette Datei, die zweite Zeile auch noch zweimal. Die überflüssigen Ausgaben unterbinden Sie mithilfe der Option **-n**:

sed -n '2p' datei

Die zweite Zeile.

Löschzug

Auch für das Kommando zum Löschen (**d**) geben Sie im einfachsten Fall

genau eine Zeile an, die bearbeitet werden soll. So löscht der Befehl

sed '1d' datei

die erste Zeile aus **datei**. Um gleich mehrere Zeilen zu erwischen, schreiben Sie die erste und letzte, durch ein Komma getrennt, in den Befehl. So löscht **sed '2,4d' datei**

die zweite bis vierte Zeile. Ebenso leicht treffen Sie bei Bedarf auch jede n-te Zeile; das Kommando

sed '1~3d' datei

entfernt jede dritte Zeile - ausgehend von der ersten. Wer von der fünften bis zur letzten Zeile der Datei komplett Tabula rasa machen möchte, muss nicht erst Zeilen zählen und diese dann explizit angeben, sondern kann einen regulären Ausdruck zur Hilfe nehmen. Mit dem Aufruf

sed '5,\$d' beispiel.txt

entfernen Sie alle Zeilen ab der einschließlich fünften bis zum Datei-Ende, das durch das Dollar-Zeichen definiert wird.

Besonders praktisch erweist sich der Befehl zum Löschen, wenn Sie Konfigurationsdateien in /etc betrachten wollen, die sehr viele Kommentare enthalten. Mit sed löschen Sie einfach alle Zeilen heraus, die ein Rautezeichen am Zeilenanfang haben:

sed '/^#.*d' /etc/inetd.conf

Als Adresse übergeben Sie dem d-Kommando einen in Schrägstriche eingeschlossenen regulären Ausdruck, der alle Zeilen bezeichnet, die mit einer Raute (#) beginnen und danach keine oder beliebige Zeichen enthalten. So landen allerdings auch leere Zeilen der Konfigurationsdatei im Terminal. Mit einem anderen regulären Ausdruck (^[^#].*) drucken Sie nur diejenigen Zeilen (p-Befehl), die mit einem Zeichen beginnen, das keine Raute ist:

sed -n '/^[^#].*/p' /etc/inetd.conf

Ersetz dich!

Zusammen mit dem s-Kommando ersetzen Sie Zeichenketten. Nach dem Befehl folgt ein Trennzeichen, das Suchmuster, ein weiteres Trennzeichen, die neu einzufügenden Zeichen und zuletzt ein abschließendes

Trennzeichen. Was Sie dabei als Trennzeichen verwenden, steht Ihnen grundsätzlich frei - das Zeichen selbst darf aber nicht im Muster vorkommen.

Um in einer Datei jedes Vorkommen des Worts »Zeile« durch »line« zu ersetzen, können Sie beispielsweise folgendes Kommando darauf loslassen:

sed 's/Zeile/line/' datei

Da sed auf diese Weise nur jeweils das erste Vorkommen des Suchmusters in einer Zeile ersetzt, können Sie dem Editor durch den Befehl **g** (g ... global) mitteilen, dass alle Treffer verwandelt werden sollen:

sed 's/Zeile/line/g' datei

Kommt der hier als Trennzeichen eingesetzte Schrägstrich im Suchmuster selbst vor, dann weichen Sie auf einen anderen Begrenzer aus, wie etwa die Raute (#) oder das Pipe-Zeichen (|):

sed 's#http://www.huhn.de#http://www.gockel.de#g' url.html

Auch das Substitutionskommando (**s**) können Sie natürlich nicht nur auf die gesamte Datei loslassen, sondern auch auf einzelne Zeilen. Der Befehl

sed '1s/Zeile/line/g' datei

sucht und ersetzt nur in der ersten Zeile der Datei. Auch hier ist das **g** wieder wichtig, falls es mehr als ein Vorkommen des Suchmusters gibt.

Mehrere Kommandos

Um gleich mehrere Aufgaben mit nur einem einzigen sed-Aufruf abzuarbeiten, stellen Sie den einzelnen Befehlen die Option **-e** voran. So löschen Sie beispielsweise mit dem folgenden Kommando erst ab der fünften Zeile bis zum Datei-Ende alles und lassen auf den Rest der Datei eine Suchen-Ersetzen-Aktion los:

sed -e '5,\$d' -e 's/KDE/Gnome/g' datei

Alternativ trennen Sie alle Befehle durch Semikolon voneinander und packen Sie in geschweifte Klammern. So lässt sich der letzte Aufruf beispielsweise alternativ auch so schreiben:

sed {'5,\$d;s/KDE/Gnome/g'} datei

Lesen und Schreiben

Auch zum Einlesen und Speichern bringt sed das passende Kommando mit. Wer beispielsweise eine Datei namens extras hinter der dritten Zeile einfügen möchte, tippt einfach:

sed '3r extras' datei

Ebenso leicht können Sie mithilfe des Befehls **w** Textstellen extrahieren und separat abspeichern:

sed '1,4w 1bis4.txt' datei

Dieses Kommando schreibt die ersten 4 Zeilen in die neue Datei 1bis4.txt.

Befehlsgewaltig

Bei Bedarf bündeln Sie mehrere sed-Befehle in einer Skript-Datei, die Sie dann mit der Option **-f** auf die gewünschten Dateien ansetzen.

Um beispielsweise aus einer Datei die zweite Zeile zu löschen und eine Zeile nach der vierten zu ergänzen, schreiben Sie die Befehl untereinander:

2d

4a

Nach der vierten Zeile steht hier etwas.

Für das a-Kommando (4a) ist es wichtig, dass Sie nach dem Befehl selbst einen Backslash (\) und Zeilenumbruch einfügen. Der neu einzusetzende Text steht in einer neuen Zeile.

Handelt es sich um mehr als eine Zeile, müssen Sie jede (bis auf die letzte Zeile) mit einem Backslash abschließen:

4a

Nach der vierten Zeile steht hier etwas.

Und noch etwas.

Und noch ein bisschen mehr :)

Soll der neue Text nicht nach einer Zeile erscheinen, sondern vorangestellt werden, wählen Sie statt **a** den **i**-Befehl:

2i

Hier steht etwas Neues ...

Die so erstellte Befehlsdatei speichern Sie ab (etwa unter dem Namen script), übergeben diese dann an sed und bearbeiten eine oder mehrere

Dateien damit:

sed -f script datei

Direkt in die Datei

Wie bereits erwähnt, verändert sed standardmäßig nicht die Originaldatei, sondern schreibt seine Ausgabe stattdessen nach stdout (meist der Bildschirm). Haben Sie sich überzeugt, dass alle Veränderungen übernommen werden sollen, können Sie die Ausgabe entsprechend umleiten. Mit dem Operator > schreiben Sie beispielsweise das Ergebnis in eine Datei:

sed -f script datei > neuedatei

Sollen die Veränderungen ohne Umweg direkt in der Originaldatei landen, bringt sed dafür den Parameter **-i** mit. So führt das Kommando

sed -i -f aufgaben datei

dazu, dass sed die Datei **datei** mit den Änderungen direkt überschreibt. Auch eine automatische Sicherungskopie erstellt der Parameter auf Wunsch: Fügen Sie einfach direkt an die Option die Datei-Endung an, welche das Backup tragen soll:

sed -i .bak -f aufgaben datei

Neben **datei** - mitsamt den vorgenommenen Änderungen - finden Sie anschließend auch das Original unter dem Namen **datei.bak** im Arbeitsverzeichnis.

Kombinationsgabe

Besonders praktisch erweist sich sed in Zusammenarbeit mit anderen Programmen auf der Kommandozeile. Angenommen, in einem Verzeichnis liegen mehrere Dateien mit Leerzeichen und Bindestrichen im Namen, die in Unterstriche (**_**) umgewandelt werden sollen. In diesem Fall sind die verschiedenen Ersetzungsregeln für sed einfach in ein Skript zu schreiben:

```
s/_/_g
```

```
s/-/_g
```

Natürlich können die beiden Zeilen auch zu einem einzigen regulären Ausdruck

s/[-]/_/g

zusammengefasst werden - das ist für das aktuelle Beispiel aber irrelevant. Das Skript ist zunächst an den Dateien zu testen, um zu überprüfen, ob im Ernstfall alles richtig ersetzt wird.

Da sed direkt von der Standardeingabe zu lesen vermag, kann die Ausgabe des ls-Kommandos in diesem Fall unmittelbar über eine Pipe an den sed-Befehl weitergereicht werden:

ls -l *.mp3 | sed -f script

Sind die Änderungen in der Ausgabe von Fehlern frei, so kann im nächsten Schritt das Programm **mv** zum Umbenennen der Dateien mit dem vorherigen Befehl kombiniert werden. Damit **sed** direkt alle fraglichen MP3-Dateien auf einmal erfasst, ist noch eine for-Schleife in den Befehl einzubauen:

for i in *.mp3; do mv -v "\$i" 'echo \$i | sed -f script'; done

Im Klartext heißt das: Für alle Dateien, die auf *.mp3 enden, mache Folgendes: Verschiebe diese sichtbar in das Ergebnis der sed-Operation. Da in den Original-Dateinamen bis zur Bearbeitung durch sed noch Leerzeichen enthalten sind, muss man den Ausdruck **\$i** in Anführungszeichen einschließen.

siehe auch: Anhang: Einführung in die Shellprogrammierung

* * * * *

Samba und Freigaben

Freigaben vernetzter Rechner werden am einfachsten über Samba erreicht. Nach dem Starten eines Dateimanagers, ist in dessen Adresszeile folgendes einzugeben:

smb:/ ... Anzeige der Windows-Arbeitsgruppen und -Domänen im LAN.

smb://<Arbeitsgruppenname> ... Anzeige der Rechner in dieser Arbeitsgruppe.

smb://<Computername> ... Anzeige der Freigaben dieses PCs.

smb://<Computername>/<Freigabename> ... Anzeige (und Bearbeitung) der Inhalte dieser Freigabe.

net usershare list ... listet alle Freigaben des aktuellen Rechners auf

net usershare info ... liefert detaillierte Informationen über die Freigaben

siehe auch: man Samba, man net, man smb.conf

sfill

Das Programm sfill implementiert die Gutmann-Methode. Bedingt durch die Gutmann-Methode ist die Anzahl der Überschreibdurchgängen (35 Schreibvorgänge) relativ hoch und damit auch die Belastung für die Festplatte.

Da ein Verzeichnis angegeben werden muss, erzeugt man vorher z.B. das Verzeichnis /tmp_sfill.

sudo sfill /tmp_sfill

Über die Optionen des Programms lässt sich die Anzahl der Überschreibdurchgänge verringern. Zum Beispiel ermöglicht folgender Befehl nur einen Überschreibdurchgang (mit Pseudozufallszahlen):

sudo sfill -l -lv /tmp_sfill

Danach kann das entsprechende Verzeichnis wieder gelöscht werden.

Hinweis: alternative Methode über dd

Das Programm dd füllt den freien Speicher mit einer Datei (Name: nulldatei.000) aus Nullen, die im aktuellen Verzeichnis des benutzten Terminals angelegt wird. Nach vollständigem Füllen des Speichers sollte eine entsprechende Fehlermeldung ("Auf dem Gerät ist kein Speicherplatz mehr verfügbar") erscheinen und die Datei kann gelöscht werden. Nach dem Abbruch durch eine Fehlermeldung, wird mithilfe von sync das Dateisystem wieder synchronisiert. Bei dieser Methode wird eine geringe Menge des Speichers nicht überschrieben.

Hinweis – Sync:

Normalerweise verwendet Linux einen Puffer (Cache) im Arbeitsspeicher, in dem sich ganze Datenblöcke eines Massenspeichers befinden. So werden Daten häufig temporär erst im Arbeitsspeicher verwaltet, da sich ein dauernd schreibender Prozess äußerst negativ auf die Performance des Systems auswirken würde.

Mit dem Kommando sync können Sie nun veranlassen, dass veränderte Daten sofort auf die Festplatte (oder auf jeden anderen Massenspeicher) geschrieben werden.

**dd if=/dev/zero of=nulldatei.000
sync**

rm nulldatei.000

Besonderheit SSD-Medien

Etwas einfacher ist die Handhabung freien Speichers auf SSD-Medien.

sudo fstrim -v /dev/<Gerätename>

Das Programm fstrim findet freie Blöcke und meldet sie dem SSD-Controller, dieser markiert sie als unbelegt. Zwar stehen die dort gespeicherten Daten noch in den Flash-Zellen, sie sind aber nicht mit Hausmitteln auslesbar - das geht nur am SSD-Controller vorbei - mit extrem aufwendiger und teurer Gerätetechnik und Methoden professioneller Datenretter.

siehe auch: dd, man sfill

shred

Mit shred kann man Daten innerhalb Dateien oder Dateien selbst sicher von der Festplatte durch mehrfaches Überschreiben löschen, so dass die Daten selbst mit teuren Hardware-Analysemitteln i.d.R. nicht wiederherzustellen sind.

shred -f -n 39 <Dateiname> ... überschreibt den Dateiinhalt 39 mal ohne die Datei zu löschen

shred -fz -n 39 <Dateiname> ... überschreibt den Dateiinhalt 39 mal ohne die Datei zu löschen, beim letzten Überschreiben werden alle Bits auf Null gesetzt (z ... zero)

shred -fzu -n 39 <Dateiname> ... überschreibt den Dateiinhalt 39 mal, beim letzten Überschreiben werden alle Bits auf Null gesetzt (z ... zero) und anschließend wird die Datei gelöscht

VORSICHT: Beachten Sie, dass shred auf einer sehr wichtigen Annahme beruht: dass das Dateisystem Daten an derselben Stelle überschreibt. Das ist die althergebrachte Vorgehensweise, doch viele moderne Betriebssystemdesigns erfüllen diese Annahme nicht. Die folgenden Systeme sind Beispiele von Dateisystemen, auf denen shred keine Wirkung hat:

- Log-strukturierte oder »journal« Dateisysteme, so wie die mit AIX und Solaris gelieferten (und JFS, ReiserFS, XFS, Ext3, usw.)

- Dateisysteme, die redundante Daten schreiben und auch dann fortfahren, wenn einige Schreibvorgänge fehlschlagen, so wie RAID-basierte Dateisysteme
- Dateisysteme, die Schnappschüsse anfertigen, so wie der NFS-Server von Network Appliance
- Dateisysteme, die an temporären Orten zwischenspeichern, so wie Client's unter NFS Version 3
- komprimierte Dateisysteme

Außerdem können Dateisystemsicherungen und entfernte Spiegel Kopien der Datei enthalten, die nicht entfernt werden können, und die es erlauben, eine zerhackte Datei wieder herzustellen.

siehe auch: rm, shred --help

shutdown

Mit dem Kommandozeilen-Befehl shutdown wird der Rechner heruntergefahren.

shutdown -h now ... fährt den Rechner jetzt herunter

shutdown -r now ... Restart, Neustart des Rechners

siehe auch: halt, reboot, init

siege

Das Kommandozeilen-Tool siege erzeugt auf dem Ziel-Server eine konfigurierbare Anzahl von Anfragen.

Hinweis: Die Anfragen an einen Ziel-Server, sollten immer zeitlich begrenzt sein. Die Provider könnten sonst den Stress-Test als DoS-Attacke (Denial-of-Service) werten und entsprechende Maßnahmen einleiten.

siege -c 50 -b http://www.example.de ... startet den Test mit 50 gleichzeitigen Verbindungen (-c 50); die Tastenkombination **[Strg] + [C]** beendet den Test

Nach dem Abbruch über die Tastenkombination **[Strg] + [C]** wird eine Statistik mit den Messwerten ausgegeben.

Mit dem Parameter **-f [Dateiname]** kann man dem Testaufruf eine Textdatei mit mehreren URLs (eine URL pro Zeile) übergeben.

siehe auch: man siege, ab, htop

Sitecopy

Sitecopy ist ideal, um eine oder mehrere Sites von einem Entwicklungsserver aus mit neuesten Skripts zu versorgen. Das Programm erlaubt, neue Dateien zu installieren und löscht Dateien, die auf dem Entwicklungsserver nicht mehr vorhanden sind. Das verhindert Datenwildwuchs auf dem Server.

In einer Konfigurationsdatei lassen sich für jeden Ziel-Server Ausnahmen (exclude) festlegen.

Das Programm arbeitet über FTP und unterstützt auch abgesicherte Verbindungen, sowie WebDAV.

Vor dem Arbeiten mit dem Programm sind noch ein Verzeichnis und eine Konfigurationsdatei manuell zu erstellen - siehe man sitecopy.

mkdir -m 700 .sitecopy

touch .sitecopyrc

chmod 600 .sitecopyrc

Im Verzeichnis .sitecopy führt das Programm über die Abgleiche Buch, dazu legt es für jede Seite eine XML-Datei an. Über die Konfigurationsdatei teilen Sie dem Programm mit, was es woher wohin kopieren soll.

Nach dem Erstellen der Konfigurationsdatei (siehe weiter unten) initialisieren Sie die XML-Datei von Sitecopy mit dem Befehl:

sitecopy --init <www.meineseite.de>

meldet das Programm keine Fehler, so laden Sie anschließend die Inhalte mit folgenden Befehl auf den Server:

sitecopy --update <www.meineseite.de>

Danach benötigen Sie nur noch die --update-Option von Sitecopy.

Beachte: Mit **sitecopy --init** werden auch alle Dateien als zu aktualisieren gekennzeichnet.

Beispiel: .sitecopyrc

Zugriffsrechte werden bei »sitecopy --update« vom lokalen Rechner zum entfernten Server übertragen und bei »sitecopy --synchronize« vom entfernten Server zum lokalen Rechner übertragen - permissions all; symbolische Links (Verknüpfungen) werden ignoriert - symlinks ignore; Sicherungsdateien mit einer Tilde (~) am Ende des Dateinamens werden vom Update ausgeschlossen - exclude *~; die anderen Zeilen dürften

selbsterklärend sein

```
site www.meineseite.de
      server ftp.server.de
      protocol ftp
      username sflfsaij
      password geheim
      local /home/ernst/website
      remote verzeichnis/
      permissions all
      symlinks ignore
      exclude *~
```

Empfehlung: Jede Sitenamen-Konfiguration mit einer Leerzeile abschließen.

In den Beispielen wird davon ausgegangen, dass die Initialisierung von Sitecopy (sitecopy --init) schon durchgeführt wurde.

Beispiel 1: Es befinden sich auf dem entfernten Server noch keine Dateien. In diesem Fall brauchen Sie nur den Update-Befehl ausführen.

```
sitecopy --update <www.meineseite.de>
```

Beispiel 2: Befinden sich schon Dateien auf dem entfernten Server, so gleichen Sie die lokalen Dateien an die Verzeichnis- und Dateistruktur des Servers manuell an. Danach sollten Sie Sitecopy wie folgt aufrufen:

```
sitecopy --catchup <www.meineseite.de>
sitecopy --update <www.meineseite.de>
```

Beispiel 3: Befinden sich schon Dateien auf dem entfernten Server, so können Sie Synchronisierung auch Sitecopy überlassen. Als erstes sollte die lokale Verzeichnis- und Dateistruktur mit dem entfernten Server synchronisiert werden. Dazu rufen Sie nacheinander die folgenden beiden Befehle auf.

```
sitecopy --fetch <www.meineseite.de>
sitecopy --synchronize <www.meineseite.de>
```

Nun können Sie die synchronisierten Dateien in Ihrer Homepage-Verzeichnis auf dem lokalen Rechner verändern, löschen oder neue Dateien einfügen. Danach rufen Sie den Update-Befehl auf.

sitecopy --update <www.meineseite.de>

Für rein statische Webseiten brauchen Sie anschließend nur den Update-Befehl aufrufen.

Für dynamische Webseiten auf denen Besucher Dateien verändern oder einfügen können, sollten Sie in regelmäßigen Abständen den gesamten Synchronisations-Vorgang wiederholen.

weitere Optionen und Modi:

sitecopy [OPTIONEN] [MODUS] [sitename]...

Optionen:

- g, --logfile=DATEI** ... schreibe Debug-Meldungen in DATEI (sonst auf stderr)
- y, --prompting** ... vor jeder Aktualisierung nachfragen
- a, --allsites** ... Aktion auf ALLEN bekannten Sites ausführen
- k, --keep-going** ... Aktualisierung bei Fehlern fortsetzen
- o, --show-progress** ... zeige, zu wie viel Prozent der Transfer beendet ist
- q, --quiet** ... beim Ausführen ruhig bleiben
- qq, --silent** ... beim Ausführen still bleiben (stärker als -q)

Aktions-Modi:

- l, --list** ... Zeige Veränderungen zwischen lokalem Verzeichnis und Server an (Voreinstellung)
- v, --view** ... gib eine Liste der definierten Sites aus
- i, --initialize** ... kennzeichne alle Dateien als zu aktualisieren
- f, --fetch** ... finde heraus, welche Dateien auf dem Server liegen
- e, --verify** ... vergleiche den gespeicherten mit dem wirklichen Zustand der Site
- c, --catchup** ... kennzeichne alle Dateien als aktualisiert
- s, --synchronize** ... gleiche die lokalen Dateien an den Server an
- u, --update** ... aktualisiere die Dateien auf dem Server

siehe auch: man sitecopy, sitecopy --help, Zugriffsrechte

skill

Mit skill kann man z.B. Prozesse beendet, die abgestürzt sind oder die sich anders nicht mehr beenden lassen.

skill -c <Kommnadoname mit dem das Programm aufgerufen wurde>

- t** ... das nächste Argument ist das Terminal (tty oder pty).
- u** ... das nächste Argument ist der Username

-p ... das nächste Argument ist die Prozess-ID

-c ... das nächste Argument ist der Kommandoname mit dem der Prozess gestartet wurden

skill -t pts/1 -c sleep ... schließt z.B. von der zweiten Konsole (pts/2) aus den Prozess sleep, dass in der 1. Konsole (pts/1) gestartet wurde

siehe auch: man skill, kill, ps, xkill

split

Mit split kann man große Dateien in kleinere Stücke zerteilen.

Zusammengesetzt werden diese Teilstücke mit dem Befehl cat. split ergänzt die Dateinamen der **kleinedatei** mit Buchstaben z.B. kleinedateiaa, kleinedateiab, kleinedateiac etc.

split -b 1m <grossedatei> <kleinedatei> ... zerteilt die grossedatei in 1 MB kleine Stücke

split -b 500k <grossedatei> <kleinedatei> ... zerteilt die grossedatei in 500 kB kleine Stücke

split -b 800 <grossedatei> <kleinedatei> ... zerteilt die grossedatei in 800 Byte kleine Stücke

cat <kleinedatei*> > <grossedatei> ... mit cat werden die Dateistücke wieder zusammengesetzt (* ist ein so genanntes Wildcard-Zeichen, es steht für beliebige Zeichen).

Die Angabe der Größe der Dateistücke für KByte, MByte, etc. kann wie folgt angegeben werden: KB (1000 Byte), K (1024 Byte), MB (1000*1000 Byte), M (1024*1024 Byte), GB (1000*1000*1000 Byte), G (1024*1024*1024 Byte), ... T, P, E, Z, Y.

Hinweis: Die 2-Buchstaben-Kombinationen (KB, MB, ...) müssen immer in der Großschreibweise eintragen werden. Bei den einzelnen Buchstaben K, M, G ..., wird durch split nicht zwischen Groß- und Kleinschreibung unterschieden.

siehe auch: man split

Sound

In den nachfolgenden Zeilen, wird nur sehr kurz auf den Sound unter Linux eingegangen. Ein Sound-Programme mit grafischer Oberfläche ist z.B.

»Audacity« .

MP3

Die Abkürzung MP3 steht für MPEG 1 Audio Layer 3 (Encoder: lame). Ein Verfahren, das Audiodaten in CD-Qualität ohne nennenswerte Verluste komprimiert.

LAME-Paket: Mittels des LAME-Pakets können MP3-Dateien erzeugt werden. Das LAME-Paket ist i.d.R. Bestandteil der meisten Linux-Distributionen.

lame -h inputfile.wav outputfile.mp3 ... erstellt aus der Wave-Datei inputfile.wav die MP3-Datei outputfile.mp3

lame -h inputfile.wav ... erstellt aus der Wave-Datei inputfile.wav die MP3-Datei inputfile.wav.mp3; die Dateiendung mp3 wird hier automatisch angehängt

-b <bitrate> ... setzen der Bitrate (minimale Bitrate), ohne Angabe wird der Vorgabewert 128 kbit/sek. gesetzt (160 kbit/sek. → sehr hohe Qualität, 48 kbit/sek. → bessere Mittelwellen-Qualität; sinnvolle Bitraten: 32, 40, 48, 56, 64, 80, 96, 112, 128, 160)

-B <bitrate> ... maximal erlaubte Bitrate

-f ... schneller Modus (niedrige Qualität)

-h ... hohe Qualität, ist aber relativ langsam

-m <mode> ... (s)tereo, (j)oint, (m)ono oder (a)uto; default ist (j) oder (s) - dies hängt von der Bitrate ab

lame --decode inputfile.mp3 -o outputfile.wav ... erstellt aus der MP3-Datei inputfile.mp3 die Wave-Datei outputfile.wav

cdparanoia 4;lame -h cdda.wav out.mp3;rm -f cdda.wav ... mit cdparanoia Track 4 von der eingelegten Audio-CD rippen und mit lame in eine MP3-Datei konvertieren; hohe Qualität, Stereo

cdparanoia 4;lame -m m -b 64 cdda.wav out.mp3;rm -f cdda.wav ... mit cdparanoia Track 4 von der eingelegten Audio-CD rippen und mit lame in eine MP3-Datei konvertieren; niedrige Qualität, Handy (Klingelsound), Mono

Da **lame** keine Wildcards (*) versteht, muss man für die Konvertierung von mehreren Dateien eine for-Schleife bemühen.

Wave-Dateien (z.B. track00.wav bis track25.wav) in Mp3-Dateien konvertieren:

```
for t in ${00..25}.wav;do lame -b 128 $t; done
```

siehe auch: lame --help, man lame, mp3splt, mp3gain

Ogg Vorbis

Es gibt viele verschiedene Encoder für Linux. Ogg Vorbis (oggenc) ist ein freies Format und liefert bei der Qualitätsstufe 6 in den meisten Fällen Transparenz. Dies bedeutet, dass Sie in der Regel keinen Unterschied zur Original-CD hören werden.

ogg123 track12.ogg ... spielt den angegebenen Titel; mit der Tastenkombination [Strg] + [C] kann der Vorgang vorzeitig abgebrochen werden

oggenc inputfile.wav ... erstellt aus der Wave-Datei inputfile.wav die Ogg Vorbis-Datei inputfile.ogg

oggenc inputfile.wav -o outputfile.ogg ... erstellt aus der Wave-Datei inputfile.wav die Ogg Vorbis-Datei outputfile.ogg

oggenc inputfile.wav -q 6 -o outputfile.ogg ... erstellt aus der Wave-Datei inputfile.wav die Ogg Vorbis-Datei outputfile.ogg, Qualitätsstufe 6, Qualitätsstufen 0 ... 10 möglich (0 .. niedrig, 10 .. hoch)

cdparanoia 4;oggenc cdda.wav -q 8 -o out.ogg;rm -f cdda.wav ... mit cdparanoia Track 4 von der eingelegten Audio-CD rippen und mit oggenc in eine OGG-Datei konvertieren; hohe Qualität, Stereo; anschließend wird die Wave-Datei cdda.wav gelöscht

cdparanoia 4;oggenc cdda.wav --downmix -q 2 -o out.ogg;rm -f cdda.wav ... mit cdparanoia Track 4 von der eingelegten Audio-CD rippen und mit oggenc in eine OGG-Datei konvertieren; niedrige Qualität, Webseiten, Mono; anschließend wird die Wave-Datei cdda.wav gelöscht

oggdec *.ogg ... wandelt alle Ogg-Vorbis-Dateien des aktuellen Verzeichnisses in WAVE-Dateien um, z.B. als Vorbereitung zur Erzeugung einer Audio-CD

Alle WAVE-Dateien im aktuellen Verzeichnis werden in OGG-Dateien (Stereo zu Mono .. --downmix, niedrige Qualität .. -q 1) konvertiert. Der Basisnamen bleibt dabei erhalten, nur die Datei-Endung wird in .mp3 umgewandelt.

```
for i in $(ls -l *.wav);do oggenc ${i} -q 1 --downmix -o "$(basename $i) .wav).ogg";done
```

siehe auch: oggenc --help, man oggenc, man oggdec, oggdec --help

WAV

Wav ist meist ein unkomprimiertes Audioformat, das zuerst in der Windows-Welt eingesetzt wurde.

aplay track4.wav ... spielt den angegebenen Titel; mit der

Tastenkombination [Strg] + [C] kann der Vorgang vorzeitig abgebrochen werden; siehe auch: play --help

cdparanoia -B -z 4 ... erstellt vom Track 4 eine Wave-Datei und speichert sie im aktuellen Verzeichnis

cdparanoia -B -z "4-6" ... erstellt von den Track's 4 bis 6 jeweils eine Wave-Datei und speichert sie im aktuellen Verzeichnis

cdparanoia -Bz ... erstellt von allen Track's der Audio-CD Wave-Dateien und speichert sie im aktuellen Verzeichnis

cdparanoia 1-15 ... speichert die Track's 1 bis 15 in eine große Wave-Datei

siehe auch: cdparanoia --help

Hinweis: Cdparanoia liest die CDs im Paranoia-Modus aus. Dies bedeutet, dass cdparanoia jeden Sektor der CD mindestens zweimal ausliest. Sind die Ergebnisse unterschiedlich, wird der Sektor noch mal ausgelesen. Das macht den Paranoia-Modus langsamer als den Burst-Modus, sorgt aber für bessere Ergebnisse.

Hinweis: Das Programm »cdparanoia« sucht sich selbstständig ein CD-Laufwerk mit eingelegter Audio-CD.

FLAC

Ein weiteres Format ist der Free Lossless Audio Codec (FLAC). Wie der Name schon andeutet, komprimiert dieser Codec die Dateien verlustfrei, ähnlich ZIP/RAR, nur besser, da er für die Kompression von Audiodaten optimiert ist. Sie sparen zwar nicht so viel Platz wie bei verlustbehafteten Formaten wie MP3 und Ogg Vorbis, haben dafür aber ein exaktes Abbild der Original-CD.

play track1.flac ... spielt den angegebenen Titel; mit der Tastenkombination [Strg] + [C] kann der Vorgang vorzeitig abgebrochen werden; siehe auch: play --help

flac *.wav ... wandelt alle WAVE-Dateien des aktuellen Verzeichnisses in FLAC-Dateien um

flac inputfile.wav --output-name=outputfile.flac ... erstellt aus der Wave-Datei inputfile.wav die FLAC-Datei outputfile.flac

flac *.wav --compression-level-8 ... wandelt alle WAVE-Dateien des aktuellen Verzeichnisses in FLAC-Dateien um; Stufe 8 ist die höchste Kompressionsstufe; Stufe 0 ist die niedrigste Kompressionsstufe; ohne Angabe wird als Vorgabe die Kompressionsstufe 5 verwendet

flac -d --output-name=track3.wav track1.flac ... konvertiert die Datei track1.flac ins WAVE-Format

siehe auch: man flac

Opus

Das Audio-Format Opus ist ein Datenformat zur verlustbehafteten Audiodatenkompression mit spezieller Eignung für interaktive Echtzeitanwendungen über das Internet. Es ist geeignet für die Sprachtelefonie und das Audio-Streaming.

Da Opus, wie auch MP3, nicht verlustfrei arbeitet, kann es die Formate FLAC oder WAV nicht ersetzen.

Opus unterstützt Bitraten zwischen 6 Kbit/s (für schmalbandige Sprachübertragung) und 512 Kbit/s (hochwertige Stereo-Übertragung von Musik), Samplingraten zwischen 8 kHz und 48 kHz und Frame-Raten zwischen 2,5 und 60 Millisekunden.

In den Linux-Distributionen Ubuntu und in den Distributionen die auf Ubuntu basieren (Linux Mint, Xubuntu, Lubuntu ...), finden sie Opus in dem Paket opus-tools.

Anmerkung: Das Format ist ein von der Internet Engineering Task Force (IETF) abgesegneter, weltweiter Standard (2012) zur Audiokodierung im Internet (Opus ist fester Bestandteil von HTML5).

Die opus-tools (sudo apt-get install libopus0 opus-tools) sind ein Paket mit nützlichen Kommandozeilenwerkzeugen zum Umgang mit Opus-Dateien und -Datenströmen. Nach der Installation stehen die Einzelanwendungen **opusenc**, **opusdec** und **opusinfo** zur Verfügung.

opusenc

opusenc [OPTIONEN] EINGABE AUSGABE

Es können ausschließlich WAVE, AIFF und PCM-Rohdaten verarbeitet werden. Für Ein- und Ausgabe können Dateien oder Standard-Datenströme (gekennzeichnet durch ein -) genutzt werden. Die wichtigste Option ist wahrscheinlich die Angabe einer Bitrate.

opusenc DATEI.wav DATEI.opus ... Wave-Datei in eine Opus-Datei konvertieren, Bitrate 96 kbps (Vorgabe)

opusenc --bitrate 128 DATEI.wav DATEI.opus ... Wave-Datei in eine Opus-Datei konvertieren; Bitrate 128 kbps

opusenc --hard-cbr --bitrate 160 DATEI.wav DATEI.opus ... eine

konstante (CBR) statt einer variablen Bitrate (VBR) verwenden

flac -c -d DATEI.flac | opusenc --bitrate 192 - DATEI.opus ... eine FLAC-Datei umwandeln; Bitrate 192 kbps

mpg123 -s "06 - Me And You.mp3" | opusenc --bitrate=48 --raw - "Me-and-You.opus" ... eine MP3-Datei umwandeln; Bitrate 48 kbps

Neben diesen Parametern zur Konvertierung können auch Metadaten in Opus-Dateien geschrieben werden. Grundlage bilden die Metadaten-Spezifikationen von Ogg/Vorbis.

Beispiele: Metadaten hinzufügen

--artist NAME ... Künstlername oder Gruppe

--title TITEL ... Songtitel

--album NAME ... Albumname

--genre GENRE ... Musikrichtung

opusdec

Das Gegenstück opusdec wandelt Opus in PCM-Daten. Die Ausgabe wird in der Standardeinstellung an die Soundkarte weitergeleitet.

opusdec DATEI.opus ... abspielen der angegebenen Datei

opusdec DATEI.opus DATEI.wav ... Konvertierung einer Opus-Datei in eine Wave-Datei erfolgt durch Angabe eines Dateinamens

opusdec "Me and You.opus" - | pacat ... abspielen der Opus-Datei mit pacat; pacat ist ein Programm aus dem Paket pulseaudio-utils, es kann mit Audio-Daten im raw-Format umgehen

opusinfo

opusinfo gibt zu einer oder auch mehreren Opus-Dateien technische und statistische Informationen aus und prüft Datenströme auf Fehler.

opusinfo DATEI.opus ... technische und statistische Informationen zur angegebenen Datei ausgeben

siehe auch: man opusenc, man opusdec, man opusinfo

* * * * *

SSH - secure shell

Das OpenSSH-Paket

Nach der Installation des OpenSSH-Paketes stehen die Programme ssh, scp

und sftp als Alternative für telnet, rlogin, rsh, rcp und ftp zur Verfügung.

Der SSH Daemon (sshd) - die Serverseite

Damit ssh und scp, die Client-Programme des SSH-Paketes, eingesetzt werden können, muss im Hintergrund der SSH-Daemon, ein Server, laufen. Dieser erwartet seine Verbindungen auf TCP/IP Port 22.

Während des ersten Starts generiert der Daemon drei Schlüsselpaare. Die Schlüsselpaare bestehen aus einem privaten und einem öffentlichen (engl. public) Teil. Deshalb bezeichnet man dies als ein public-key basiertes Verfahren. Um die Sicherheit der Kommunikation mittels SSH zu gewährleisten, darf ausschließlich der Systemadministrator die Dateien der privaten Schlüssel einsehen können. Die Dateirechte werden per Voreinstellung entsprechend restriktiv gesetzt. Die privaten Schlüssel werden lediglich lokal vom SSH-Daemon benötigt und dürfen an niemanden weitergegeben werden. Demgegenüber werden die öffentlichen Schlüsselbestandteile (an der Namensendung .pub erkennbar) an Kommunikationspartner weitergegeben und sind entsprechend für alle Benutzer lesbar.

Eine Verbindung wird vom SSH-Client eingeleitet. Der wartende SSH-Daemon und der anfragende SSH-Client tauschen Identifikationsdaten aus, um die Protokoll- und Softwareversion abzugleichen und die Verbindung zu einem falschen Port auszuschließen. Da ein Kindprozess des ursprünglichen SSH-Daemons antwortet, sind gleichzeitig viele SSH-Verbindungen möglich.

OpenSSH unterstützt zur Kommunikation zwischen SSH-Server und SSH-Client das SSH-Protokoll in den Versionen 1 und 2.

Bei Verwendung der SSH Protokoll-Version 1 sendet der Server sodann seinen öffentlichen host key und einen stündlich vom SSH-Daemon neu generierten server key. Mittels beider verschlüsselt (engl. encrypt) der SSH-Client einen von ihm frei gewählten Sitzungsschlüssel (engl. session key) und sendet ihn an den SSH-Server. Er teilt dem Server zudem die gewählte Verschlüsselungsmethode (engl. cipher) mit.

Die SSH Protokoll-Version 2 kommt ohne den server key aus. Stattdessen wird ein Algorithmus nach Diffie-Hellman verwendet, um die Schlüssel auszutauschen.

Die zur Entschlüsselung des Sitzungsschlüssels zwingend erforderlichen privaten host und server keys, können nicht aus den öffentlichen Teilen abgeleitet werden. Somit kann allein der kontaktierte SSH-Daemon mit seinen privaten Schlüsseln den Sitzungsschlüssel entziffern (vgl. man /usr/share/doc/packages/openssh/RFC.nroff). Diese einleitende Phase der

Verbindung kann man mittels der Fehlersuchoption `-v` des SSH-Clientprogramms gut nachvollziehen. Per Default wird SSH Protokoll-Version 2 verwendet, man kann jedoch mit dem Parameter `-1` auch die SSH Protokoll-Version 1 erzwingen. Indem der Client alle öffentlichen host keys nach der ersten Kontaktaufnahme in `/.ssh/known_hosts` ablegt, können so genannte »man-in-the-middle«-Angriffe unterbunden werden. SSH-Server, die versuchen, Name und IP-Adresse eines anderen vorzutäuschen, werden durch einen deutlichen Hinweis enttarnt. Sie fallen entweder durch einen gegenüber `/.ssh/known_hosts` abweichenden host-Schlüssel auf, oder können mangels passendem privaten Gegenstück den vereinbarten Sitzungsschlüssel nicht entschlüsseln.

Es empfiehlt sich, die in `/etc/ssh/` abgelegten privaten und öffentlichen Schlüssel extern und gut geschützt zu archivieren. So können Änderungen der Schlüssel festgestellt und nach einer Neuinstallation die alten wieder eingespielt werden. Dies erspart den Benutzern die beunruhigende Warnung. Ist es sichergestellt, dass es sich trotz der Warnung um den korrekten SSH-Server handelt, muss der vorhandene Eintrag zu diesem System aus `/.ssh/known_hosts` entfernt werden.

Das ssh-Programm

Mit dem ssh-Programm können Sie Verbindung zu einem entfernten System aufnehmen und dort interaktiv arbeiten. Es ist somit gleichermaßen ein Ersatz für `telnet` und `rlogin`. Aufgrund der Verwandtschaft zu `rlogin` zeigt der zusätzliche symbolische Name `slogin` ebenfalls auf `ssh`. Zum Beispiel kann man sich mit dem Befehl `ssh sonne` auf dem Rechner `sonne` anmelden. Anschließend wird man nach seinem Passwort auf dem System `sonne` gefragt.

Nach erfolgreicher Authentifizierung kann dort auf der Kommandozeile gearbeitet werden. Sollten sich der lokale Benutzername und der auf dem entfernten System unterscheiden, kann ein abweichender Name angegeben werden, z.B. `ssh -l august sonne` oder `ssh august@sonne`. Darüber hinaus bietet `ssh` die von `rsh` bekannte Möglichkeit, Kommandos auf einem anderen System auszuführen.

Im nachfolgenden Beispiel wird das Kommando `uptime` auf dem Rechner `sonne` ausgeführt und ein Verzeichnis mit dem Namen `tmp` angelegt. Die Programmausgabe erfolgt auf dem lokalen Terminal des Rechners `erde`.

Hinweis: Mit `logout` oder `exit` die SSH-Verbindung unterbrechen.

ssh sonne "uptime; mkdir tmp" bzw.


```
ssh <Benutzername>@<Ip-Nummer des Rechners sonne> 'uptime;  
mkdir tmp'  
tux@sonne's password:
```

1:21pm up 2:17, 9 users, load average: 0.15, 0.04, 0.02

Anführungszeichen sind hier zum Zusammenfassen der beiden Anweisungen in einem Kommando erforderlich. Nur so wird auch der zweite Befehl auf dem Rechner sonne ausgeführt.

scp - sicheres Kopieren

Mittels scp kopieren Sie Dateien auf einen entfernten Rechner. scp ist der sichere, verschlüsselte Ersatz für rcp. Zum Beispiel kopiert **scp MeinBrief.tex sonne:** die Datei MeinBrief.tex vom Rechner erde auf den Rechner sonne. Insoweit sich die beteiligten Nutzernamen auf erde und sonne unterscheiden, geben Sie bei scp die Schreibweise **Nutzername@Rechnername** an. Eine Option -l existiert nicht. Nachdem das Passwort eingegeben wurde, beginnt scp mit der Datenübertragung und zeigt dabei den Fortschritt anhand eines von links nach rechts anwachsenden Balkens aus Sternen an. Zudem wird am rechten Rand die geschätzte Restübertragungszeit (engl. estimated time of arrival) angezeigt.

Jegliche Ausgabe kann durch die Option **-q** unterdrückt werden. scp bietet neben dem Kopieren einzelner Dateien ein rekursives Verfahren zum Übertragen ganzer Verzeichnisse:

scp -r /tmp/Backup backup@192.168.1.100:~ ... kopiert das Verzeichnis Backup des Rechners 192.168.1.200 zum Rechner 192.168.1.100 ins Home-Verzeichnis des Benutzers backup

scp -r /tmp/Backup backup@192.168.1.100:/home/backup/ ... kopiert das Verzeichnis Backup des Rechners 192.168.1.200 zum Rechner 192.168.1.100 ins Home-Verzeichnis des Benutzers backup

scp -r ./Backup backup@192.168.1.100:/home/backup/ ... kopiert das Verzeichnis Backup (Ausgangspunkt ist der aktuelle Standort ./) des Rechners 192.168.1.200 zum Rechner 192.168.1.100 ins Home-Verzeichnis des Benutzers backup

scp backup@192.168.1.200:/home/backup/histo.txt ./history.txt ... kopiert die Datei histo.txt vom Rechner 192.168.1.200 zum Rechner 192.168.1.100 unter dem Namen history.txt ins aktuelle Verzeichnis

scp -r /tmp/Backup/* backup@192.168.1.100:/home/backup/ ... kopiert nur den **Inhalt** des Verzeichnisses Backup des Rechners 192.168.1.200 zum Rechner 192.168.1.100 ins Home-Verzeichnis des Benutzers backup

Fehlende Unterverzeichnisse auf dem Zielrechner werden automatisch angelegt, falls sie nicht existieren.

Mittels der Option **-p** kann scp die Zeitstempel der Dateien erhalten. **-C** sorgt für eine komprimierte Übertragung. Dadurch wird einerseits das zu übertragende Datenvolumen minimiert, andererseits aber ein höherer Rechenaufwand erforderlich.

ssh -XC <Benutzername>@<Rechnername oder IP-Adresse>

Nach der Anmeldung können Programme aufgerufen werden. Die Option **-X** bewirkt, das SSH X-Forwarding einzusetzt. Es zeigt dann die Oberfläche eines Programms auf dem lokalen Bildschirm an, obwohl das Programm auf einen anderen Rechner läuft. Mit **-C** komprimiert ssh die Daten, bevor sie über die Leitung gehen, spart Bandbreite.

Hinweis: Damit das X-Forwarding funktioniert, muss der Server - also der entfernte Rechner - dies erlauben. In der Konfigurations-Datei **/etc/ssh/sshd_config** muss der Eintrag auf

X11Forwarding yes

stehen. Steht hier ein **no** oder ist die Zeile durch eine Raute (#) auskommentiert, so bearbeiten Sie die Datei als Benutzer root.

sftp - sicherere Dateiübertragung

Alternativ kann man zur sicheren Datenübertragung sftp verwenden. sftp bietet innerhalb der Sitzung viele der von ftp bekannten Kommandos. Gegenüber scp mag es vor allem beim Übertragen von Daten, deren Dateinamen unbekannt sind, von Vorteil sein.

sftp [[user@]host[:file [file]]]

sftp [[user@]host[:dir|/]]

sftp ftp.server.de ... Verbindung mit einem FTP-Server herstellen z.B. **ftp.server.de**

ls	Anzeige des Inhaltsverzeichnisses
cd <Zielverzeichnis>	Verzeichniswechsel auf dem Server

ls	Anzeige des Inhaltsverzeichnisses
lcd <Zielverzeichnis>	Verzeichniswechsel auf dem Client
get <Datei>	Angegebene Datei vom Server laden.
get <Datei(en)>	Mehrere Dateien vom Server laden, Wildcards * und ? sind erlaubt.
put <Datei>	Datei zum Server übertragen.
put <Datei(en)>	Mehrere Dateien zum Server übertragen, Wildcards * und ? sind erlaubt.
quit	Programm beenden.

SSH-Authentifizierungsmechanismen

Jetzt erfolgt die eigentliche Authentifizierung, die in ihrer einfachsten Weise aus der Eingabe eines Passwortes besteht, wie es in den oben aufgezeigten Beispielen erfolgte. Ziel von SSH war die Einführung einer sicheren, aber zugleich einfach zu nutzenden Software. Wie bei den abzulösenden Programmen rsh und rlogin muss deshalb auch SSH eine im Alltag einfach zu nutzende Authentifizierungsmethode bieten. SSH realisiert dies mittels eines weiteren hier vom Nutzer erzeugten Schlüsselpaares. Dazu liefert das SSH-Paket das Hilfsprogramm ssh-keygen mit. Nach der Eingabe von ssh-keygen -t rsa oder ssh-keygen -t dsa wird das Schlüsselpaar generiert und der Basisdateiname zur Ablage der Schlüssel erfragt:

Enter file in which to save the key (/home/tux /.ssh/id_rsa):

Bestätigen Sie die Voreinstellung und beantworten Sie die Frage nach einer Passphrase. Auch wenn die Software eine leere Passphrase nahelegt, sollte bei der hier vorgeschlagenen Vorgehensweise ein Text von zehn bis 30 Zeichen Länge gewählt werden. Verwenden Sie möglichst keine kurzen und einfachen Worte oder Sätze. Nach erfolgter Eingabe wird zur Bestätigung eine Wiederholung der Eingabe verlangt. Anschließend wird der Ablageort des privaten und öffentlichen Schlüssels, in unserem Beispiel der Dateien id_rsa und id_rsa.pub, ausgegeben.

Enter same passphrase again:

Your identification has been saved in /home/tux /.ssh/id_rsa

Your public key has been saved in /home/tux /.ssh/id_rsa.pub.

The key fingerprint is:

79:c1:79:b2:e1:c8:20:c1:89:0f:99:94:a8:4e:da:e8 tux@sonne

Verwenden Sie **ssh-keygen -p -t rsa** bzw. **ssh-keygen -p -t dsa**, um Ihre alte Passphrase zu ändern. Kopieren Sie den öffentlichen Teil des Schlüssels (in unserem Beispiel id_rsa.pub) auf den entfernten Rechner und legen Sie ihn dort als **/.ssh/authorized_keys** ab. Zur Authentifizierung werden Sie

beim nächsten Verbindungsaufbau nach Ihrer Passphrase gefragt. Sollte dies nicht der Fall sein, überprüfen Sie bitte Ort und Inhalt der zuvor erwähnten Dateien.

Auf Dauer ist diese Vorgehensweise mühsamer, als die Eingabe eines Passwortes. Entsprechend liefert das SSH-Paket ein weiteres Hilfsprogramm, den `ssh-agent`, der für die Dauer einer »X-session« private Schlüssel bereit hält. Dazu wird das gesamte X als Kindprozess des `ssh-agent` gestartet. Sie erreichen dies am einfachsten, indem Sie am Anfang der Datei `.xsession` die Variable `usessh` auf `yes` setzen und sich über einen Displaymanager, z.B. KDM oder XDM, anmelden. Alternativ können Sie `ssh-agent startx` verwenden.

Nun können Sie wie gewohnt `ssh` oder `scp` nutzen. Insoweit Sie Ihren öffentlichen Schlüssel wie zuvor verteilt haben, sollten Sie jetzt nicht mehr nach dem Passwort gefragt werden. Achten Sie beim Verlassen Ihres Rechners darauf, dass Sie Ihre X-session beenden oder mittels einer passwortgeschützten Bildschirmsperre, z.B. `xlock`, verriegeln.

Authentifizierungs- und sonstige Weiterleitung

Über die bisher beschriebenen sicherheitsrelevanten Verbesserungen hinaus erleichtert `ssh` auch die Verwendung von entfernten X-Anwendungen. Insoweit Sie `ssh` mit der Option **-X** aufrufen, wird auf dem entfernten System automatisch die `DISPLAY`-Variable gesetzt und alle X-Ausgaben werden durch die bestehende `ssh`-Verbindung auf den Ausgangsrechner weitergeleitet. Diese bequeme Funktion unterbindet gleichzeitig die bisher bestehenden Abhörmöglichkeiten bei entfernt aufgerufenen und lokal betrachteten X-Anwendungen.

Durch die gesetzte Option **-A** wird der Mechanismus zur Authentifizierung des `ssh-agent` auf den nächsten Rechner mit übernommen. Man kann so von einem Rechner zum anderen gehen, **ohne ein Passwort** eingeben zu müssen. Allerdings nur, wenn man zuvor seinen öffentlichen Schlüssel auf die beteiligten Zielrechner verteilt und korrekt abgelegt hat.

Beide Mechanismen sind vorsichtshalber in der Voreinstellung deaktiviert, können jedoch in der systemweiten Konfigurationsdatei `/etc/ssh/ssh_config` oder der nutzereigenen `~/.ssh/config` permanent eingeschaltet werden.

Man kann `ssh` auch zur beliebigen **Umleitung von TCP/IP-Verbindungen** benutzen. Als Beispiel sei hier die Weiterleitung des SMTP- und POP3-Ports aufgeführt:

ssh -L 25:erde:25 sonne

Hier wird jede Verbindung zu `sonne` (»SSH-Server«, d.h. der entfernte

Zielrechner) Port 25, SMTP auf den SMTP-Port von erde (»Quellrechner«, d.h. der lokale Rechner) über den verschlüsselten Kanal zu sonne weitergeleitet. Dies ist insbesondere für Nutzer von SMTP-Servern ohne SMTP-AUTH oder POP-before-SMTP-Fähigkeiten von Nutzen. Mail kann so von jedem beliebigen Ort mit Netzanschluss zur Auslieferung durch den heimischen Mailserver übertragen werden. Analog können mit folgendem Befehl alle POP3-Anfragen (Port 110) an erde auf den POP3-Port von sonne weitergeleitet werden:

ssh -L 110:erde:110 sonne

Beide Beispiele müssen Sie als **Benutzer root** ausführen, da auf privilegierte, lokale Ports verbunden wird. Bei bestehender SSH-Verbindung wird Mail wie gewohnt als normaler Benutzer versandt und abgeholt. Der SMTP- und POP3-Host muss dabei auf localhost konfiguriert werden. Zusätzliche Informationen entnehmen Sie den Manualpages der einzelnen Programme und den Dateien unter /usr/share/doc/packages/openssh.

Anmerkung: Falls kein Domain Name Server installiert ist, so ist statt des Rechnernamens die IP-Nummer zu verwenden.

Hinweis: Mit **logout** oder **exit** die SSH-Verbindung unterbrechen.

Name Based Hosting und SSH

Jeden trifft es irgendwann: Der gewünschte Webserver steht hinter einer restriktiven Firewall und hört auf einen exotischen Port z.B: 8443. Eine einfache Lösung ist ein SSH-Tunnel.

ssh -L 8443:<IP-Nummer lokal>:8443 <Ip-Nummer entfernt>

Bei abweichenden Benutzernamen gibt man folgendes ein:

ssh -L 8443:<IP-Nummer lokal>:8443 <Benutzer>@<Ip-Nummer entfernt>

Nun gibt man im lokalen Browser statt

https://<Ip-Nummer entfernt>:8443

die URL

https://localhost:8443

ein. Die gewünschte Seite des Webservers erscheint aber nur, wenn der entfernte Webserver seine Webseite anhand der IP-Adresse erkennt. Die meisten Systeme nutzen aber Name Based Hosting und identifizieren die Webseite anhand eines zusätzlichen Headers, der erst seit HTTP 1.1 zum Standard gehört. Im Host-Header steht der gewünschte Zielrechner. Den Namen ermittelt der Browser (leider) ganz einfach: Er nimmt den in seiner URL-Zeile stehenden Rechnernamen. Bei dem SSH-Tunnel steht da aber nicht mehr »Zielrechner«, sondern »localhost«.

Einen Ausweg bahnt die »/etc/hosts«-Datei, die der Rechner vor dem DNS zur Namensauflösung verwendet. Modifiziert root dort die »localhost«-Zeile so, dass der eigene genauso wie der Zielrechner heißt, klappt es:

127.0.0.1 localhost <Zielrechner>

Wenn man jetzt in die Adresszeile des Browsers wieder

http://<Ip-Nummer entfernt>:8443

einträgt, wird er dank der Namensauflösung mit SSH auf 127.0.0.1 verbunden, sendet selbst aber den passenden Host im HTTP-Header.

Sicherheitsdatenbank »known_hosts«

Der Client speichert den Host-Key in der Textdatei »~/.ssh/known_hosts«. In den Drafts zu SSH 2 ist festgelegt, dass SSH-Clients bei bislang unbekannten Servern eine Bestätigung des Anwenders einholen müssen, ob er wirklich eine Verbindung dorthin wünscht. Verneint der Anwender das, bricht die Verbindung ab.

Der User sollte hier nicht leichtfertig »yes« tippen, diese Abfrage ist schließlich eines der wesentlichen Sicherheits-Features von SSH. Der angezeigte Fingerprint ist bestens geeignet, um den Key durch ein Telefonat mit dem Admin des Ziel-Hosts zu verifizieren: Der Admin kann sich den Fingerprint des originalen Host-Keys mit »ssh-keygen -l -f <Keyfile>« ansehen. Nur wenn beide Zeichenfolgen übereinstimmen, sollte der User den Key in seine »known_hosts«-Datei aufnehmen.

Das Prüfen des Server-Host-Keys schützt vor Man-in-the-Middle-Attacken. Dabei gibt sich ein Angreifer beispielsweise durch Manipulation des DNS, von ARP oder durch die Übernahme der IP des echten Servers für diesen aus. Der Angreifer verbindet sich seinerseits mit dem echten Server und leitet die Daten weiter, so dass der Benutzer keinen Unterschied bemerkt. Vor diesem Angriff kann die Kryptographie ihre Anwender schützen - aber nur, wenn diese auch mitspielen. Wenn der Client nichts über den Server weiß, kann der dessen Authentizität auch nicht verifizieren.

Wenn der SSH-Client den Public Key des echten Servers bereits kennt, der

Schlüssel also in der »known_hosts«-Datei steht, kann der Client den Angriff automatisch erkennen. Der Angreifer kennt den originalen Secret Key nicht und kann daher auch nicht den Public Key des gewünschten Ziels verwenden, vielmehr muss er seinen eigenen öffentlichen Schlüssel senden. Beim Vergleich mit seiner Schlüsselliste stellt der Client die Abweichung fest, warnt den User ausdrücklich und bricht die Verbindung ab.

Die Warnmeldung kann aber auch einen harmlosen Anlass haben, wenn sich der Server-Key tatsächlich geändert hat. Das passiert, wenn der Admin den Key neu generiert, weil beispielsweise die alte Festplatte defekt und kein Backup vorhanden ist, der Rechner ausgetauscht wurde oder einfach nur jemand SSH neu installiert hat, ohne den alten Schlüssel zu übernehmen. Die Warnung des Clients bleibt so lange bestehen, bis der User den alten Server-Key aus »known_hosts« löscht. Bei der nächsten Kontaktaufnahme wird der Dialog zur Bestätigung des Servers dann erneut durchlaufen.

Agenten weiterleiten

Loggt man sich hintereinander auf mehreren Hosts ein, kann die »ForwardAgent«-Option behilflich sein. Damit muss nicht auf jeder Zwischenstation eine Kombination aus privatem und öffentlichem Schlüssel liegen und ein weiterer Agenten-Prozess gestartet werden. Ein einzelner SSH-Agent auf dem vertrauenswürdigen Rechner genügt; alle Hosts, zu denen man sich weiter verbindet, benutzen rückwärts den »ssh-agent« am Anfang der Kette.

Das Agent Forwarding lässt sich auf drei Arten aktivieren: systemweit als Eintrag »ForwardAgent yes« in der Datei »/etc /ssh/ssh_config«, für einzelne User im File »~/.ssh/config« oder durch die Option »-A« des »ssh«-Kommandos.

Das Verfahren hat aber auch seine Schattenseiten. Prinzipiell kann Root durch einen Speicherdump die entschlüsselten Private Keys lesen - wer dem Root-Account nicht vertraut, sollte auf der jeweiligen Maschine aber generell keine Geheimnisse speichern.

Selbst wenn man auf den SSH-Agent verzichtet, kann Root durch einen trojanisierten SSH-Client oder mit Hilfe eines TTY-Sniffers an die geheimen Daten kommen, sobald der User die Passphrase eintippt.

Backup über SSH

Besonders erfreulich für Administratoren ist die Möglichkeit, Backups über SSH durch das Netz zu schicken. Dafür ist gar kein »scp« oder »sftp« nötig, SSH lässt sich direkt in der Shell-Pipe nutzen:

tar -czvf - /das/Verzeichnis | ssh user@host cat /tmp/foo.tar.gz

Die Empfängerseite kann auch direkt auf ein Bandlaufwerk schreiben:

tar -cvf - /das/Verzeichnis | ssh user@host dd of=/dev/tape

Dabei kann die Performance des Bandlaufwerks allerdings ziemlich in den Keller gehen. Der Grund dafür ist, dass »dd« und Bandlaufwerk wechselseitig auf Daten warten müssen. Reißt der Strom neuer Daten ab, muss das Laufwerk absetzen, etwas zurückspulen und kann dann erst weiter schreiben. Ein kleines Tool hilft hier:

tar -cvf - /das/Verzeichnis | buffer | ssh -c blowfish root@vaio buffer -o /dev/tape

Buffer teilt sich in zwei Prozesse, die das Lesen der Daten vom Netz und das Schreiben auf Band entkoppeln und durch einen Zwischenpuffer ergänzen. In diesem Beispiel schaltet zudem die OpenSSH-Option »-c blowfish« auf den sehr schnellen und trotzdem als sicher geltenden Verschlüsselungsalgorithmus Blowfish um. Damit erfüllt OpenSSH die oft gegensätzlichen Forderungen nach Sicherheit und Schnelligkeit.

Hinweis: Mit **logout** oder **exit** die SSH-Verbindung unterbrechen.

Anmerkung: Falls eine SSH-Verbindung nicht zustande kommt, liegt dies möglicherweise an den Firewall-Einstellungen.

siehe auch: man ssh, chroot

* * * * *

Sniffer

siehe auch: tcpdump

sort

Mit dem Kommandozeilen-Tool sort erfolgt eine sortierte Ausgabe z.B. von Dateiinhalten auf dem Bildschirm.

sort [OPTION] [DATEI]

-b, --ignore-leading-blanks ... führende Leerzeichen ignorieren

-d, --dictionary-order ... nur Leer- und alphanumerische Zeichen beachten

-f, --ignore-case ... Klein- als Großbuchstaben behandeln

-g, --general-numeric-sort ... anhand des allgemeinen numerischen Wertes sortieren
-i, --ignore-nonprinting ... nur druckbare Zeichen beachten
-n, --numeric-sort ... anhand des numerischen Werts sortieren
-r, --reverse ... das Ergebnis der Sortierung umkehren
-o, --output=DATEI ... Ergebnis in DATEI schreiben statt auf Standardausgabe
-z, --zero-terminated ... Zeilen mit Nullbyte beenden, nicht mit Zeilenvorschub

Achtung: Die länderspezifischen Einstellungen (deutsch, us-englisch etc.) beeinflussen die Sortierreihenfolge.

Beispiele:

sort adressen | uniq | less

Das sort-Kommando liest die Datei **adressen** und sortiert deren Inhalt zeilenweise. Die sortierten Zeilen, werden dem Programm **uniq** zugeführt, dass die Zeilen vergleicht und alle Zeilen entfernt, die doppelt vorkommen. Die Ausgabe von **uniq** wird wiederum dem Programm **less** zugeführt, dass die Adressendatei von den Doppeln befreit, sortiert anzeigt.

sort < unsortiert.txt

Die unsortierte Liste mit Begriffen - z.B. Städtenamen - werden an das Programm **sort** übergeben und auf der Standardausgabe, dem Bildschirm, ausgegeben.

du /home/<benutzername>/bin | sort -rn | less

Zeigt den Platzverbrauch des angegebenen Verzeichnisses an. Die beiden Parameter **-rn**, sorgen dafür, dass das Programm die größten Verzeichnisse oder Dateien zuerst anzeigt.

sort adressen | uniq > Adressenliste.txt

Im Beispiel wird eine sortierte Adressenliste, die von doppelten Einträgen befreit wurde, in der Datei **Adressenliste.txt** gespeichert.

find \${HOME} -type f -regex "^.*/backup.*(\.tgz\.gpg\|.tar\.gz)\$" | sort -gr

Im Home-Verzeichniss (globale Variable - **\${HOME}** - enthält das Home-Verzeichnis) des aktuellen Benutzers wird nach Dateien mit der Endung **.tgz.gpg** oder **.tar.gz** gesucht. Die Dateien werden anhand des allgemeinen numerischen Wertes (**-g**) sortiert, dabei wird die Sortierung auch noch umgekehrt (**-r**).

Sinnvoll für gepackte oder für gepackte und verschlüsselte Backup-Dateien in deren Dateinamen das Backup-Datum noch enthalten ist. Die neusten Backup-Dateien werden zuerst angezeigt.

ls | tr [:upper:] [:lower:] | grep -oP '\.[^\.]+\$' | sort | uniq -c | sort ... listet die Anzahl der einzelnen Dateitypen (txt, odt, jpg ...) auf und gibt am Schluss die Gesamtsumme aller Dateien (ohne Verzeichnisse) des aktuellen Verzeichnisses (ohne Unterverzeichnisse) aus

siehe auch: sort --help

stat

Status einer Datei oder eines Dateisystems anzeigen. Mit dem Befehl stat lassen sich Zugriffs- und Änderungs-Zeitstempel von Dateien und Ordnern anzeigen. Weiterhin werden Informationen zu Rechten, zu Besitzer und Gruppe und zum Dateityp ausgegeben.

stat [OPTION] ... DATEI ...

-f ... Dateisystemstatus anstelle von Dateistatus anzeigen (Blockgröße, Blockanzahl etc.); Blockgröße: der bei der Eingabe und Ausgabe mit einem mal vom Dateisystem gelesen bzw. geschrieben wird

Informationen über eine Datei anzeigen (Dateigröße, Anzahl der Blöcke die von der Datei verwendet wird, User-ID, Access: letzte Zugriffszeit auf die Datei, Modify: Zeitpunkt der letzten Änderung, Change: letzte Änderung der Dateirechte, ...):

```
karl@tux:~/$ stat datei.odt
File: „datei.odt“
Size: 25327    Blocks: 56          IO Block: 4096    reguläre
Datei
Device: 805h/2053    dInode: 526025      Links: 1
Access: (0644/-rw-r--r--)  Uid: ( 1000/   karl)   Gid:
( 1000/   karl)
Access: 2014-07-05 12:13:13.956000682 +0200
Modify: 2014-03-27 14:41:46.000000000 +0100
Change: 2014-03-27 14:50:39.323751610 +0100
```

Informationen über die Inodes, Blöcke und Blockgröße (der bei der Eingabe und Ausgabe mit einem mal vom Dateisystem gelesen bzw. geschrieben wird) einer Partition anzeigen:

```
karl@tux:~/$ stat -f /dev/sda1
```

```
File: "/dev/sda1"
ID: 0          Namelen: 255      Type: tmpfs
Block size: 4096      Fundamental block size: 4096
Blocks: Total: 252828  Free: 252749  Available: 252749
Inodes: Total: 218601  Free: 217758
```

Die Blockgröße beträgt hier 4096 Byte. Dies bedeutet aber auch, dass Dateien die kleiner sind als 4096 Byte immer einen ganzen Block auf dem Speichermedium (Festplatte, externe Festplatte, ...) belegen.

Beispiele:

stat -c '%a %U %G %s %n' * ... Anzeige des aktuellen Verzeichnisinhaltes wie mit dem Kommando `ls -l`, nur dass die Zugriffsrechte statt durch die Buchstaben `rwX` mit ihren oktalen Werten dargestellt werden

stat --printf='%a %h %U %G %s\t%.19y %n\n' * ... mit dieser Form wird die `stat`-Ausgabe dem Kommando `ls -l` noch ähnlicher

Natürlich sind diese Kommandos zu lang um sie bei Bedarf immer wieder in ein Terminal einzugeben. Mit einer Alias-Definition in der versteckten Datei `.bash_rc` kann man dieses Problem umgehen. Falls die versteckte Datei `.bash_rc` (Punkt-Dateien, im Dateimanager ist die Anzeige der versteckten Dateien zu aktivieren) im Home-Verzeichnis nicht existiert, so ist sie neu anzulegen und mit einem Texteditor zu bearbeiten.

Datei: `.bash_rc`

```
# stat-Kommando 1
```

```
alias lso1='stat --printf='%a %h %U %G %s\t%.19y %n\n' .'
```

```
# stat-Kommando 2
```

```
alias lso2='stat -c '%a %U %G %s %n' .'
```

Beim nächsten Öffnen, Neustart eines Terminals stehen die neuen Alias-Definitionen **lso1** und **lso2** zur Verfügung.

siehe auch: `stat --help`

steghide

Der Name Steganographie stammt aus dem Altgriechischen. Er setzt sich aus den Wörtern »steganos« (verdeckt) und »graphie« (schreiben) zusammen. In der frühen Geschichte der Steganographie wurden Geheimnisse meistens in Schriftform versteckt. Man rasierte Sklaven den Kopf kahl, tätowierte ihnen eine Botschaft auf die Kopfhaut und wartete, bis die Kopfhaut von den nachgewachsenen Haaren verdeckt wurde.

Aktuelle computergestützte Steganographie verdeckt Daten in Bildern oder Audiodateien. Bei Audiodateien macht man sich zunutze, dass sie immer ein nicht wahrnehmbares Rauschen enthalten. Dieses wird dahingehend verändert, dass man in ihnen die Daten verbirgt. Bei Bildern werden nicht wahrnehmbare Veränderungen in der Farbgebung ausgenutzt. Dabei ist es wichtig, dass kein »Bildrauschen« entsteht; dies würde sofort verraten, dass im Bild eine Datei verborgen ist.

Hinweis: Die EXIF-Daten von Bildern gehen durch die Verwendung von Steghide verloren.

Als Dateihülle für die versteckten Informationen, werden von Steghide folgende Dateiformate unterstützt: **AU**, **BMP**, **JPEG** und **WAV** (max. Dateigröße: etwa 300 MByte). Für die Dateihülle eignen sich besonders selbstproduzierte Sound-Dateien (plätschernder Bach, Laubbaum im Herbstwind, belebte Straßenkreuzung). Von diesen selbstproduzierten Sound-Dateien kann sich keine Organisation der Welt, für die Entschlüsselung, Vergleichsdateien beschaffen.

Auch sollte man bei höheren Sicherheitsanforderungen die von Steghide erzeugte Prüfsumme nicht verwenden. Prüfsummen verwenden einen typischen und verringerten Zeichensatz und dies ist möglicherweise hilfreich bei einer nicht autorisierten Entschlüsselung durch Dritte. Die Prüfsummen für die Dateien (Dateihülle und versteckte Datei) sollte durch ein anderes Programm erzeugt werden. Diese Prüfsummen sind getrennt von der Steganographie-Datei aufzubewahren.

Dasselbe gilt für den Dateinamen, der mit den Vorgabewerten immer in die Steghide-Datei verankert wird. Bei höheren Sicherheitsanforderungen, sollte daher auf die Einbettung des Dateinamen der versteckten Datei verzichtet werden (siehe weiter unten: Nr. 4).

Das von Steghide – beim Verpacken und Entpacken bzw. Verschlüsselung und Entschlüsselung - verlangte Passwort sollte ausreichend sicher sein. Das Passwort sollte aus mindestens 12 Zeichen bestehen, die als Wort in keinem Wörterbuch zu finden sind (**siehe auch:** pwgen).

Das Paket steghide muss bei allen bekannten Linux-Distributionen erst noch installiert werden.

steghide command [arguments]

steghide ... Kurzhilfe anzeigen

1. Die geheime Textdatei secret.txt in dem Bild picture.jpg verstecken.

steghide embed -cf picture.jpg -ef secret.txt

Enter passphrase:

Re-Enter passphrase:

embedding "secret.txt" in "picture.jpg"... done

-cf <Dateiname> ... cover file – Dateihülle für die zu versteckende Datei

-ef <Dateiname> ... embed file –Dateiname der zu versteckenden Datei

Reicht die Kapazität der Dateihülle nicht aus (Test: steghide info datei.jpg) so ist die zu versteckende Datei in kleinere Dateifragmente zu splitten (**siehe auch:** split).

Achtung: Bei der kleinste Änderung an der Dateihülle ist es mit Steghide nicht mehr möglich die versteckte Datei zu extrahieren. Selbst der unbedachte Klick auf das Speichern-Icon innerhalb eines Programms, führt zu kleineren nicht sichtbaren Änderungen an einer Datei. Die Komprimierung der erzeugten Steghide-Dateien mit den Standardwerkzeugen (GZIP, BZIP ...) ist in jedem Fall als unbedenklich anzusehen, d.h. die Steghide-Dateien werden durch eine Komprimierung nicht beschädigt.

2. Die versteckte Datei secret.txt aus der Dateihülle picture.jpg befreien, extrahieren.

steghide extract -sf picture.jpg

Enter passphrase:

wrote extracted data to "secret.txt"

-sf <Dateiname> ... stego file – die benannte Datei enthält eine versteckte Information

Steghide entpackt die Datei im aktuellen Verzeichnis, in dem sich die Stenographiedatei befindet.

3. Verschlüsselung einsetzen

steghide encinfo ... zeigt die von Steghide unterstützten Verschlüsselungsverfahren

steghide embed -e rijndael-128 -cf picture.jpg -ef secret.txt

-e <Verschlüsselungsverfahren> ... über diese Option wird das Verschlüsselungsverfahren angegeben (**rijndael-128**, **rijndael-256**, **twofish**, **serpent**, **blowfish**, ...)

Die Entschlüsselung erfolgt mit demselben extract-Kommando, mit dem auch unverschlüsselte Steghide-Dateien entpackt, extrahiert werden.

Hinweis: An Steghide können auch bereits verschlüsselte Dateien übergeben werden.

4. Verschlüsselung einsetzen – Prüfsumme durch ein externes Programm erzeugen, keinen Dateinamen verankern

steghide embed -e twofish -K -N -cf landscape_sound.wav -ef secret.txt

-K ... keine Prüfsumme für die Integritätsprüfung erstellen

-N ... den Dateinamen der versteckten Datei nicht in die Steghide-Datei verankern

steghide extract -sf landscape_sound.wav -xf secret.txt

-sf <Dateiname> ... stego file – die benannte Datei enthält eine versteckte Information

-xf <Dateiname> ... extract file – Dateiname der versteckten Datei, da der ursprüngliche Name nicht in der Steghide-Datei verankert wurde

Die Prüfsumme für die Integritätsprüfung kann man entweder mit den externen Programmen **md5sum** oder **sha1sum** erzeugen. Für sehr große Dateien ist **sha1sum** möglicherweise die bessere Wahl.

md5sum landscape_sound.wav > ./md5.txt ... Prüfsumme der Dateihülle mit md5sum erstellen und in der Datei md5.txt speichern; Dateihülle enthält bereits die Datei secret.txt

md5sum secret.txt >> ./md5.txt ... Prüfsumme der versteckten Datei mit md5sum erstellen und in der Datei md5.txt speichern

sha1sum landscape_sound.wav > ./sha1.txt ... Prüfsumme der Dateihülle mit Sha1sum erstellen und in der Datei sha1.txt speichern; Dateihülle enthält bereits die Datei secret.txt

sha1sum secret.txt >> ./sha1.txt ... Prüfsumme der versteckten Datei mit Sha1sum erstellen und in der Datei sha1.txt speichern

Überprüfung der Datei landscape_soundf.wav und secret.txt mit dem in der Datei md5.txt bzw. sha1.txt gespeicherten Prüfschlüssel.

md5sum landscape_sound.wav ... aktuelle Prüfsumme ist mit dem gespeicherten Wert zu vergleichen oder
pv landscape_sound.wav | md5sum ... Prüfsumme ermitteln – mit Fortschrittsanzeige
md5sum secret.txt ... aktuelle Prüfsumme ist mit dem gespeicherten Wert zu vergleichen

bzw.

md5sum -wc ./md5.txt ... die zu überprüfenden Dateien und die Datei md5.txt müssen sich in demselben Verzeichnis befinden; w .. warn, c .. check

sha1sum landscape_sound.wav ... aktuelle Prüfsumme ist mit dem gespeicherten Wert zu vergleichen oder
pv landscape_sound.wav | sha1sum ... Prüfsumme ermitteln – mit Fortschrittsanzeige
sha1sum secret.txt ... aktuelle Prüfsumme ist mit dem gespeicherten Wert zu vergleichen

bzw.

sha1sum -wc ./sha1.txt ... die zu überprüfenden Dateien und die Datei md5.txt müssen sich in demselben Verzeichnis befinden; w .. warn, c .. check

Eine Datei ist nicht mehr als vertrauenswürdig anzusehen, sobald die aktuell ermittelte Prüfsumme sich von der gespeicherten Prüfsumme unterscheidet.

Hinweis: Die Datei mit der gespeicherten Prüfsumme ist getrennt von der Steganographie-Datei aufzubewahren.

5. Informationen über eine Steghide-Datei anzeigen.

steghide info received_file.wav
"received_file.wav":
format: wave audio, PCM encoding
capacity: 3.5 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
embedded file "secret.txt":
size: 1.6 KB

encrypted: rijndael-128, cbc
compressed: yes

Außerdem lässt sich mit steghide info im vorab ermitteln, wie groß die Kapazität der Dateihülle in kByte ist.

5. Kompression verwenden

Um Dateien komprimiert zu verstecken, ist das Argument -z zusätzlich anzugeben.

steghide embed -z 9 -cf picture.jpg -ef secret.txt

-z 5 ... 1 .. niedrige Kompression, 9 .. höchste Kompression

Anmerkung: Es existieren Programme die mit einiger Sicherheit erkennen können, ob in einer Datei versteckte Informationen enthalten sind.

siehe auch: man steghide

streamripper

Mit streamripper ist es möglich Sendungen eines Internet-Radiosenders aufzuzeichnen.

Beispiel: Mitschnitt eines Internet-Radiosenders – mukulcast.com

Das koreanische Internet-Radio (Mukulcast, K-POP) wird mit streamripper mitgeschnitten. Der Radiomitschnitt ist ein einziger geschlossener Datenstream, einschließlich der Moderation.

Das Programm streamripper legt den Datenstream und die cue-Datei im existierenden Verzeichnis InternetRadio ab. Das Terminalprogramm streamripper wird mit der Tastenkombination **[Strg] + [C]** abgebrochen.

Da das koreanische Internetradio eine andere Textkodierung verwendet, sind hier die codeset-Optionen anzugeben. Verwendet das Internetradio dieselbe Textkodierung wie das eigene System (UTF-8; eigene Textkodierung des Systems ermitteln: **locale**), so können die codeset-Optionen entfallen.

**streamripper <http://www.mukulcast.com/#> -a -A -u "Rhythmbox" -d
./InternetRadio --codeset-filesys=UTF-8 --codeset-id3=EUC-KR --
codeset-metadata=EUC-KR --codeset-relay=EUC-KR**

http://www.mukulcast.com/# ... Internetadresse des Streamservers
-a ... rippen als einzelnen Track; Dateiname: sr_program + aktuelle Unix-Timestamp + .mp3
-A ... keine einzelne Tracks speichern; Gesamtmitschnitt
-d ... Zielverzeichnis für die Dateien; hier InternetRadio
-u "Rhythmbox" ... verändert den Namen des UserAgent's für Streamripper (Vorgabe: Streamripper/1.x); einige Radiosender sehen es nicht sehr gern, wenn Ripper-Programme auf ihren Streamserver zugreifen; mit dieser Option kann man versuchen einer Abweisung durch den Streamserver zu entgehen

Mit recode wird die Textkodierung der cue-Datei an das eigene System angepasst.

recode EUC-KR..UTF-8 file_name.cue

Mit dem Terminalprogramm **mp3splt** wird versucht den Datenstream in einzelne Musikstücke aufzulösen, aufzusplitten. Falls dies bei einzelnen Musikstücken nicht gelingt (Pause innerhalb eines Musikstücks, Zeit zwischen Musikstück und Moderation ist zu kurz), muss der Datenstream z.B. mit dem grafischen Programm Audacity manuell bearbeitet werden.

mp3splt -c file_name.cue -o @n_@a_@t file_name.mp3

Hinweis: codeset

Falls in der Terminalausgabe von streamripper seltsame Zeichen auftauchen, so stimmen der Zeichensatz des Streamservers und des eigenen Systems nicht überein. Nachfolgend sind einige Erfahrungswerte aufgelistet.

Westeuropa: --codeset-filesys=UTF-8 --codeset-id3=ISO-8859-15 --codeset-metadata=ISO-8859-15 --codeset-relay=ISO-8859-15

Korea: --codeset-filesys=UTF-8 --codeset-id3=EUC-KR --codeset-metadata=EUC-KR --codeset-relay=EUC-KR

siehe auch: man streamripper, mp3splt, mp3gain, recode

strings

Mit strings lassen sich alle lesbaren Zeichenketten auch von binären Dateien, Partitionen und kompletten Laufwerken anzeigen. In der Grundeinstellung sind für strings lesbare Zeichenketten die eine Länge von mindestens 4 Byte (Zeichen) haben.

Beispiel:

Das SWAP-Laufwerk - hier /dev/sda7 - mit strings nach Webadressen durchsuchen. Die Gerätebezeichnung ihres SWAP-Laufwerkes können sie mit **sudo fdisk -l** ermitteln.

sudo strings /dev/sda7 | grep "http://" | less

siehe auch: man strings, SWAP leeren, ausnullen

su

Benutzer-Login wechseln. Bei der Eingabe von **su** – ohne Benutzernamen – geht der Kommandozeileninterpreter automatisch vom Benutzer **root** aus.

su <Benutzername>

Bei der Eingabe von »su« wird im Hintergrund eine neue Shell unter einer neuen Benutzerkennung (UID) und Gruppenkennung (GID) gestartet. Der Name »su« steht für »substitute user«.

Grundsätzlich lautet das Kommando **su [-] [benutzername]**; dabei gibt es den kleinen aber feinen Unterschied, ob Sie das Minuszeichen verwenden oder nicht. Das Minuszeichen (alternativ können Sie hier den Parameter **-l** oder seine Langform **--login** verwenden) sorgt dafür, dass ein wirkliches Einloggen stattfindet, mit Setzen der richtigen Umgebungsvariablen, Shell und einem Wechsel in das neue Home-Verzeichnis. Ohne das Minuszeichen ändern sich die Umgebungsvariablen nicht, und der neue User besitzt eventuell keine Rechte, sich im aktuellen Arbeitsverzeichnis aufzuhalten.

Standardmäßig erlaubt die Verwendung von **su** dem neuen Benutzer aber nicht, X-Anwendungen zu starten. Einem »fremden« Benutzer muss zunächst erlaubt werden, den X-Server zur Ausgabe zu benutzen. Eine solche Erlaubnis erteilen Sie über die Datei **.Xauthority** im Home-Verzeichnis (siehe auch **man xauth**). Um dem Benutzer **root** zu erlauben, aus einem Xterm, das eigentlich dem User **petronella** gehört, ein X-Programm zu starten, müssen Sie einen passenden »Schlüssel« aus **.Xauthority** extrahieren, der **.Xauthority** des Administrators hinzufügen und danach die **DISPLAY**-Variable neu setzen.

Mit **su** ist es darüber hinaus möglich, ein einziges Kommando unter fremder Identität zu starten. Dazu verwenden Sie die Option **-c** (**--command**):

su -c "less /var/log/messages" ... Anzeige der Logdatei **messages** im

Dateibetrachter für einfache Textdateien `less`

`su -c 'usermod -a -G sudo user_name'` ... einen Benutzer in die Gruppe `sudo` aufnehmen

Die Benutzung des `su`-Kommandos wird protokolliert. Je nach Distribution finden sich diese Log-Einträge in der Datei `/var/log/auth.log` (z. B. Debian) oder auch `/var/log/messages` (z. B. OpenSuse, Ubuntu). Fehlgeschlagene Versuche sind deutlich zu erkennen, so finden Sie als Administrator schnell heraus, dass jemand versucht hat, Root-Rechte zu erlangen.

Als Administrator müssen Sie nach dem `su`-Befehl übrigens kein Passwort eingeben. Sie können jede beliebige Identität annehmen, um beispielsweise schnell etwas aus Sicht eines bestimmten Anwenders zu testen.

* * * * *

sudo

Der Befehl `sudo` kann Programm-Aufrufen vorangestellt werden. Er ermöglicht berechtigten Benutzern, das Programm im Namen und mit den Rechten eines anderen Benutzers auszuführen.

Sudo fragt vor der Ausführung des Programms unter einem anderem Namen nach dem Passwort des aufrufenden Benutzers. Damit wird überprüft, ob dieser den Befehl selbst eingegeben hat. Der Kreis der berechtigten Benutzer ist in der Datei **`/etc/sudoers`** festgelegt. Standardmäßig wird als Ziel-Benutzer `root` angenommen.

`sudo -H -u <BENUTZERNAME> <PROGRAMM>`

Zwei Eigenheiten von `sudo` können zu Problemen führen:

1. Der Befehl **`sudo`** ändert die Umgebungsvariable `$HOME` standardmäßig nicht auf den entsprechenden Pfad zum Ziel-Benutzer. Startet man Programme mit **`sudo`** besteht die Gefahr, dass Konfigurationsdateien mit falschen Rechten im Heimatverzeichnis des ursprünglichen Benutzers erstellt werden. Startet dieser das Programm später unter seinem eigenen Namen - also ohne `sudo` - so kann er die Konfiguration nur noch lesend oder eventuell gar nicht mehr öffnen. Daher sollte **`sudo`** immer mit der Option **`-H`** verwendet werden. Dies gilt auch bei Systembefehlen, die keine Konfigurationsdateien unter `$HOME` ablegen. Die grafischen **`sudo`**-Alternativen (`gksu` bzw. `kdesu`) leiden nicht unter diesem Problem: Dort wird die Umgebungsvariable `$HOME` umgestellt.

2. Grafische Programme lassen sich in manchen Desktopumgebungen mit **sudo -H** nicht aus einem Terminal starten. Dies liegt daran, dass die Grafikumleitung nicht vollständig konfiguriert wird (Xauthority).
Da "sudo -H" also nicht immer funktioniert und auch keine grafische Passwortabfrage erfolgt, sollten grafischen Anwendungen (unter anderem Namen) grundsätzlich über die grafischen Alternativen (gksudo bzw. kdesudo) gestartet werden.

Der Benutzer root

Standardmäßig existiert unter Linux immer ein Konto für den Benutzer "root" mit der User-ID 0. Dies ist ein Systemaccount mit vollem Zugriff auf das gesamte System, und damit auch auf alle Dateien und Einstellungen aller Benutzer.

Bei Ubuntu und bei den Linux-Distributionen die auf Ubuntu basieren, wird dem Benutzer root allerdings kein Passwort zugewiesen. Dadurch kann sich niemand unter dem Namen "root" anmelden.

root im Terminal

Sind für einen Terminal-Befehl Root-Rechte erforderlich, so reicht es dem auszuführenden Befehl das Kommando sudo voranzustellen. Nach der Eingabe wird man nach seinem Passwort gefragt.

Nachdem das Passwort einmal eingegeben wurde, ist dies für die nächsten 15 Minuten bei erneuter Verwendung von sudo in diesem Terminal nicht mehr notwendig (kann angepasst werden). Die erneute Eingabe des Passwortes kann aber auch vorzeitig durch Angabe der Option -k erzwungen werden.

Rootshell

Für größere administrative Aufgaben kann das ständige Voranstellen von sudo auch hinderlich sein. Um länger als root zu arbeiten - sprich mehrere Befehle hintereinander ausführen zu können, ohne immer wieder sudo eingeben zu müssen - kann man in eine Rootshell wechseln. Nach der Eingabe von

sudo -i

werden alle folgenden Befehle mit Rootrechten ausgeführt, bis man diese Rootshell mit

exit

verlässt. Innerhalb dieser Shell muss kein sudo mehr verwendet werden. Die Dauer der Rootshell selbst ist nicht beschränkt. Es liegt in der Verantwortung des Anwenders, diese zu verlassen.

Programme im Kontext anderer Benutzer ausführen

Programme können auch unter dem Namen eines anderen 'regulären' Benutzers gestartet werden.

Terminal - Ausführung durch Benutzer mit Erlaubnis in /etc/sudoers

Sudo fragt immer nach dem Passwort des aufrufenden Benutzers. Ein Benutzerwechsel wird aber nur gestattet wenn dies in /etc/sudoers erlaubt wurde. Dies ist standardmäßig nur für die Gruppe sudo ohne Einschränkungen der Fall.

sudo -H -u BENUTZERNAME PROGRAMM

Terminal - Ausführung mit Passwort des Zielbenutzers

Wenn das Passwort des Zielbenutzers bekannt ist, aber man z.B. nicht zur Gruppe sudo gehört, kann der Befehl su genutzt werden.

su BENUTZERNAME -c PROGRAMM

Für BENUTZERNAME den Login-Namen des anderen Benutzers eingeben. Man Beachte, dass bei diesem Befehl nach dem Passwort des anderen Benutzers gefragt wird.

Hinweis: Rechteeinstellungen, die in der Datei /etc/sudoers vorgenommen wurden, haben bei su keine Wirkung.

Sudo in einem Skript verwenden

Möchte man sudo in einem Skript benutzen und das Passwort über eine grafische Abfrage eingeben lassen, kann dies folgendermaßen geschehen. Der Befehl gksu öffnet die grafische Abfrage für das Passwort und durch die Option -p wird das Passwort in stdout geschrieben. Die Option -S hinter dem sudo Befehl liest das Passwort, welches per Pipe an sudo übergeben wird, von stdin wieder ein. Die Option -- beendet das Einlesen von weiteren Argumenten. Dadurch können dem auszuführenden Befehl eigene Optionsschalter mit gegeben werden.

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

```
gksu -p -m "Bitte Passwort eingeben:" | sudo -S -s -- apt-get update
```

Eine Alternativen zu gksu ist beispielsweise das Programm ssh-askpass.

sudo-Eingaben umleiten

Soll die Ausgabe eines per sudo ausgeführten Befehls umgeleitet werden, so erfolgt dies im allgemeinen nicht mit Root-Rechten. Vor allem, wenn man mit dem Befehl echo etwas in eine Datei schreiben oder anhängen will, funktioniert dies nicht:

```
sudo echo mem > /sys/power/state      # Zugriff verweigert
```

Abhilfe schafft das Kapseln des Befehls in eine eigene Shell (bash):

```
sudo bash -c "echo mem > /sys/power/state"
```

oder einfach die Umsetzung mit tee:

```
echo mem | sudo tee /sys/power/state
```

Optional

sudo bietet auch einige Kommandozeilenparameter. Der wichtigste ist sicherlich **-s**, um eine Root-Shell zu starten. Einen Zugriff auf den X-Server müssen Sie nicht extra konfigurieren, ein einfaches **sudo -s** reicht aus, um als Administrator auch X-Programme starten zu dürfen.

Der Parameter **-L** listet alle Optionen aus der Datei /etc/sudoers auf. Wer sein Ticket verlängern möchte, ohne ein Kommando auszuführen, ruft **sudo -v** auf. Ist der Timeout überschritten, fragt das Kommando nach dem Passwort.

Wollen Sie Ihr Ticket hingegen löschen, wählen Sie **sudo -k**. Mit dem Parameter **-b** ist es möglich, einen Befehl direkt in den Hintergrund zu schieben, er lässt sich mit dem üblichen Shell-Kommando zur Job-Kontrolle (fg) allerdings nicht wieder in den Vordergrund holen.

siehe auch: su, gksu

* * * * *

sudo - /etc/sudoers bearbeiten

Wer aus Sicherheitsgründen das Root-Passwort des Rechners nicht weitergeben möchte, setzt einfach das Programm sudo ein. Der Name ist Programm: »sudo« steht für »substitute user, do« und gibt einzelnen Benutzern oder Gruppen für einen begrenzten Zeitraum oder für immer

Administratorrechte, aber nur für spezielle Aufgaben. Anstelle des Root-Passwortes benutzt der Anwender sein eigenes Kennwort, um einen privilegierten Befehl auszuführen.

Dazu trägt der Administrator in die Datei `/etc/sudoers` ein, welcher Anwender bestimmte Befehle auf dem Computer ausführen darf. Diese Datei sollten Sie (als root) in jedem Fall mit dem Kommando **visudo** editieren, dieses Programm bietet die gewohnten Features des Editors vi zusammen mit ein paar zusätzlichen Funktionen. visudo »sperrt« die Datei `/etc/sudoers`, so dass sie nicht aus Versehen zwei Benutzer gleichzeitig editieren. Außerdem prüft visudo beim Beenden des Editors die Syntax der Datei und meldet eventuelle Fehler:

```
>>> sudoers file: syntax error, line 20 <<<
What now?
```

Drei Möglichkeiten stehen nun zur Wahl: Tippen Sie **e**, um die Datei erneut zu editieren, **x**, um die Änderungen zu verwerfen und den Editor zu verlassen, und **Q**, um die Änderungen trotzdem zu speichern.

Als Standardeintrag findet sich in `/etc/sudoers` **root ALL=(ALL) ALL**, der Administrator darf alles. (Aber das kann er eh, auch ohne sudo.) Wer einem weiteren Benutzer auf dem System uneingeschränkte Root-Rechte geben möchte, kopiert die Zeile und setzt statt root den anderen Benutzernamen ein. Anschließend darf dieser Benutzer Administratorkommandos mit vorangestelltem sudo ausführen, z. B.:

sudo /sbin/shutdown

Ist der User nicht zur Verwendung von sudo berechtigt, meldet **sudo**:

<Benutzername> is not in the sudoers file. This incident will be reported.

Sofern nicht anders in `/etc/sudoers` definiert, erhält der Administrator eine Mail mit genauen Angaben, welcher Benutzer wann versucht hat, sudo aufzurufen. Zur Sicherheit rufen Sie als einfacher User **sudo -l** auf, um eine Liste der erlaubten Kommandos zu erhalten.

Gezielte Freigabe

In `/etc/sudoers` definieren Sie in der Sektion **Host alias specification** Computer, für die bestimmte sudo-Aufrufe gelten sollen. Ein dazu eingerichtetes Host_Alias bildet eine Gruppe von Computern wahlweise

durch Aufführung ihrer Namen oder durch Definition bestimmter IP-Netzbereiche unter einem Namen ab. Sinnvoll ist dieses Feature nur, wenn Sie die sudo-Konfiguration auf mehreren Rechnern verwenden aber zentral verwalten wollen.

Im Bereich `User_Alias` fassen Sie mehrere Benutzer zu einer Gruppe zusammen, wenn alle die gleichen Rechte haben sollen. Zunächst definieren Sie den Alias-Typ (z. B. `User_Alias`), dann den Alias-Namen (darf Großbuchstaben, Unterstrich und Zahlen enthalten), eine Zuweisung in Form eines "="-Zeichen und als letztes die Benutzernamen durch Kommata getrennt. Um die Benutzer `huhn` und `petronella` in einer Gruppe zusammenzufassen, die beispielsweise den Rechner herunterfahren darf, tragen Sie ein:

```
# User alias specification
User_Alias ABSCHALTER=petronella,huhn
```

Als nächstes definieren Sie in der Sektion **Cmnd alias specification** einen Alias für das Kommando `shutdown`. Dabei geben Sie den kompletten Pfad zum Programm an:

```
# Cmnd alias specification
Cmnd_Alias DOWN = /sbin/shutdown
```

Damit `sudo` auch weiß, dass die `ABSCHALTER` dieses Kommando ausführen dürfen, fehlt noch ein Eintrag unter **User privilege specification**:

```
ABSCHALTER ALL = DOWN
```

Ein Benutzer aus der Gruppe **ABSCHALTER** darf den Computer nun mit dem Befehl `sudo /sbin/shutdown` herunterfahren. Wer einfach nur für einen Benutzer einen einzigen Befehl freigeben möchte, kann das einfacher erreichen: Der Eintrag

```
huhn ALL = /usr/sbin/sudo
```

erlaubt dem Benutzer `huhn`, mit `sudo /usr/sbin/sudo` die Datei `/etc/sudoers` zu editieren.

Ausgegrenzt oder uneingeschränkt?

Durch einen einfachen Eintrag in `/etc/sudoers` können Sie einzelnen Benutzern Kommandos auch wieder wegnehmen. Die Syntax dazu lautet beispielsweise:

ABSCHALTER ALL = DOWN

petronella ALL = !DOWN

Wichtig ist, dass die Ausnahme direkt unter der Regel steht, da die Datei von oben nach unten abgearbeitet wird. So kann die Gruppe **ABSCHALTER**, die möglicherweise noch andere Kommandos ausführen darf, weiter bestehen, aber **petronella** kann den Rechner nicht mehr herunterfahren.

Falls dieser Benutzer dennoch versucht, den Befehl auszuführen, erhält er die Meldung **»Sorry, user petronella is not allowed to execute '/usr/sbin/visudo' as root on asteroid.cologne.de.«**

Wer möchte, kann die Passwort-Abfrage für einzelne oder alle Befehle unterdrücken. Dazu setzt man das Flag **NOPASSWD**:

ABSCHALTER ALL=NOPASSWD:DOWN

Die Sicherheit für **sudo** lässt sich aber nicht nur heruntersetzen, sondern auch verschärfen. Sie können Ihre Benutzer dazu **»zwingen«**, bei jedem **sudo**-Aufruf das Passwort einzugeben.

Standardmäßig verwendet **sudo** eine Art **»Ticket«**-System, in dem ein Timeout dafür sorgt, dass eine verlassene root-Shell auf der Konsole nicht Tür und Tor für Unbefugte öffnet.

Die Vorgabe liegt bei den meisten Distributionen bei 15 Minuten, die das Ticket gültig ist. Um den Timeout auf 0 Minuten herunterzusetzen, tragen Sie in **/etc/sudoers** beispielsweise ein:

Defaults timestamp_timeout = 0

* * * * *

sudo – Privilegien gewähren

Das Tool **sudo** vereinfacht die Server-Administration erheblich, da man für Befehle, die root-Rechte benötigen, nicht immer mit **su** zum root-Account wechseln muss.

Die Berechtigungen für **sudo** sind in der Datei **/etc/sudoers** definiert und können dort geändert werden. Manuelle Anpassungen sollten immer mit dem Editor **/usr/bin/visudo** erfolgen, da dieser die korrekte Syntax überprüft.

Debian: sudo ist hier zwar installiert, aber noch nicht für die Verwendung vorkonfiguriert. Um sudo zu nutzen, ist ein Benutzer in die Gruppe sudo aufzunehmen.

usermod -a -G sudo [Benutzernamen] ... Befehl mit root-Rechten aufrufen

Ubuntu (Linux Mint, Xubuntu ...): sudo ist bereits installiert und kann bereits nach der Installation verwendet werden.

Weiteren Benutzern wird die Benutzung von sudo - wie bei Debian – durch die Aufnahme in die Gruppe sudo ermöglicht.

sudo usermod -a -G sudo [Benutzernamen] ... Befehl mit root-Rechten aufrufen

CentOS: Bei CentOS ist es zunächst nötig, das Paket sudo durch den Benutzer root zu installieren.

yum install sudo

Die Standardkonfiguration sieht bereits die vorhandene Gruppe wheel (nicht sudo) für die Verwendung vor, allerdings muss dies erst noch in der Datei /etc/sudoers aktiviert werden. Dazu startet man vom root-Konto mit dem Kommando visudo den Editor (Editor Vi) und schaltet mit der Einfügen-Taste [Einfg] in den Bearbeitungsmodus und entfernt das Kommentarzeichen (#) vor der Zeile

```
# %wheel ALL= (ALL) ALL
```

Nach einem Druck auf die Escape-Taste (Esc) speichert man mit der Eingabe von :wq (w .. write, q .. quit) die Datei ab und schließt den Editor. Mit root-Rechten können Benutzer in die Gruppe wheel aufgenommen werden.

usermod -a -G wheel [Benutzernamen] ... Befehl mit root-Rechten aufrufen

OpenSuse: Bereits bei der Installation konfiguriert OpenSuse den zuerst eingerichteten Benutzer für sudo vor, allerdings auf eine eigenwillige Weise. Wenn es nur einen Admin auf dem Rechner gibt, brauchen an der Konfiguration keine Änderungen vorgenommen werden. Für weitere Benutzer ist es notwendig, die Datei /etc/sudoers anzupassen. Zuerst sind alle Benutzer die sudo verwenden dürfen der Gruppe wheel hinzuzufügen,

auch den bereits für sudo aktivierten Benutzer.

/usr/bin/usermod -a -G wheel [Benutzernamen] ... Befehl mit root-Rechten aufrufen

Zum Bearbeiten der Datei /etc/sudoers ist der Editor visudo mit root-Rechten aufzurufen.

sudo /usr/bin/visudo

Über die Einfügen-Taste (Einfg) wird der Bearbeitungsmodus im Editor visudo aktiviert. Danach ist jeweils vor den nachfolgenden Zeilen ein Kommentarzeichen (#) zu setzen.

Defaults targetpw
ALL ALL= (ALL) ALL

Zudem ist das vorhandene Kommentarzeichen (#) in der nachfolgenden Zeile zu entfernen.

%wheel ALL= (ALL) ALL

Nach einem Druck auf die Escape-Taste (Esc) speichert man mit der Eingabe von :wq (w .. write, q .. quit) die Änderungen in der Datei /etc/sudoers ab und schließt den Editor.

Hinweis: Generell sind neue Gruppenmitgliedschaften immer erst nach einer erneuten Anmeldung des betreffenden Benutzer-Accounts gültig.

* * * * *

SWAP - Auslagerungsspeicher

Der Swap wird entweder als einzelne Datei in einem anderen Dateisystem oder als Partition auf einer Festplatte abgelegt. Im Swap werden temporäre Hauptspeicherdaten ausgelagert, für die im eigentlichen Hauptspeicher kein Platz mehr ist. Dadurch kann ein System, das mit weniger RAM versorgt ist, als es für seine Aufgaben benötigt, mit zusätzlichem virtuellen RAM versorgt werden. Der Nachteil von Swap ist, dass es im Vergleich zum eigentlichen RAM deutlich langsamer ist.

Swap-Partitionen enthalten im eigentliche Sinne kein Dateisystem. Sie müssen aber entsprechend initialisiert werden und die Partitions-ID 82

(Linux Swap) tragen. Diese Partitions-ID wird beim Partitionieren mit den entsprechenden Tools (z.B. fdisk, Gparted) automatisch gesetzt.

Nach einer manuellen Partitionierung wird die Swap-Partition mittels **mkswap** <Partition> initialisiert und mittels **swapon** <Partition> der Auslagerungsspeicher aktiviert.

Zum Einbinden der nachträglich erstellten Swap-Partition beim Booten trägt man sie in die Datei /etc/fstab ein:

```
# /etc/fstab: static file system information.
#
[...]
# <file system> <mount point> <type> <options> <dump> <pass>
[...]
# swap was on /dev/sda7 during installation
UUID=4735d251-88c2-47da-9bbc-d5e99eece14e none          swap  sw
0 0
```

Im Beispieleintrag wird der SWAP (/dev/sda7) mit der automatisch generierten UUID angesprochen.

Hinweis: Einige Linux-Distributionen benutzen statt des Labels none in der 2. Spalte auch swap. UUID's (Universally Unique Identifiers) können sich bei einigen systemnahen Aktionen (z.B. Formatierung) ändern. Hat sich eine UUID geändert (sudo blkid; cat /etc/fstab) so ist die /etc/fstab entsprechend anzupassen.

Ein ungültiger Eintrag in der Datei /etc/fstab führt bei einem Neustart zu erheblichen Problemen. Deshalb ist ein Test der geänderten fstab-Datei mehr als ratsam:

sudo mount -a -f -v ... a .. alles aus der /etc/fstab einhängen, f .. fake, fingierter Mount-Versuch, v .. verbose mode, ausführliche Meldungen

sudo fdisk -l ... alle vorhandenen Partitionen anzeigen; im Beispiel heißt die Swap-Partition /dev/sda7

sudo blkid /dev/sda7 ... UUID der Swap-Partition /dev/sda7 anzeigen; im Beispiel heißt die Swap-Partition /dev/sda7

sudo blkid ... alle UUID's die dem System bekannt sind anzeigen oder

ls -l /dev/disk/by-uuid ... alle UUID's die dem System bekannt sind anzeigen

sudo mkswap /dev/sda7 ... Swap-Partition initialisieren

sudo swapon /dev/sda7 ... Swap-Partition /dev/sda7 aktivieren

sudo swapon -a ... alle in der /etc/fstab eingetragene Swap-Partitionen bzw. Swap-Dateien aktivieren
swapon -s ... zeigt alle Swap-Partitionen bzw. Swap-Dateien an

sudo swapoff /dev/sda7 ... Swap-Partition /dev/sda7 deaktivieren
sudo swapoff -a ... alle in der /etc/fstab eingetragene Swap-Partitionen bzw. Swap-Dateien deaktivieren

Je nach Kernelversion und Architektur unterscheiden sich die maximale Größe und Anzahl der Swap-Partitionen bzw. Swap-Dateien. Lange Zeit galt für die kaum noch unterstützte i386-Architektur eine Obergrenze von 2 GByte pro Swap-Partition und ein Maximum von acht Swap-Bereichen. Als Faustregel für die Größe der Swap-Partition gilt heute, die ein- bis zweifache Größe des Hauptspeichers.

Bei ausreichend Hauptspeicher ($\geq 8 - 16$ GByte) kann auf eine Swap-Partition häufig verzichtet werden. Die Installation eines Systems ohne einen Swap-Bereich, ist aber auch dann vorteilhaft, wenn sie nach der Installation der gewählten Linux-Distribution die manuelle Einrichtung einer Auslagerungsdatei anstatt einer Partition planen.

SWAP-Datei erstellen

Das Linux ganze Partitionen als Swap bevorzugt, hat historische Gründe, obwohl das System dafür auch eine Auslagerungsdatei nutzen kann.

Bis zur Kernelversion 2.6 war die Auslagerungsdatei bei der Verwendung einer Datei noch ein gutes Stück langsamer als bei einer eigenen Partition. Diesen Nachteil kennen aktuelle Kernelversionen nicht mehr.

Einen Nachteil hat die Swap-Datei: bei einem Komplett-Backup wird die Swap-Datei in das Komplett-Backup mit aufgenommen (evtl. Swap-Datei vom Backup ausschließen, siehe auch: tar).

Um eine Swap-Datei anzulegen, wird zunächst eine mit Nullen gefüllte Datei mit einer Größe von 1024 MByte angelegt.

sudo dd if=/dev/zero of=<Dateipfad>/swap bs=1M count=1000

sudo chmod 0600 <Dateipfad>/swap ... Zugriffsrechte setzen, die Datei swap kann nur mit root-Rechten gelesen und bearbeitet werden

sudo mkswap <Dateipfad>/swap ... Datei als Auslagerungsdatei formatieren

sudo swapon -v /<Dateipfad>/swap ... Aktivierung der Swap-Datei
Danach kann das System den Swap-Bereich sofort benutzen.

swapon -s ... Anzeige aller aktivierten Swap-Bereiche, die vom System genutzt werden

Ein weiterer Handgriff ist trotzdem noch nötig, denn bisher würde sich das System nach einem Neustart nicht mehr an die neue Swap-Datei erinnern. Um diese permanent einzurichten, ist die Datei /etc/fstab mit root-Rechten behutsam anzupassen. Fügen sie am Schluss der Datei fstab folgende Zeile ein:

```
Datei: /etc/fstab  
[...]  
# swap  
/<Dateipfad>/swap none swap sw 0 0
```

Das Ende der Datei fstab ist mit einer Leerzeile abzuschließen.

Hinweis: Einige Linux-Distributionen benutzen statt des Labels none in der 2. Spalte auch swap.

Achtung: Ein ungültiger Eintrag in der Datei fstab führt bei einem Neustart zu erheblichen Problemen. Deshalb ist ein Test der geänderten fstab-Datei mehr als ratsam:

sudo mount -a -f -v ... a .. alles aus der /etc/fstab einhängen, f .. fake, fingierter Mount-Versuch, v .. verbose mode, ausführliche Meldungen

Übrigens, es können mehrere Auslagerungsbereiche eingerichtet werden, egal ob als Datei oder als Partition.

Zu beachten ist, dass der Ruhezustand des Linux-Kernels mit einer Swap-Datei standardmäßig nicht funktioniert. Auf einem Notebook ist die Verwendung einer Swap-Datei statt einer Swap-Partition also nicht empfehlenswert.

SWAP leeren

Der Swap ist nach dem Starten des Systems unbenutzt. Möchte man den Swap im laufenden Betrieb leeren, muss der Swap zunächst de- und dann wieder aktiviert werden. Ist der Swap in der /etc/fstab eingetragen, lauten die Befehle hierfür:

sudo swapoff -a ... alle dort eingetragene Swap-Bereiche ausschalten

sudo swapon -a ... alle dort eingetragene Swap-Bereiche einschalten

Zwischendurch kann man über ein weiteres Terminalfenster das Leeren des Swaps beobachten:

free -s 3 | grep Swap

oder einfach

top

Falls es nur um eine spezielle Swap-Partition gehen soll, z.B. sda7, lauten die Befehle entsprechend:

sudo swapoff /dev/sda7

free -s 3 | grep Swap

swapon /dev/sda7

SWAP leeren, ausnullen

Mit dem Ausnullen des Swap-Bereiches werden die dort gespeicherten Informationen gelöscht. Die so gelöschten Informationen sind mit den Bordmitteln nicht mehr auslesbar.

Nach dem Ausnullen ist die Swap-Partition mittels **mkswap <Partition>** wieder zu initialisieren und mit **swapon <Partition>** wieder als Auslagerungsspeicher zu aktivieren.

sudo fdisk -l ... alle vorhandenen Partitionen anzeigen; im Beispiel heißt die Swap-Partition /dev/sda7

swapoff /dev/sda7 ... Swap-Bereich ausschalten

sudo dd if=/dev/zero of=/dev/sda7

sudo sync

sudo mkswap /dev/sda7 ... Swap-Partition initialisieren

Ausgabe:

Setting up swapspace version 1, size = 510972 KiB

kein Label, UUID=bd461eb0-e31f-4125-b8e3-97a0fe0bfab6

sudo -H gedit /etc/fstab ... /etc/fstab anpassen; neue UUID eintragen

Datei: /etc/fstab

[...]

swap was on /dev/sda7 during installation

UUID=bd461eb0-e31f-4125-b8e3-97a0fe0bfab6 none swap sw
0 0

Achtung: Ein ungültiger Eintrag in der Datei fstab führt bei einem Neustart zu erheblichen Problemen. Deshalb ist ein Test der geänderten fstab-Datei mehr als ratsam:

sudo mount -a -f -v ... a .. alles aus der /etc/fstab einhängen, f .. fake, fingierter Mount-Versuch, v .. verbose mode, ausführliche Meldungen

sudo swapon /dev/sda7 ... Swap-Bereich wieder einschalten

siehe auch: strings, sudo, blkid

* * * * *

systemd – Das Init-System

Bei Linux-Distributionen übergibt der Kernel während des Bootens dem Init-System die Verantwortung zur Einrichtung des Systems. Der Init-Prozess ist der erste Prozess (Prozess-ID 1), den der Kernel erzeugt. Init-Systeme, wie systemd, dienen dementsprechend dem Starten, Überwachen und Beenden weiterer Prozesse (PID > 1).

Das Init-System systemd erschien im April 2010. Einige Distributionen, wie Fedora, OpenSuse und Arch Linux, verwenden schon das neue Init-System systemd. Bei einigen weitere Linux-Distributionen (Ubuntu und Linux-Distributionen die aus Ubuntu basieren) wird der optionale Einsatz oder der Umstieg diskutiert (Stand: 2014).

Eine der Besonderheiten von Systemd ist der parallele Start von Hintergrunddiensten (weitgehend werden alle Prozesse gleichzeitig gestartet), ohne dass Abhängigkeiten zwischen diesen explizit festgelegt werden müssen; das nutzt Hardware-Ressourcen effizienter und lässt das System schneller starten. Außerdem werden von systemd beim Systemstart deutlich weniger Dienste aufgerufen. Nur gelegentlich benötigte Dienste werden von systemd ereignisbasiert erst bei Bedarf gestartet.

Systemd erledigt zudem einige Aufgaben, um die sich bislang meist distributionsspezifische Skripte kümmern; ganz nebenbei beseitigt es damit einige Unterschiede bei der Bedienung und Konfiguration von Distributionen.

Das Init-System systemd ist abwärtskompatibel zu SysVinit-Skripten. Allerdings werden bewusst Features benutzt, die nur auf Linux zur Verfügung stehen, nicht aber auf anderen Unix-Betriebssystemen. Es kann daher nur auf Systemen mit Linux-Kernel laufen.

Aufgaben

Der Init-Prozess ist der erste Prozess (Prozess-ID 1), den der Kernel erzeugt. Alle weiteren Prozesse sind Kinder des Init-Prozesses, der daher die Verantwortung für die komplette Einrichtung des Userlands trägt. Dazu gehört nicht nur das Einhängen von Dateisystemen und die Netzwerkeinrichtung, sondern auch das Starten von Hintergrund-Diensten und Programmen – darunter auch jene, über die sich Benutzer am System anmelden.

Nach dem Abschluss der Systemeinrichtung läuft der Init-Prozess weitgehend untätig im Hintergrund weiter. Er kommuniziert mit dem Kernel und wird beispielsweise informiert, wenn der Benutzer [Strg] + [Alt] + [Entf] drückt. Genau wie beim Aufruf von Befehlen wie **shutdown -r now** oder **reboot** erledigt der Prozess mit der PID1 dann alles Nötige, um das System sauber zum Stillstand zu bringen.

Mit diesen Aufgaben wurde in den 80er Jahren in Unix SystemV das einfache, aber flexible "System V Init System" betraut. In den 90er Jahren entstand eine SysVinit genannte Neuimplementierung dieses Init-Systems. Sie arbeitet mit einer ganz ähnlichen Logik und kommt bis heute bei vielen Linux-Distributionen zum Einsatz. SysVinit erledigt die Aufgaben des Systemstarts im Wesentlichen mit Shell-Skripten, die einfach der Reihe nach abgearbeitet werden.

Mit der Verbreitung von Linux in Mobilgeräten, Desktop-PCs, Fernsehern und zahlreichen anderen Gebieten wandelten sich allerdings die Anforderungen an den Init-Prozess: Der Systemstart sollte flexibler werden und dank Parallelisierung deutlich schneller ablaufen.

Systemd bedient sich einiger Ideen aus früheren Unit-Systemen und kombiniert diese mit einer einheitlichen Konfigurations- und Administrationsschnittstelle. Systemd arbeitet als Hintergrunddienst (Daemon) und steuert wichtige Aspekte der Systemkonfiguration von der

Initialisierung der Hardware bis zu den gestarteten Server-Prozessen.

Vor- und Nachteile von systemd

Das Init-System `systemd` ersetzt alle Shell-Boot-Skripte durch deklarative Konfigurationsdateien, in denen definiert wird, wie die jeweiligen Dienste gestartet werden. Diese Dateien sind in der Regel deutlich einfacher zu schreiben als `init`-Skripte und vermeiden die hohe Anzahl von Shell-Skripten.

Vorteile

- Die Arbeitsweise wird von vielen kleinen Skripten (`SysVinit`) nach `systemd`, also **einem Skript**, verlagert. Der Administrator beschäftigt sich also nicht mehr mit dem Schreiben von `Init`-Skripten, sondern erstellt lediglich Anweisungen (Unit Files) wie ein Programm zu starten ist und welche Abhängigkeiten dieses hat.
- Das `Init`system `systemd` kann genau feststellen, ob ein bestimmter Dienst läuft und kann diesen darüber hinaus auch zuverlässig beenden.
- `Runlevel`, bei `systemd` eigentlich `Targets`, können unabhängig von der aktuellen Position und unabhängig davon, ob andere Dienste zwischendurch gestartet oder beendet wurden, zielsicher erreicht werden. Gehört zum Beispiel zu einem `Target` »`serverbetrieb.target`« kein `Apache`, wird dieser beim Wechsel vom `Target` »`privatstuf.target`« zuverlässig beendet, und dafür gegebenenfalls ein anderer Dienst aktiviert.
- `Socket Activation`: `systemd` ist in der Lage Dienste erst zu starten, wenn dies tatsächlich erforderlich ist. Dies ist vor allem hilfreich für Maschinen aus der Softwareentwicklung, welche wohl nicht immer alle Dienste benötigen, diese aber gerne bei Bedarf automatisch gestartet hätten.
- Einheitliche Konfigurationsdateien: `systemd` definiert genau wo welche Informationen konfiguriert werden müssen, das heißt, dass sich jede Distribution mit `systemd` zu weiten Teilen gleich verhält – was zumindest Dienste angeht, Paketverwaltungen und Co. bleiben hiervon unberührt.
- Dienste, welche nicht selbstständig in einen anderen Benutzerkontext wechseln, können dies in Zukunft ohne **`sudo`** oder **`su`** erledigen, da `systemd` hierzu Funktionen bereitstellt.
- Das `Init`system `systemd` kann sich zur Laufzeit durch eine neuere Version ersetzen, ein Neustart für ein Sicherheitsupdate oder neue

Features am Init-System sind also nicht nötig.

Nachteile

- systemd läuft nur auf einem Kernel, welcher bestimmte Features wie zum Beispiel Control Groups bereitstellt. Dies ist aktuell ausschließlich bei Linux der Fall, eine Portierung auf andere Unix-Derivate ist aktuell nicht geplant und daher unwahrscheinlich.
- Bruch mit Bestehendem: systemd stellt zu weiten Teilen einen kompletten Neuanfang da. Dies bedeutet aber auch, dass Bekanntes so nicht mehr funktioniert und ein Umdenken beim Anwender erforderlich ist.
- Das Initsystem systemd verlagert die Komplexität von vielen kleinen Skripten in eine zentrale Software.

journald

journald ist ein Teil von systemd und ist ein Ersatz für den bestehenden Syslog- und logrotate-Dienst. Nachteil ist, dass die Logdateien in einem bisher nicht dokumentiertem binärem Format, das nicht von Menschen gelesen werden kann, abgespeichert werden und somit ein Zugriff mittels den Tools wie less, more oder grep nicht mehr möglich ist.

journald definiert darüber hinaus aber auch die Möglichkeit, Metadaten in Logdateien zu schreiben oder Logdateien zu signieren (FSS). Das sorgt in manchen Anwendungsfällen dafür, dass Logdateien nicht manipuliert, aber dennoch auffällig gelöscht werden können.

Ebenfalls wurden bei journald einige Kritikpunkte der bisherigen syslog/logrotate-Lösung behoben. So war es möglich, dass diese einem das Dateisystem voll schreiben – journald passt hier automatisch auf – oder das Informationen über viele Dateien verstreut liegen.

Zugriff auf diese Logfiles erfolgt über das Tool **journalctl**, welches auch einem normalen Benutzer das vollständige Systemlog anzeigt, sofern man Mitglieder der Gruppe adm ist.

Dienste starten, anhalten, aktivieren etc.

Zum Aktivieren eines Dienstes sind beim Init-System systemd Links im Verzeichnis `/etc/systemd/system` erforderlich. Zur Verwaltung und Anlegen dieser Links ist beim Init-System systemd **systemctl** (z.B. `systemctl enable dienst.service`) zuständig.

systemctl -t service ... Auflistung aller Dienste, keine root-Rechte erforderlich; in der Liste mit den Pfeiltasten scrollen; Liste schließen über die Taste [Q]

sudo systemctl start [Dienstname].service ... Dienst starten

sudo systemctl stop [Dienstname].service ... Dienst anhalten

sudo systemctl disable [Dienstname].service ... Dienst dauerhaft deaktivieren

systemctl restart dienstname.service ... Dienst neu starten

sudo systemctl enable [Dienstname].service ... einen Dienst für den automatischen Start beim Booten aktivieren

systemctl isolate runlevel.target ... Runlevel ändern

Dienst ändern: Überschreiben des Distributorskripts in /etc

Unit Files

Ein zentrales Konzept von systemd sind die Unit-Files. Diese ersetzen die Init-Skripte von anderen Systemen und sind wesentlich einfacher aufgebaut. Es gibt verschiedene Arten von Unit Files (Unit Typen), nachfolgend ein paar Beispiele:

.service ... Typ für normale Dienste

.target ... Zieltyp, dient zum Beispiel als Ersatz für Runlevels (graphical.target), aber auch für Zwischenschritte (network.target, local-fs.target, ...)

.mount ... Typ für Mountpoints, meist automatisch durch systemd-fstab-generator erzeugt

.socket ... Typ für Socket Activation von Diensten

Beispieleintrag für systemd:

[Unit]

Description=Periodic Command Scheduler

[Service]

ExecStart=/usr/sbin/crond -n

ExecReload=/bin/kill -HUP \$MAINPID

Restart=always

[Install]

WantedBy=multi-user.target

Im systemd-Unit wurde hier definiert, dass cron.service zum Target multi-user.target gehört, das entspricht einem Mehrbenutzer-Runlevel im normalen SysVinit ohne grafische Oberfläche. Wer jetzt glaubt im grafischen Modus (mit GDM, KDM, ...) keinen cron haben zu können, irrt: Targets können von anderen Targets abhängen und so definiert beispielsweise graphical.target – welches bei den meisten Distributionen der Standard ist – dass zuerst ein multi-user.target gestartet wird.

Die WantedBy Definition ist übrigens nur ein Vorschlag für systemctl. Es ist jederzeit möglich durch eigene Symlinks unterhalb von /etc/systemd/system/ das Verhalten und die Reihenfolge zu modifizieren. Unabhängig davon werden jedoch andere Abhängigkeiten der einzelnen Units beachtet. Dies führt zum Beispiel immer dazu, dass Avahi gestartet wird, wenn ein Dienst diesen benötigt, unabhängig davon, in welchem Target der Dienst gestartet wird.

Das Init-System systemd verlagert die Aufgaben von vielen kleinen Skripten in einen einzigen Dienst.

siehe auch: Upstart, init (SysVinit), Dienste starten, anhalten und deaktivieren

T

Tastatur

Sonderzeichen über die Tastatur eingeben:

« ... [Alt Gr] + [y]

» ... [Alt Gr] + [x]

ø ... [Alt Gr] + [o]

Bei den nachfolgenden Tastenkombinationen, ist erst mit [Alt Gr] + [] die entsprechende »tote Taste« zu aktivieren, danach ist die Tastenkombination wieder loszulassen - anschließend kann der angegebene Buchstabe auf der Tastatur gedrückt werden.

ç ... [Alt Gr] + ['] anschließend [c]

ã ... [Alt Gr] + [~] anschließend [a]

ñ ... [Alt Gr] + [~] anschließend [n]

õ ... [Alt Gr] + [~] anschließend [o]

ë ... [Alt Gr] + [ü] anschließend [e]

ï ... [Alt Gr] + [ü] anschließend [i]

Für die Großschreibung ist zusätzlich die [Shift]-Taste zu drücken, z.B. Ø
... [Alt Gr] + [Shift] + [o].

Dasselbe erreicht man auch mit OpenOffice über das Menü «Einfügen»
«Sonderzeichen». Die Tastaturvariante ist aber nach einer
Eingewöhnungsphase mit Abstand die schnellere Variante.

Falls diese Tastenkombination nicht die beabsichtigte Wirkung zeigen, so
sollte über das Kontrollzentrum die «toten Tasten» aktiviert werden. Die
«toten Tasten» zeigen bei einem einmaligen Druck auf die Tastatur keine
sichtbare Wirkung, die Wirkung wird erst bei einer Tastenkombination
sichtbar.

Hinweis: Unicode in ein Terminal eingeben

Durch die Tastenkombination [Strg] + [Shift] + [u] gefolgt von dem
hexadezimalen Code, kann man beliebige Sonderzeichen am Bildschirm
ausgeben ([Strg] und [Shift] gedrückt halten während der Code eingegeben
wird).

Terminal

siehe auch: Shell

tac

Gibt wie »cat« eine Datei aus, aber rückwärts, die letzte Zeile zuerst.

tac datei.txt

siehe auch: man tac, cat

tail

Die letzten 10 Zeilen jeder DATEI auf Standardausgabe ausgeben.

Beispiele:

tail /var/log/messages ... Ausgabe der letzten 10 Zeilen der angegebenen Datei

tail -n 50 /var/log/messages ... Ausgabe der letzten 50 Zeilen der angegebenen Datei

tail -f /var/log/messages ... Neueinträge in dieser Datei werden sofort auf den Bildschirm ausgegeben; diese Option ist besonders interessant um Log-Dateien in Echtzeit zu überwachen; mit [Strg] + [C] wird das laufende Programm beendet

tail -c 20 /var/log/messages ... es werden die letzten 20 Zeichen ausgegeben (c steht für char=Zeichen)

Hinweis: Die Datei /var/log/messages wird nicht von allen Linux-Distributionen verwendet. Einige Linux-Distributionen verteilen die Log-Einträge auf mehrere Dateien (z.B.: /var/log/kern.log, /var/log/ufw.log, etc.).

weiteres Beispiel:

head -n 27 dateiname | tail -n 1

head ... übergibt an **tail** die ersten 27 Zeilen von dateiname und tail gibt davon die letzte Zeile, also die 27. Zeile, aus

siehe auch: man tail

tar

tar fasst viele Dateien in einem einzigen Archiv zusammen und kann alle oder einzelne Dateien aus dem Archiv wiederherstellen. Der Name tar kommt ursprünglich von Tape Archive.

tar -xvf <MeinArchiv.tar> ... entpackt das unkomprimierte Archiv; ohne Angabe eines Zielverzeichnisses wird der Inhalt des Archivs immer im aktuellen Arbeitsverzeichnis entpackt

tar -xvzf <MeinArchiv.tar.gz> ... entpackt das mit gzip komprimierte Archiv; ohne Angabe eines Zielverzeichnisses wird der Inhalt des Archivs immer im aktuellen Arbeitsverzeichnis entpackt

tar -xzf <MeinArchiv.tgz> -C /tmp/ ... entpackt das mit gzip komprimierte Archiv ins Zielverzeichnis /tmp/ (-C /tmp/)

for f in *.tar.gz; do tar -xvzf \$f; done ... entpackt alle TAR.GZ-Archive des aktuellen Verzeichnisses mittels einer for-Schleife

tar -cvzf <MeinArchiv.tar.gz> <MeinVerzeichnis> ... Erstellt ein mit gzip komprimiertes Archiv (c .. create, z .. gzip), d.h. es wird mit tar erst das benannte Verzeichnis (samt seiner Unterverzeichnisse) in ein Archiv (f ... file) gepackt und anschließend mit gzip nochmals komprimiert.

tar -cf archiv.tar foo bar ... Erstellt das Archiv archiv.tar mit den Dateien foo und bar.

tar -xvjf dateiname.tar.bz2 ... entpackt ein bzip2-komprimiertes Archiv namens dateiname.tar.bz2

for f in *.tar.bz2; do tar -xvjf \$f; done ... entpackt alle TAR.BZ2-Archive des aktuellen Verzeichnisses mittels einer for-Schleife

Optionen:

-x ... Dateien sollen aus einem Archiv entpackt werden

-z ... das Archiv ist gezippt oder wird mit gzip komprimiert

-p ... Informationen über Dateizugriffsrechte mit extrahieren (Voreinstellung für Root)

-C ... dieser Parameter bestimmt das Zielverzeichnis

-c ... legt ein neues tar-Archiv an (c ... create)

-f ... das Archiv liegt in Form einer Datei vor (f .. file)

-u ... fügt Dateien hinzu, aber nur wenn sie neuer sind als die im Archiv

-v ... gibt die Namen aller bearbeiteter Dateien aus (v .. verbose)

-t ... gibt den Inhalt eines Archivs aus (extrahiert) - z.B. **tar -tvzf test.tar.gz**

-r ... fügt Dateien einem bestimmten Archiv hinzu

Beispiel: Komprimierung eines ganzen Verzeichnisses

1. Verzeichnis heißt z.B. test

2. mit **cd <Pfad>** ins Verzeichnis wechseln, dass das Verzeichnis test enthält

3. **tar -cvf test.tar test**

4. überprüfen des Inhalts der neuen Datei mit **tar -tf test.tar**

5. **gzip test.tar** ... das neue TAR-Archiv zusätzlich mit gzip komprimieren - die neue Datei heißt jetzt **test.tar.gz**

6. die Datei test.tar.gz in ein anderes Verzeichnis kopieren und mit cd in dieses wechseln

7. **gunzip test.tar.gz** ... das TAR-Archiv wird aus der gezippten Datei entpackt

8. **tar -xvf test.tar** ... wird das TAR-Archiv entpackt, d.h. das Archiv wird in die einzelnen Dateien aufgelöst

Der Schritt 3 und 5 kann zusammengefasst werden:

tar -cvzf test.tar test

Der Schritt 7 und 8 kann ebenfalls zusammengefasst werden:

tar -xvzf test.tar.gz

siehe auch: tar --help, gzip, unrar, unzip

tcpdump

Wenn etwas in Ihrem Linux-Netzwerk nicht richtig funktioniert, lohnt es sich immer, einen Blick auf tcpdump zu werfen.

tcpdump [Optionen] [Filterausdrücke]

Achtung: tcpdump kann nur von **root** oder mit sudo-Rechten ausgeführt werden.

tcpdump ist das Standard-Netzwerkanalyse-Programm, das mit nahezu jeder Linux Distribution mitgeliefert wird. tcpdump arbeitet wie jeder herkömmliche Sniffer, indem es die Netzwerkschnittstelle in den »promiscuous mode« bringt, um den kompletten Netzwerkverkehr zu beobachten. tcpdump besitzt jedoch einige sehr ausgereifte Analysefunktion, die den entsprechenden Verkehr auch interpretieren und zahlreiche Informationen daraus extrahieren kann.

Dieses Tool schreibt **live** alle Protokoll-Header und gibt sie auf dem Bildschirm aus. Sie überwachen damit also den kompletten Netzwerkverkehr zwischen dem Linux-Rechner und dem Netzwerk. Falls Sie an Stelle eines Switches einen Hub einsetzen, können Sie sogar den kompletten Datenverkehr im Netz überwachen. Allerdings erschlägt Sie dann die schiere Menge an Informationen.

Aber zurück zum Anfang: Falls Sie das Gefühl haben, irgendetwas stimmt nicht zwischen den Linux-Rechnern und dem Netz, dann starten Sie **tcpdump** (ohne Optionen). Sofort sehen Sie den Datenverkehr.

Darüber hinaus lässt sich hier aber auch feststellen, ob der TCP-Verkehr klappt. Versuchen Sie bei laufendem TCP-Dump einmal, per http auf den

Rechner zuzugreifen. Sofort erscheint eine Reihe von http-Header-Meldungen.

Mit tcpdump können Sie also gezielt überwachen, welche Anwendungen wie kommunizieren. Bei Fehlern erhalten Sie hier auch wertvolle Meldungen, die Ihnen weiter helfen. Ganz abgesehen davon sehen Sie einmal, was sich überhaupt im Netzwerk tut und wer mit wem schwätzt.

Natürlich können Sie auch einschränken, was tcpdump überwacht. Wenn Sie zum Beispiel nur den Datenverkehr über http betrachten wollen, geben Sie ein:

tcpdump port 80

Denn über diesen Port kommen per Standard alle http-Verbindungen zu Stande. Alternativ können Sie auch nur einen bestimmten Rechner überwachen:

tcpdump host 192.168.0.1

Tip: Verwenden Sie tcpdump ausschließlich von der Konsole Ihres Linux-Rechners aus. Falls Sie ihn per Telnet oder ssh von einem anderen Rechner aus starten, protokolliert tcpdump ständig den Datenverkehr zwischen dem Terminal und dem Linux-Rechner – und erzeugt damit nur neuen Datenverkehr.

Optionen:

tcpdump -w <Dateiname> ... tcpdump schreibt den Dump in eine Textdatei; mit der Option -w hat die Datei auch gleich das richtige Format um es mit Ethereal auswerten zu können; Ethereal ist ein Sniffer mit grafischer Oberfläche

tcpdump -r <Dateiname> ... tcpdump liest eine Textdatei ein die über die Option -w erstellt wurde

tcpdump -v ... die Zusammenfassung des Paket-Headers wird etwas ausführlicher als normal ausgegeben

tcpdump -vv ... eine besonders ausführliche Ausgabe am Bildschirm

tcpdump -X ... nach dem Header wird ein Hexdump des Paketinhalts ausgegeben

Filterausdrücke:

Da auf einem Netzwerk u. U. sehr viele verschiedene Pakete gesendet werden, ist es notwendig, die interessanten Pakete mittels eines Filters zu selektieren. Hierzu bietet tcpdump eine Filtersprache an. Die wichtigsten

Schlüsselwörter:

udp port n ... UDP-Pakete, die den Start- oder Zielpport n haben

tcp port n ... TCP-Pakete, die den Start- oder Zielpport n haben

a and b ... Pakete, auf die sowohl die Bedingung a als auch b zutrifft

a or b ... Pakete, auf die entweder die Bedingung a oder b (oder beide) zutrifft

Interpretation der tcpdump-Ausgabe

Falls die Option -X angegeben ist, gliedert sich die Ausgabe eines jeden Pakets in zwei Teile: oben werden die wichtigsten Daten der Header (IP und TCP bzw. UDP) zusammengefasst (Nummern 1-12), unten wird das Paket noch einmal vollständig als Hexdump ausgegeben (Nummern 13-15). Ohne die Option -X entfällt der untere Teil.

Die Ausgabe eines typischen Pakets in der Abbildung wird im folgenden einzeln erläutert.

```
1 12:32:23.034868 141.35.14.194.23 > 141.35.14.22.63795: P 133:145(12) ack 44
2 win 5840 (DF) [tos 0x10] (ttl 64, id 31917, len 52)
3
4
5
6
7 0x0000 4510 0034 7cad 4000 4006 86e8 8d23 0ec2 E..4l.@.d...#..
8 0x0010 8d23 0e16 0017 f933 fbb2 e1f7 8664 8b31 .#.....3....d.1
9 0x0020 5018 16d0 8474 0000 4861 6c6c 6f20 5765 P...t..Hallo.We
10 0x0030 6c74 0d0a
11
```

Die einzelnen Felder der Ausgabe haben folgende Bedeutung (1 ... links oben, 15 ... rechts unten):

1. Zeitpunkt, zu dem das Paket auf dem Netzwerk abgehört worden ist.
2. IP-Adresse und Portnummer der Quelle.
3. IP-Adresse und Portnummer des Ziels.
4. Gesetzte TCP-Flags. Die Buchstaben haben folgende Bedeutung: SYN, FIN, PSH, RST. Das ACK-Flag wird gesondert im Feld 6 aufgeführt.
5. Bereich von Sequence-Nummern, der in diesem Paket abgedeckt wird. Das im Beispiel ausgegebene TCP-Segment enthält die Bytes 133-145 des TCP-Datenstroms.
6. Im Falle des gesetzten ACK-Flags wird hier die Acknowledgement-Nummer ausgegeben. Im Beispiel wurden alle Bytes bis 44 des (gegenläufigen) TCP-Datenstroms korrekt empfangen.
7. Aktuelle Fenstergröße für den Empfangspuffer.
8. Gesetzte Flags im IP-Header.
9. Wert des Type of Service-Feldes im IP-Header.
10. Wert des Time To Live-Feldes im IP-Header.
11. Identification Number im IP-Header.
12. Wert des Total Length-Feldes im IP-Header, d. h. die Gesamtgröße des Datagramms inklusive Header.

13. Hexdump: Position des ersten Zeichens dieser Zeile (Adresse), hexadezimal.
14. Jeweils 16 Byte Paketinhalt in Hexadezimaldarstellung (inklusive Header). Falls es nicht mehr präsent sein sollte bei Ihnen, wiederholen Sie zur Vorbereitung bitte die Umrechnung von der Hexadezimaldarstellung in Dezimal- und Binärdarstellung.
15. Dieselben Daten noch einmal in ASCII-Darstellung. Falls es sich um ein druckbares Zeichen handelt, wird es ausgegeben, ansonsten ein Punkt.

Hinweis zu Punkt 5: Das Betriebssystem vergibt als erste Sequence-Nummer (initial sequence number, ISN) einen zufälligen, i. d. R. großen Wert. Zwecks Übersichtlichkeit werden von tcpdump alle Sequence-Nummern, die danach auftreten, relativ zur ISN umgerechnet.

siehe auch: man tcpdump

tee

Leitet die Eingabe sowohl in eine (als Argument angegebene) Datei als auch an die Standardausgabe weiter und erinnert damit an ein T-Stück im Wasserrohr.

Es eignet sich hervorragend dazu, die Ausgabe eines Prozesses parallel in eine Datei zu schreiben und live an der Konsole mitzuverfolgen oder an ein anderes Programm weiterzuleiten: »Kommando | tee Datei | Kommando«.

Beispiele:

cat datei.txt | tee datei1.txt ... Inhalt von datei.txt in datei1.txt schreiben und gleichzeitig den Dateiinhalte am Bildschirm anzeigen

sudo echo 'manual' | sudo tee -a /etc/init/Upstart-Dienstname.override

... das Wort manual in die Datei Upstart-Dienstname.override; das Wort manual wird angehängt (Option: -a), d.h. ein existierender Inhalt in der Datei wird nicht überschrieben

siehe auch: man tee

test

test überprüft Dateitypen und vergleicht Werte.

test -f <Dateiname> ... Überprüfung, ob die Datei vorhanden ist. Der Befehl test wird in Shellskripts verwendet.

siehe auch: if, Anhang: Einführung in die Shellprogrammierung

Textkodierungen

siehe auch: Konvertierung von Textkodierungen, iconv, recode, fromdos, todos

todos

Konvertiert Textdateien von UNIX ins DOS-Format. todos ist Bestandteil des Paketes tofromdos (fromdos, todos).

Die verschiedenen Betriebssysteme (Linux, Unix, Windows) benutzen für das Zeilenende jeweils ein anderes nicht sichtbares Steuerzeichen (Linux \n; MAC \r; Windows \r\n); mit den Terminalprogrammen fromdos (Windows/DOS → Linux) und todos (Linux → Windows/DOS) können die Zeilenumbrüche von Textdateien konvertiert werden.

todos a.txt b.txt ... a.txt und b.txt werden ins DOS/WINDOWS-Format konvertiert; Originaldatei wird dabei überschrieben

todos -b a.txt ... von der Originaldatei a.txt wird ein Backup (a.bak) erstellt

siehe auch: man todos, fromdos, man tofromdos, recode, iconv, Konvertierung von Textkodierungen

top

Wer von den laufenden Programmen eine etwas schönere Darstellung hätte, ruft top auf. Top ist ein Systemmonitor, der im oberen Bildschirmbereich Informationen zum Ressourcenverbrauch des Systems bietet und darunter eine Tabelle mit laufenden Programmen anzeigt.

? - Hilfe zu top

[P] - sortiert die Programme nach oben, die die meiste Prozessorleistung benötigen

[m] - entlarvt die größten Speicherfresser

[u] <Benutzername> - zeigt die Programme eines bestimmten Benutzers

[c] - zeigt die vollständige Befehlszeile an, mit dem das Programm aufgerufen wurde; durch erneutes drücken von [c] kehren Sie wieder zur Standardansicht zurück

[q] - beendet top

siehe auch: man top, htop

touch

Falls die Datei nicht existiert, so wird sie mit touch angelegt und mit dem aktuellen Zeitstempel versehen. Existiert die Datei, so wird als Zeitstempel

die aktuelle Zeit verwendet.

`touch [OPTIONEN] Dateinamen`

Optionen:

touch -a <Dateiname> ... aktualisiert nur die Zugriffszeit (a ... access time)

touch -a -t 200511011205.12 test1.txt ... manuelle Definition, ändert nur die Zugriffszeit [JJJJMMTTSSmm.ss]

touch -m <Dateiname> ... aktualisiert nur die Modifikationszeit (m ... modification time)

touch -am -t 200511011211.12 test1.txt ... manuelle Definition, ändert die Zugriffs- und Modifikationszeit [JJJJMMTTSSmm.ss]

touch -r <Bezugsdatei> <Dateiname> ... verwendet den Zeitstempel der Bezugsdatei bzw.

touch --reference=<Bezugsdatei> <Dateiname> ... benutzt die Zeiten der Bezugsdatei (Zugriffs- und Modifikationszeit, nicht die Change-Zeit) anstatt der momentanen Zeit

touch -t Jahr,Monat,Tag,Stunde,Minute [.Sekunde]<Dateiname> ... manuelle Definition der Modifikationszeit [JJJJMMTTSSmm.ss] der Datei - z.B. `touch -t 200511011202.14 test1.txt`

touch --date="2038-01-19 03:14:07 UTC" test1.txt ... ändert bzw. aktualisiert die Zugriffs- und Modifikationszeit [JJJJ-MM-TT SS:mm:ss], UTC-Zeit

touch --date="2038-01-19 03:14:07" test1.txt ... ändert bzw. aktualisiert die Zugriffs- und Modifikationszeit [JJJJ-MM-TT SS:mm:ss], lokale Rechnerzeit

siehe auch: man touch, ls, stat

tree

Anzeige von Verzeichnis- und Dateibäume. tree ist besonders nützlich für die Erstellung von schnellen Inhaltsverzeichnissen.

tree ... Bildschirmausgabe aller Verzeichnisse und Dateien des aktuellen Verzeichnis und seiner Unterverzeichnisse

tree > output.txt ... Umleitung der Bildschirmausgabe in eine Textdatei

tree | tee output.txt ... Bildschirmausgabe aller Verzeichnisse und Dateien und gleichzeitige Umleitung der Bildschirmausgabe in eine Textdatei

tree -d ... Bildschirmausgabe nur der Verzeichnisse

tree -d | tr _ ' ' > output.txt ... Bildschirmausgabe der Verzeichnisse; Unterstriche () in den Verzeichnisnamen werden durch Leerzeichen ersetzt; Umleitung der Bildschirmausgabe in eine Textdatei

siehe auch: man tree

tr

Zeichen von der Standardeingabe wandeln, verdichten und/oder löschen;
und auf der Standardausgabe (Bildschirm) schreiben.

tr [OPTION]... MENGE1 [MENGE2]

-c, --complement ... erstes Komplement MENGE1

-d, --delete ... Zeichen der MENGE1 löschen, nicht wandeln

-s, --squeeze-repeats ... jede Eingabefolge eines wiederholten Zeichens,
das in MENGE1 enthalten ist, durch ein einzelnes Vorkommens dieses
Zeichens ersetzen

-t, --truncate-set1 ... zuerst MENGE1 auf die Länge von MENGE2
abschneiden

MENGE[n] werden angegeben als Zeichenketten. Die meisten Zeichen
stehen für sich selbst.

Interpretierte Folgen sind:

\NNN ... Zeichen mit Oktalwert NNN (1 bis 3 oktale Ziffern)

**** ... Backslash (\)

\a ... hörbarer Ton (Piep)

\b ... Zeichen zurück

\f ... Seitenvorschub

\n ... Zeilenvorschub

\r ... Wagenrücklauf

\t ... horizontaler Tabulator

\v ... vertikaler Tabulator

ZEICH1-ZEICH2 ... alle Zeichen von ZEICH1 bis ZEICH2 aufsteigend

[ZEICH*] ... in MENGE2, Kopien von ZEICH bis zur Länge von
MENGE1

[ZEICH*ANZ] ... ANZ Kopien von ZEICHEN, ANZ ist oktal, wenn es mit
0 beginnt

[:alnum:] ... alle Buchstaben und Ziffern

[:alpha:] ... alle Buchstaben, auch ä,ö,ü,Ä,Ö,Ü etc.

[:blank:] ... alle horizontalen Leerzeichen/Tabulatoren

[:cntrl:] ... alle Kontrollzeichen

[:digit:] ... alle Ziffern

[:graph:] ... alle druckbaren Zeichen, ohne Leerzeichen

[:lower:] ... alle Kleinbuchstaben

[:print:] ... alle druckbaren Zeichen, einschl. Leerzeichen
[:punct:] ... alle Satzzeichen
[:space:] ... alle horizontalen oder vertikalen Leerzeichen/Tabulatoren
[:upper:] ... alle Großbuchstaben
[:xdigit:] ... alle hexadezimalen Ziffern
[=ZEICHEN=] ... alle Zeichen äquivalent zu ZEICHEN

Wandlung wird durchgeführt, wenn nicht -d spezifiziert ist und sowohl MENGE1 als auch MENGE2 angegeben sind. -t darf nur bei Wandlung benutzt werden.

MENGE2 wird, wenn nötig, durch Wiederholung des letzten Zeichens auf die Länge von MENGE1 vergrößert. Zusätzliche Zeichen in MENGE2 werden ignoriert.

Nur **[:lower:]** und **[:upper:]** werden mit Sicherheit in aufsteigender Reihenfolge expandiert. In MENGE2 dürfen sie zum Wandeln nur in Paaren benutzt werden, um eine Groß-/Kleinschreibung anzuzeigen.

-s benutzt MENGE1, wenn nicht umgewandelt oder gelöscht wird; anderenfalls wird MENGE2 zum Verdichten benutzt und erscheint nach Wandlung und Löschung.

Hinweis: tr entwickelt seine besonderen Stärken in Shellskripte.

Beispiele:

```
for i in *.mp3;do mv "$i" "$(echo "$i" | tr " " _);done
```

Ersetzt im aktuellen Verzeichnis alle Leerzeichen in den Dateinamen der MP3-Dateien durch einen Unterstrich (_).

```
cat datei.txt | tr "\n" ' ' > datei1.txt
```

Ersetzt in der Datei datei.txt den Zeilenumbruch durch ein Leerzeichen und schreibt das Ergebnis in die Datei datei1.txt.

```
echo "in diesem      Text befinden sich      teilweise zu viele  
Leerzeichen" | tr -s [:blank:] ' '
```

Ersetzt 2 oder mehr Leerzeichen durch ein einzelnes Leerzeichen - die Ausgabe erfolgt auf der Standardausgabe.

```
PFAD="/home/Bilder/2005/August"; echo ${PFAD#/} | tr / " "
```

Ersetzt den Slash (/) im Verzeichnispfad durch ein Leerzeichen, dabei wird der erste Slash vollkommen gelöscht und nicht durch ein Leerzeichen ersetzt.

```
ls | tr [:upper:] [:lower:] | grep -oP '[^\\]+$' | sort | uniq -c | sort ... listet
```


die Anzahl der einzelnen Dateitypen (txt, odt, jpg ...) auf und gibt am Schluss die Gesamtsumme aller Dateien (ohne Verzeichnisse) des aktuellen Verzeichnisses (ohne Unterverzeichnisse) aus

siehe auch: `tr --help`, `sed`, Anhang: Einführung in die Shellprogrammierung

traceroute

Mit `traceroute` kann die Route zu einer IP-Adresse oder eines Domainnamens ermitteln.

`traceroute <host>`

*** ... tauchen in der Route Sternchen auf, so ist der Rechnername unbekannt oder es ist das von `traceroute` verwendete Protokoll gesperrt
-w [n] ... Timeout, warten für [n] Sekunden bis zum Senden des nächsten Paketes

-m ... max. Zahl der Hops, der Default-Wert ist 30; z.B. `traceroute -m 15 www.easylinux.de`

Hinweis: Übersichtlicher zeigt `mtr` (My Traceroute) die Route eines Datenpaketes an. `mtr` gehört aber nicht zur Standardausrüstung eines Linux-Rechners, es muss erst nachinstalliert werden.

Mit den Kommandozeilentools `ifconfig`, `host`, `route`, `ping`, `traceroute` bzw. `mtr` und `netstat` diagnostizieren Sie bei Netzwerkstörungen - richtig eingesetzt - gezielt jeden Teilaspekt einer Verbindung, intern oder im Internet. Damit stellen Sie genau fest, wo alles glatt läuft oder wo es hapert - die wichtigste Voraussetzung, um die Ursache einer Störung zu beheben. So finden Sie möglicherweise heraus, ob es sich bei einer nicht funktionierenden internen oder Internet-Verbindung um ein Problem mit einer Anwendung, der Systemkonfiguration, der Namensauflösung per DNS oder eines der Gegenseite handelt.

siehe auch: `man traceroute`, `host`, `route`, `ping`, `netstat`, `mtr`, `ifconfig`

TurboPrint

Die Linux-Distributionen bringen eine große Auswahl von Druckertreibern mit, aber nicht immer sind die gewünschten aktuellen Treiber dort zu finden.

TurboPrint (www.turboprint.de) enthält viele optimierte Druckertreiber, vor allem für Tintenstrahldrucker z.B. von Canon, Epson, HP, Brother - die Software sollte aber registriert werden, um wirklich brauchbar zu sein. Ohne

Registrierung wird jede Druckseite mit einem TurboPrint-Logo »verziert«.

Wie TurboPrint installiert und konfiguriert wird ist den TurboPrint-Webseiten zu entnehmen bzw. den Internet-Erfahrungsberichten von TurboPrint-Benutzern.

siehe auch: INTERNET, www.turboprint.de

UEFI - Unified Extensible Firmware Interface

siehe auch: EFI

UFW

Ein frisch aufgesetztes Ubuntu-System oder Systeme die auf Ubuntu basieren (Linux Mint, Xubuntu, Lubuntu ...) öffnen aufgrund der "Keine-offenen-Ports"-Regel keine Ports in die angebundenen Netzwerke. Diese Regel macht das Einrichten einer Firewall auf einem Ubuntu-System meist überflüssig. Einfach zu merken: Wo nichts lauscht, gibt's auch keine Lauschangriffe. Mit der Uncomplicated Firewall (kurz UFW) installiert Ubuntu zwar auch ein einfach zu bedienendes Kommandozeilen-Frontend für Iptables, doch UFW ist von Hause aus nicht aktiv.

Eine Personal Firewall hat prinzipiell zwei Aufgaben:

- Sie blockiert Zugriffe aus dem Internet auf Dienste, die auf dem Rechner laufen.
- Sie blockiert ebenfalls unerwünschte Zugriffe vom Computer auf das Internet für Programme, die man absichtlich oder unabsichtlich (Viren, Trojaner, versteckte Spionageprogramme) auf seinem Computer installiert hat.

Uncomplicated Firewall

Seit Ubuntu 8.04 ist in Ubuntu und in den Linux-Distributionen die auf Ubuntu basieren (Linux Mint, Xubuntu, Lubuntu ...) das Paket ufw (uncomplicated firewall. dt. unkomplizierte Firewall) enthalten. ufw ist nichts anderes als ein Verwaltungswerkzeug, um Firewall-Regeln auf dem Level des Kernels zu generieren. Es stellt also keine neue Technik zum Abwehren von Angriffen dar, sondern bedient sich zweier etablierter und in Ubuntu enthaltener Werkzeuge:

- netfilter ... Vom technischen Standpunkt aus gesehen befindet sich in jedem Linux-Kernel eine Firewall, die mit netfilter-Kernel-Modulen realisiert ist. Um diese Kernel-Module zu nutzen, sind allerdings Regeln notwendig, die ein spezielles Filtern explizit erlaubt oder verbietet. Ohne diese Regeln sind die Kernel-Module untätig.
- Iptables ... Im Userspace (also außerhalb des Kernels) befindet sich das zweite Werkzeug: die iptables. Die iptables sind ein weit verbreitetes Tool, das auch in Ubuntu standardmäßig installiert ist. Allerdings sind in Ubuntu keinerlei Regeln für die iptables

definiert.

Die uncomplicated firewall wurde entwickelt, um das Erstellen von Firewall-Regeln zu vereinfachen. iptables besitzt leider eine sehr komplizierte Syntax, sodass das Erstellen eigener Regeln zu Beginn sehr zeitaufwendig ist.

So müssen Sie normalerweise für iptables folgendes Kommando verwenden, um die Verbindungen einer spezifischen IP-Adresse (192.168.1.12) zu blockieren:

sudo iptables -A INPUT -s 192.168.1.12 -j REJECT

Mit ufw verwendet man für den gleichen Zweck das folgende Kommando:

sudo ufw deny from 192.168.1.12

Der Befehl ist durch ufw nicht nur kürzer, sondern auch lesbarer und damit für den Administrator verständlicher geworden. Man sollte dabei aber nicht außer Acht lassen, dass ufw im Hintergrund trotzdem weiterhin iptables verwendet. ufw fungiert quasi lediglich als Übersetzer von einem Kommando in ein anderes. Das Einsatzgebiet liegt hauptsächlich im Serverbereich. Hier spart das einfache Erstellen von Firewall-Regeln wertvolle Zeit und Nerven. Mit anderen Worten ufw ist nur ein einfacher »Kommando-Übersetzer«.

Kommando	Bedeutung
ufw enable	Die Firewall einschalten
ufw disable	Die Firewall ausschalten
ufw default allow	Alle Verbindungen standardmäßig erlauben
ufw default deny	Alle Verbindungen standardmäßig verbieten
ufw status	Zeigt den aktuellen Status und Regeln an
ufw allow 'port'	Erlaube Traffic auf 'port'
ufw deny 'port'	Verbiete Traffic auf 'port'
ufw deny from 'ip'	Blockiere eine spezielle 'ip'
ufw app list	Übersicht über alle aktuellen Applikationsfilter

sudo ufw enable ... die Firewall wird scharf geschaltet und in der Default-Einstellung sind jetzt alle Port's geschlossen; beim nächsten Neustart des Rechners wird die Firewall automatisch gestartet

sudo ufw status ... der Status der Firewall; aktiv oder nicht aktiv; anzeigen der aktivierten Regeln, d.h. welche Port's geschlossen sind und welchem Rechner der Zugang verwehrt wird

sudo ufw disable ... deaktiviert die Firewall; diese Einstellung bleibt auch nach einem Rechner-Neustart erhalten

sudo ufw allow proto tcp from 192.168.1.66 to 192.168.1.53 port 80 ... der Rechner mit der IP-Adresse 192.168.1.66 darf auf den Webserver (Port 80) vom Rechner 192.168.1.53 zugreifen

sudo ufw delete allow proto tcp from 192.168.1.66 to 192.168.1.53 port 80 ... eine Regel muss so gelöscht werden, wie sie erstellt wurde – nur mit vorangestellten **sudo ufw delete**

sudo ufw status verbose ... ausführlichere Statusangaben

sudo ufw deny from 192.168.220.115 to 173.194.69.138 ... dem Rechner mit der IP-Adresse 173.194.69.138 ist es verboten auf dem Rechner 192.168.220.115 (alle Protokolle, alle Ports) zuzugreifen

sudo ufw delete deny from 192.168.220.115 to 173.194.69.138 ... eine Regel muss so gelöscht werden, wie sie erstellt wurde – nur mit vorangestellten **sudo ufw delete**

Beispiel: Web- und FTP-Server

Auf einem Server sind ein Webserver (Port 80) und ein FTP-Server (Port 20, 21) installiert und betriebsbereit. Auf diesen Server können alle Rechner des Netzwerkes zugreifen.

1. **sudo ufw enable**
2. **sudo ufw allow 80**
3. **sudo ufw allow 20**
4. **sudo ufw allow 21**

Im Laufe der Zeit stellt sich heraus, dass es nicht sinnvoll ist, dass der Rechner mit der IP 10.102.20.202 auf den FTP-Server zugreifen kann.

1. **sudo ufw deny from 10.102.20.202 port 20**
2. **sudo ufw deny from 10.102.20.202 port 21**

Während einer Wartungsmaßnahme am Webserver wird der Zugriff auf den Webserver für alle gesperrt.

1. **sudo ufw deny 80**

Nach der Beendigung der Wartungsarbeiten wird der Webserver wieder frei geschaltet.

2. **sudo ufw allow 80**

Um die Leistungsfähigkeit des Webserver zu erhöhen, wird der FTP-Server auf eine eigene Maschine ausgelagert. Die Regeln für den Zugang zum FTP-Server können damit gelöscht werden.

1. **sudo ufw delete allow 20**

2. **sudo ufw delete allow 21**

3. **sudo ufw status**

Log-Dateien anzeigen:

Die Logeinträge der Firewall werden entweder in der Datei messages (/var/log/messages) oder in ufw.log (/var/log/ufw.log) gespeichert. In welcher Datei die UFW-Log-Einträge gespeichert werden, hängt von der verwendeten Linux-Distribution ab. Linux Mint speichert die Log-Einträge in der Datei /var/log/ufw.log.

sudo cat /var/log/messages | grep UFW ... die mit grep gefilterten Log-Einträge der Firewall anzeigen; von Interesse sind eigentlich nur die Einträge in denen der Wert BLOCK steht

sudo tail -f /var/log/messages ... Log-Einträge des Kernels fast in Echtzeit betrachten

sudo cat /var/log/ufw.log ... Log-Einträge der aktivierten Firewall anzeigen

sudo tail -f /var/log/ufw.log ... Log-Einträge der Firewall fast in Echtzeit betrachten; die Log-Einträge werden durch den Kernel generiert

Beispiel:

```
karl@tux ~ $ sudo tail -f /var/log/ufw.log
Jun 15 13:42:02 tux kernel: [ 4705.372236] [UFW BLOCK]
IN=ppp0 OUT= MAC= SRC=205.128.68.254 DST=10.230.199.92
LEN=52 TOS=0x00 PREC=0x00 TTL=61 ID=28618 DF PROTO=TCP
SPT=1935 DPT=49198 WINDOW=32851 RES=0x00 ACK URG=0
Jun 15 13:46:16 tux kernel: [ 4959.231680] [UFW BLOCK]
IN=ppp0 OUT= MAC= SRC=74.125.136.84 DST=10.230.199.92
LEN=117 TOS=0x00 PREC=0x00 TTL=45 ID=57189 PROTO=TCP
SPT=443 DPT=39534 WINDOW=670 RES=0x00 ACK PSH URG=0
```

```
Jun 15 13:46:16 tux kernel: [ 4959.271457] [UFW BLOCK]  
IN=ppp0 OUT= MAC= SRC=74.125.136.84 DST=10.230.199.92  
LEN=97 TOS=0x00 PREC=0x00 TTL=45 ID=57190 PROTO=TCP SPT=443  
DPT=39534 WINDOW=670 RES=0x00 ACK PSH URGP=0
```

Mit diesen Angaben (SRC=74.125.136.84, SRC steht für SOURCE, Quelle) ist es z.B. möglich zu ermitteln, woher ein vermeintlicher Angriff auf den Rechner erfolgt ist. Im Internet gibt es einige Anbieter (z.B. www.utrace.de) die den Standort eines Rechners über die IP-Adresse (Signallaufzeit, IP-Adressen-Bereiche die einem Land, Region zugeordnet werden können, etc.) ermitteln können. Diese Anbieter benutzen i.d.R. eine bereits bestehende Datenbank, das zugrunde liegende Verfahren nennt sich GEO-Targeting. Die Richtigkeit der Angaben über das GEO-Targeting liegt nach meinen Erfahrungen bei weit über 90 Prozent.

Hinweis: Nicht jeder UFW BLOCK ist ein ernst gemeinter Angriff. Einige Programme und Internet-Dienste wollen einfach nur etwas mehr über die Rechner, die am Internet angeschlossen sind, erfahren (Suchmaschinen, Wetterdienste, Spieleanbieter, etc.)

host -a 74.125.136.84 ... host zeigt alle verfügbaren Informationen; falls dieser IP-Adresse ein Servernamen zugeordnet ist, so wird auch dieser angezeigt

Für weitere Informationen ist auf der Webseite www.wikipedia.org oder in einer beliebigen Suchmaschine das Suchwort »GEO-Targeting« einzugeben.

Allgemeine Informationen zu den Ports:

Insgesamt gibt es 65535 Ports, von denen die wichtigsten im Bereich zwischen 1 und 1000 liegen.

Web-Seiten im HTTP-Protokoll werden über Port 80 versandt, sichere Web-Seiten über Port 443. Der Mailempfang per POP3 erfolgt über Port 110, das sichere POPS per Port 995, der Versand per SMTP über Port 25. Wer einen IMAP-Server ansprechen will, benötigt den Port 143.

Der Datenaustausch per FTP findet über Port 20 statt, FTP-Befehle erfolgen über Port 21.

Über den Port 22 erreichen Sie entfernte Rechner per SSH.

ICQ verwendet den Port 5190, der Yahoo- Messenger 5050 und Skype den Port 443 und 12452.

Samba (Netzwerkressourcen, Freigaben in einem Windows-Netzwerk) benötigt die Ports 137, 138, 139 und 445 (**siehe auch:** `sudo ufw app list`; `sudo ufw app info Samba`).

siehe auch: `ufw -h`, `man ufw`, `host`, `traceroute`, `mtr`, `chroot`, `hosts.allow`, `hosts.deny`

Umleitung von Befehlen

Es kann mitunter sinnvoll sein, die Ausgabe eines Programms umzuleiten. Dies kann z.B. zu einem anderen Befehl, der diese weiterverarbeitet, oder in eine Datei (oder ein Gerät) erfolgen. Diese zwei verschiedenen Formen der Umleitung sollen nun erklärt werden.

Umleiten in eine Datei

Ein typisches Problem: Wir haben ein Verzeichnis, dessen Inhalt katalogisiert werden soll. Um Rechte und Größe der einzelnen Dateien festzuhalten, eignet sich das Tool **ls** (welches den Befehl `dir` aus DOS entspricht) gut, wenn man es mit dem Parameter **-l** kombiniert. (Mitunter ist dieser Schritt nicht notwendig, weil je nach Distribution dieser Parameter als Standard eingestellt ist). Leider (oder in den meisten Fällen glücklicherweise) gibt **ls -l** seine Ausgabe auf dem Bildschirm aus. Wie bekommen wir die Ausgabe in eine Datei? Der Operator **>** hilft uns weiter. Er sorgt dafür, dass die Ausgabe des Programms nicht auf den Bildschirm, sondern auf die angegebene Datei erfolgt.

Beispiel 1: `ls -l > ~/listing.txt` ... speichert den Verzeichnisinhalt des aktuellen Verzeichnisses in die Datei `listing.txt` im Heimatverzeichnis des Benutzers.

Beispiel 2: `echo "<?php phpinfo(); ?>" > /srv/www/htdocs/index.php` ... erzeugt im angegebenen Verzeichnis eine PHP-Datei

Beispiel 3: `sort` adressen | `uniq` | `less`

Das `sort`-Kommando liest die Datei `adressen` und sortiert deren Inhalt zeilenweise. Die sortierten Zeilen, werden dem Programm `uniq` zugeführt, dass die Zeilen vergleicht und alle Zeilen entfernt, die doppelt vorkommen. Die Ausgabe von `uniq` wird wiederum dem Programm `less` zugeführt, dass die Adressendatei von den Doppeln befreit, sortiert anzeigt.

Umleitung auf Geräte

Die Umleitung kann auch auf Geräte erfolgen, was jedoch nur bedingt Sinn macht. Soll z.B. die Ausgabe eines Programms direkt (nicht als Datei) auf

USB-Stick geschrieben werden (jeglicher bisheriger Inhalt des USB-Sticks geht verloren!), so wird die Ausgabe an den USB-Stick (im Regelfall /dev/sdd1) weitergeleitet.

Ein sinnloses Beispiel: **ls -l > /dev/sdd1** gibt die Liste der Dateien im aktuellen Verzeichnis an den USB-Stick ... Dieses Beispiel macht wenig Sinn, da diese Liste praktisch nicht mehr auslesbar ist und den Inhalt des USB-Sticks zerstört.

Beispiele:

Umleitung nach /dev/null, also dem Datennirvana unter Linux. Dies ist vor allem dann sinnvoll, falls einem die Zwischenmeldungen eines Shellskripts stören. Diese werden vom Bildschirm zum Pseudogerät /dev/null umgeleitet.

shellscript.sh start 30 > /dev/null

Dasselbe wie vorher, nur das der Prozess im Hintergrund ausgeführt wird. Durch das &-Zeichen wird die Blockierung der Konsole während der Abarbeitung des Shellskripts aufgehoben (siehe auch: Hintergrundprozess starten).

shellscript.sh start 30 > /dev/null &

Umleitung an ein Programm (via Pipe)

Wie bereits oben beschrieben kann es nötig sein, die Ausgabe einer Datei umzuleiten, um sie besser verarbeiten zu können. Die häufigste Anwendung ist die Weiterleitung an die Programme **less** oder **more**. Diese beiden Programme sorgen dafür, dass die Ausgabe eines Programms auch dann vollständig zu lesen ist, wenn sie sowohl größer als der Bildschirm, als auch größer als der Ausgabepuffer ist. Dies kommt besonders häufig beim Listing langer Verzeichnisse oder ähnlichen Vorgängen vor. Ein ausführliches Listing des Geräteordners /dev zeigt dies anschaulich, da dieser Ordner mitunter weit über 1500 »Dateien« enthält.

Folgendes Beispiel soll die Funktionsweise der Pipe verdeutlichen, macht jedoch wiederum wenig Sinn: **ls -l /dev | less**. Dieser Befehl gibt eine Liste der Geräte aus, die sich mit den Bildlauf-Tasten [Bild hoch] und [Bild runter] beliebig scrollen lässt. Die Ausgabe lässt sich mit der Taste [q] beenden.

Ein weiteres Anwendungsgebiet der Pipe ist die Suche. So lässt sich z.B. über den Befehl **grep** in Verbindung mit dem Befehl **cat** eine Reihe von Dateien in einem Verzeichnis durchsuchen: **cat * | grep Blubberbläschen**. Diese Befehlsreihe durchsucht alle Dateien im aktuellen Verzeichnis nach dem Begriff Blubberbläschen. Anzumerken ist hier, dass **grep** die Suche unter Beachtung von Groß- und Kleinschreibung durchführt (was mit dem Schalter **-i** verhindert werden kann). Der Befehl **cat** gibt in diesem Fall den

Inhalt aller Dateien nacheinander an die Pipe weiter (normalerweise auf den Bildschirm) die zum Befehl **grep** führt. Dieser durchsucht jede einzelne Zeile nach dem Wort **Blubberbläschen** und gibt die Zeilen aus, in denen das Wort vorkommt.

Beispiele:

ls /home/* > home.txt

Einen Haken hat diese Umleitung mit > allerdings - existiert die Datei home.txt bereits, überschreiben Sie diese mit den neuen Daten.

ls /home/* >> home.txt

Existiert die Datei home.txt bereits, hängen Sie die neuen Daten an die evt. bereits vorhandenen Daten einfach an.

ls /home/* 2> /dev/null

Hier leiten Sie evt. Fehlerausgaben (z.B. bei ungenügenden Zugriffsrechten auf Dateien und Verzeichnissen) nicht in eine Datei, sondern nach /dev/null um, das ist eine »Pseudo-Datei« ohne Inhalt - d.h. die Fehlerausgaben verschwinden ins Daten-Nirvana.

Alle anderen Daten werden an die Standardausgabe umgeleitet, also im Normalfall auf den Bildschirm.

Um den Inhalt der lesbaren Verzeichnisse in eine Datei abzuspeichern und die Fehler wegzuworfen kombinieren Sie die Operatoren wie folgt:

ls /home/* >> home.txt 2> /dev/null

Soll die Fehlerausgabe hingegen nicht verschwinden, sondern zusammen mit der Standardausgabe abgelegt werden, setzen Sie das Kaufmannsund-Zeichen nach dem Operator:

ls /home/* >& home.txt

sort < unsortiert.txt

Die unsortierte Liste mit Begriffen - z.B. Städtenamen - werden an das Programm sort übergeben und auf der Standardausgabe, dem Bildschirm, ausgegeben.

Sinnvoll ist dieser Operator < vor allem bei Programmen die typischerweise interaktiv arbeiten, wie z.B. die Programme ftp oder gpg.

Eine andere Möglichkeit mehrere Befehle in einem Zuge auszuführen, ist die Verwendung des Sonderzeichens [;] (Semikolon). Mittels des Semikolons können mehrere Befehle in einer Zeile des Kommandozeileninterpreters eingetragen werden.

Beachte: Die Befehle werden unabhängig voneinander ausgeführt, d.h. es werden keine Ergebnisse von einem Befehl an den anderen Befehl übergeben.

Beispiel:

cd /opt;ls -al

Im Beispiel wird zuerst ins Verzeichnis /opt gewechselt und anschließend alle Dateien dieses Verzeichnisses im Langformat angezeigt.

Prozesse

Prozesse, die im Hintergrund laufen, geben ihre Ausgaben auf das Terminal aus. Wer dies vermeiden will, leitet die Ausgabe in eine Datei oder nach »/dev/null« um, z.B. mit »**Kommando 2**> /dev/null«.

Standardeingabe **stdin: 0**

Standardausgabe **stdout: 1**

Standardfehlerausgabe **stderr: 2**

Um die Standardausgabe und Standardfehlerausgabe gleichzeitig umzuleiten, sind zwei Methoden möglich, wobei die zweite Variante einfacher ist:

Kommando 1> /dev/null **2**>&**1**

Kommando &> /dev/null

Manchmal sollen Prozesse im Hintergrund weiterlaufen, obwohl sich der Administrator mit seiner Login- oder SSH-Shell wieder verabschieden möchte. Dies führt normalerweise zum Abbruch aller darin gestarteten Prozesse, sofern er sie nicht explizit mit »**nohup Kommando &**« abgekoppelt hat. Damit die Ausgaben des Kommandos nicht im Nirwana verschwinden, schreibt Nohup sie in die Datei »**nohup.out**«.

siehe auch: Hintergrundprozess starten, Anhang: Einführung in die Shellprogrammierung

umask

Bevor Sie z.B. ein Verzeichnis anlegen, können Sie mittels des umask Befehls festlegen, welche Zugriffsrechte gleich bei der Erstellung maskiert werden sollen:

umask 027 beschränkt die Rechte der einzelnen Benutzergruppen folgendermaßen: der Besitzer der Datei behält sämtliche Rechte (0), die Besitzergruppe darf nicht schreibend auf die Datei zugreifen (2) und alle anderen Benutzer erhalten keinerlei Zugriff (7). Die Zahlen sind als

Bitmaske zu lesen. Details zu umask entnehmen Sie der entsprechenden Manualpage (man umask).

umask ist praktisch gesehen, ein Filter der von den Standardvorgaben der Zugriffsrechte Rechte wegnimmt (1 .. execute, 2 .. write, 4 .. read). Verzeichnisse werden mit den Zugriffsrechten 777 und Dateien mit den Zugriffsrechten 666 erstellt. Bei einer Standardinstallation der Linux-Distributionen ist für die Benutzer schon eine Standard-umask aktiv - umask 022. Für die Zugriffsrechte bedeutet dies, dass bei der Erstellung von Verzeichnissen das Zugriffsrecht 755 und bei Dateien das Zugriffsrecht 644 automatisch vergeben wird. Die automatische Vergabe dieser Zugriffsrechte kann mit dem Befehl umask abgestellt werden, dabei ist der Befehl vor der Erstellung von Verzeichnissen und Dateien aufzurufen.

Die umask-Maske ist immer nur im aktuellen Terminalfenster aktiv, d.h. beim Schließen des Terminalfensters werden sofort wieder die Standardeinstellungen aktiv - also umask 022.

Typ	Eigentümer			Gruppe			Andere		
d	r	w	x	r	w	x	r	w	x

	4 + 2 + 1			4 + 2 + 1			4 + 2 + 1		

Um umask global beziehungsweise dauerhaft zu ändern, muss sie in eine der Bash-Konfigurationsdateien aufgenommen werden. Für die systemweite Einstellung müssen Sie die »/etc/profile« und für benutzerbezogene z.B. die »~/.bashrc« ändern.

siehe auch: chmod, Zugriffsrechte

umount

umount löst die Verbindung zu einem Dateisystem das in den Linux-Verzeichnisbaum eingebunden ist.

umount <Verzeichnisname>

umount /media/sdd1 ... löst die Einbindung in die Verzeichnisstruktur (Linux-Verzeichnisbaum) wieder und erst jetzt werden alle geänderten Daten - die sich bis jetzt evtl. noch im Cache befinden – auf ein USB-Stick (Datenträger) geschrieben (die Gerätebezeichnung – hier sdd1 - weicht in den verschiedenen Distributionen voneinander ab).

siehe auch: mount

uniq

Alle hintereinander stehenden identischen Zeilen von EINGABE (oder Standardeingabe) bis auf eine löschen, und auf AUSGABE (oder Standardausgabe, Bildschirm) schreiben.

`uniq [OPTION]... [EINGABE [AUSGABE]]`

- c, --count ... den Zeilen die Anzahl des Vorkommens voranstellen
- d, --repeated ... nur die doppelten Zeilen ausgeben
- i, --ignore-case ... Abweichung in Groß/Kleinschreibung ignorieren
- s, --skip-chars=N ... nicht die ersten N Zeichen vergleichen
- u, --unique ... nur einmal vorkommende Zeilen ausgeben
- w, --check-chars=N ... nicht mehr als N Zeichen pro Zeile vergleichen

Beispiele:

`sort adressen | uniq | less` ... Das sort-Kommando liest die Datei **adressen** und sortiert deren Inhalt zeilenweise. Die sortierten Zeilen, werden dem Programm **uniq** zugeführt, dass die Zeilen vergleicht und alle Zeilen entfernt, die doppelt vorkommen. Die Ausgabe von **uniq** wird wiederum dem Programm **less** zugeführt, dass die Adressendatei von den Doppeln befreit, sortiert anzeigt.

`sort adressen | uniq > Adressenliste.txt` ... Im Beispiel wird eine sortierte Adressenliste, die von doppelten Einträgen befreit wurde, in der Datei **Adressenliste.txt** gespeichert.

`ls | tr [:upper:] [:lower:] | grep -oP '[^\.]+\$' | sort | uniq -c | sort` ... listet die Anzahl der einzelnen Dateitypen (txt, odt, jpg ...) auf und gibt am Schluss die Gesamtsumme aller Dateien (ohne Verzeichnisse) des aktuellen Verzeichnisses (ohne Unterverzeichnisse) aus

unix2dos

siehe auch: fromdos, todos

Upstart

Upstart ist ein Ubuntu-Projekt, welches den Systemstart von Ubuntu bzw. allgemein Linux, durch parallelen Start von Diensten, beschleunigt.

Beim Systemstart werden eine Reihe von Programmen und Diensten gestartet (z.B. der XServer für die Grafik, der CUPS-Daemon zum Drucken

etc.), bevor man mit dem System arbeiten kann. Traditionell werden diese Programme bzw. Dienste über eine in den Runleveln des klassischen SysV-Init-Systems festgelegte Reihenfolge geladen. Das Laden erfolgt dabei linear, d.h. ein Dienst nach dem anderen.

Upstart verwendet einen neuen Ansatz, um den Systemstart zu beschleunigen. Programme und Dienste werden ereignisbasiert geladen und gestartet, wodurch mehrere – voneinander unabhängige – Dienste parallel aufgerufen werden können.

Systemstart

Upstart (Ubuntu, CentOS) ersetzt das klassische SysV-Init-System und damit auch die Start-/Stop-Mechanismen von Diensten. Allerdings sind noch lange nicht alle Dienste auf Upstart umgestellt, sondern vorrangig die zum Systemstart (Init-Jobs) erforderlichen. Die Datei `/etc/inittab` entfällt.

Wann ein Init-Job aktiv wird (Runlevel), legen die Konfigurationsdateien im Verzeichnis `/etc/init/` fest. Alle Dateien im Verzeichnis `/etc/init` werden automatisch gestartet. Man muss also z.B. neue Skripte nicht mehr umständlich mit `update-rc.d` (Debian/Ubuntu) aktivieren, wie es beim SysV-Init-System der Fall ist.

Das zentrale Werkzeug ist nun **initctl**, das Init-Jobs startet oder stoppt, Signale verschickt und den Status abfragt. So gibt beispielsweise der Befehl:

initctl list

eine Liste aller Init-Jobs und ihres Status aus.

Dienste starten und anhalten

sudo initctl start [Upstart-Dienstname] ... Dienst starten

sudo initctl stop [Upstart-Dienstname] ... Dienst anhalten

Verhalten von Upstart beeinflussen

Möchte man bestimmte Init-Jobs abändern, so muss man die entsprechende Konfigurationsdatei im Verzeichnis `/etc/init/` editieren.

Run-Level ändern

Um zu verhindern, dass ein Dienst in einem bestimmten Run-Level gestartet

wird, öffnet man die entsprechende .conf-Datei und editiert die Zeile

start on runlevel [2345]

Die Zahlen in den eckigen Klammern, repräsentieren dabei die Runlevel. Entsprechend sollte auch die Zeile

stop on runlevel [!2345]

geändert werden. Das Ausrufezeichen in der Stop-Anweisung bedeutet, dass der Dienst gestoppt wird, wenn sich der Run-Level außerhalb der Aufzählung befindet.

Upstart überwacht die Konfigurationsdateien mittels **inotify**, so dass man beim Testen der Änderungen folgende Meldung erhält:

initctl: Unknown job: Init-Job

Erst mit einem Neustart liest Upstart die geänderten Dateien ein. Schneller (und ohne Rechner-Neustart) geht es mit folgendem Befehl:

sudo initctl reload-configuration

Deaktivieren von Init-Jobs

Wenn man Init-Jobs dauerhaft deaktivieren möchte, kann man entweder

- die Datei /etc/init/jobname.conf umbenennen bzw. deren Endung .conf entfernen oder
- innerhalb der Datei jobname.conf die Zeile mit "start on" auskommentieren

Zum Reaktivieren sind diese Änderungen wieder rückgängig zu machen oder ab Ubuntu 11.10 existiert noch eine dritte Variante: .override-Dateien.

Beispiele:

```
echo "manual" | sudo tee /etc/init/jobname.override
```

oder

```
sudo sh -c "echo 'manual' > /etc/init/jobname.override"
```

Der Vorteil gegenüber den ersten beiden Varianten ist, dass die Originaldatei `/etc/init/jobname.conf` nicht verändert wird. Möchte man einen Init-Job wieder aktivieren, löscht man einfach die `.override`-Datei (oder entfernt das Schlüsselwort "manual" in dieser Datei).

Problembehebung - Job unbekannt

Nach Änderungen an bestehenden Skripten oder bei eigenen Skripten erscheint trotz Einlesen der neuen Konfiguration die Fehlermeldung

initctl: Unknown job: jobname

Es liegt eventuell ein Fehler in der Syntax vor. Das Skript lässt sich mit folgendem Befehl prüfen:

init-checkconf jobname .conf

siehe auch: `systemd` – Das Init-System, `init` (SysVinit), Dienste starten, anhalten und deaktivieren

useradd

`useradd` erstellt ein neues Benutzer-Konto (Account) und verwendet dazu die Einstellungen in der Datei `/etc/default/useradd` und die Optionen die auf der Kommandozeile spezifiziert wurden.

Mittels eines Shellskriptes können mit `useradd` viele Benutzerkonten in einem Zuge automatisch erstellt werden.

Das Benutzer-Konto muss mit einem Buchstaben beginnen, der Rest der Zeichenfolge kann aus diesem Zeichenvorrat entnommen werden:

`[A-Za-z][A-Za-z0-9_-]*[A-Za-z0-9_-.$]`

Dateien für die Benutzerverwaltung:

`/etc/passwd` ... enthält die Benutzer-Konten

`/etc/shadow` ... enthält das verschlüsselte Passwort, sowie Informationen zur Gültigkeit und Ablaufzeit des Kontos

`/etc/default/useradd` ... enthält die Standardeinstellungen für den Befehl `useradd`

`/etc/skel` ... Enthält die Standarddateien für das Home-Verzeichnis des Benutzers. Diese werden beim Anlegen eines neuen Benutzer-Konto in das Home-Verzeichnis des Benutzers kopiert.

`/etc/pam.conf` ... Konfigurationsdatei für PAM (Pluggable Authentication Modules for Linux)

`/etc/pam.d/*` ... diverse Dateien zur Konfiguration von PAM (Pluggable


```
useradd --groups users,dialout,video --home /home/che --shell /bin/bash  
--create-home --password '2CCu.2JCMimEY' che oder  
useradd -G users,dialout,video -d /home/che -s /bin/bash -m -p  
'2CCu.2JCMimEY' che
```

Diese Kommandofolge erzeugt einen neuen Benutzer »che« innerhalb der bestehenden Gruppe »users«, sein Heimatverzeichnis heißt »/home/che«. Gleichzeitig ist er auch Mitglied der Gruppen »dialout« und »video« - das sind die Standardgruppen die ein normaler Benutzer i.d.R. angehört. Als Standardshell wird die Bash-Shell vorgegeben. Mit »--create-home« wird gleichzeitig das Home-Verzeichnis erzeugt, als Passwort wird vom Systemadministrator root »nostromo« festgelegt (Passwort muss verschlüsselt sein und in Hochkommas eingeschlossen sein; siehe auch: man useradd). Welche Verschlüsselung zu verwenden ist, hängt von Ihrem System ab. Der Benutzer »che« sollte nach der ersten Anmeldung das Passwort mittels »passwd« sofort ändern.

Achtung: Im Beispiel wird davon ausgegangen, dass passwd die DES-Verschlüsselung verwendet. Falls das System eine andere Verschlüsselung verwendet, so ist das verschlüsselte Passwort anzupassen (siehe auch: mkpasswd).

Hinweis: Wird ein neues Benutzerkonto ohne die Option password erstellt, so ist dieses Konto solange gesperrt, bis für den neuen Benutzers ein vorläufiges Passwort gesetzt wird (Beispiel: sudo passwd che).

weitere Optionen:

- c, --comment KOMMENTAR ...** Kommentar für das GECOS-Feld des neuen Benutzers
- d, --home-dir HOME_VERZEICHNIS...** Home-Verzeichnis des neuen Benutzers
- g, --gid GID ...** GRUPPEN-ID oder Gruppennamen für neuen Benutzer erzwingen (Hauptgruppe des neuen Benutzers) , die Gruppe users hat i.d.R. die GID 100; die hier eingetragene Gruppe muss bereits existieren (siehe auch: groupadd)
- G, --groups GRUPPEN ...** Liste der zusätzlichen Gruppen für den neuen Benutzer
- m, --create-home ...** Home-Verzeichnis des neuen Benutzers erstellen
- M, --no-create-home ...** kein Home-Verzeichnis für den Benutzer erstellen (überschreibt /etc/login.defs)
- p, --password PASSWORT ...** benutze ein **verschlüsseltes** Passwort für den neuen Benutzerzugang (siehe auch: mkpasswd)
- s, --shell SHELL ...** die Login-Shell des neuen Benutzerzugangs

-u, --uid UID ... Benutzung dieser UID (Benutzer-ID) erzwingen ,
numerischer Wert; siehe auch: id
-U, --user-group ... erstelle eine Gruppe mit dem gleichen Namen wie dem
des Benutzers

siehe auch: useradd -h, man useradd

Hinweis: Möglich sind in aktuellen Linux-Distributionen die
Verschlüsselungsverfahren: DES, MD5, BLOWFISH, SHA-256 und SHA-
512. Das verschlüsselte Passwort wird in der Datei **/etc/shadow** gespeichert.

Ein Beispieleintrag könnte in etwa so aussehen (siehe auch: passwd):

username:Xldlkasoo2bn90lsal:12455:0:99999:-1::

Das verschlüsselte Passwort ist die Zeichenfolge nach username, zwischen
den beiden Doppelpunkten (:). Bei einer DES-Verschlüsselung besteht diese
Zeichenfolge aus genau 13 Zeichen. Besteht diese Zeichenfolge aus genau
60 Zeichen, so wird sehr wahrscheinlich die BLOWFISH-Verschlüsselung
verwendet und bei 32 bzw. 34 Zeichen die MD5-Verschlüsselung.

Beispiel: das Passwort soll **nostromo** heißen

DES: **2CCu.2JCMimEY**

MD5: **\$1\$KzO6eCI2\$TGMTzaJpvbfG3TTtAGBbA0**

BLOWFISH:

**\$2a\$10\$g4e99hx0AWogZLfNMR789uUd8.VQenw1ndXEYahNO03hD
uabE7J.2**

SHA-512:

**\$6\$19uKKdas/ehHdq0E\$IRlxcO1uQcRV6PVvEi2m3IHRI5t4xJB7bgTv
WQH0uRX/m.WRHXxDx2t3GFVqXcFagiW.odTgCBt5UtUGZ03wfl**

siehe auch: groupadd, userdel, groupdel, mkpasswd, passwd

userdel

Löschung eines Benutzerkontos.

userdel <Optionen> <Benutzername>

userdel --remove karl ... der Benutzeraccount »karl« wird samt seines
Home-Verzeichnisse gelöscht, sowie die Mail-Verzeichnisse (Mail-Spool-
Verzeichnis) und Dateien des Kontos; Dateien in anderen Verzeichnissen
müssen manuell ermittelt und gelöscht werden. System-Benutzer die nach
einer Standardinstallation von Linux eingerichtet werden, sollten nicht

gelöscht werden - sie werden vom System benötigt.

userdel -rf karl ... der Benutzeraccount »karl« wird samt seines Home-Verzeichnisse gelöscht, auch wenn der Benutzer karl zur Zeit angemeldet ist (Option: f)

usermod -R [Gruppenname] [Benutzername] ... Benutzer aus einer Gruppe entfernen

siehe auch: userdel -h, useradd, usermod, groupadd, groupdel

usermod

Modifizierung eines Benutzerkontos.

usermod <Optionen> <Benutzername>

usermod -m -d /home/ben2 ben ... das Home-Verzeichnis vom Benutzer ben wird nach /home/ben2 verschoben

-l Login-Name ... der neue Login-Name für das Konto

-m, --move-home ... das Home-Verzeichnis des Benutzers wird auf das unter -d spezifizierte Verzeichnis verschoben

-d, --home HOME_DIR ... neues Home-Verzeichnis

-f INAKTIV ... Passwort nach Ablauf von INAKTIV (z.B. 7 ... sieben Tage) deaktivieren

-a, --append ... Benutzer zu zusätzlichen Gruppen hinzufügen, die mit der Option -G angegeben werden, ohne ihn dabei aus anderen Gruppen zu entfernen

-G, --groups GRUPPEN ... neue Liste zusätzlicher GRUPPEN

-L, --lock ... Benutzerzugang sperren

-U, --unlock ... Benutzerzugang entsperren

-p, --password PASSWORD ... ein verschlüsseltes Passwort als neues Passwort verwenden (siehe auch: mkpasswd)

Beispiele:

sudo usermod -a -G sudo karl ... Benutzer karl wird zusätzlich in der Gruppe sudo aufgenommen; die primäre Mitgliedschaft (die erste Gruppenmitgliedschaft des Benutzers; **siehe auch:** groups oder id) des Benutzers karl bleibt unberührt;

Hinweis: Die neuen Gruppenzugehörigkeiten sind immer erst nach einer erneuten Anmeldung des betroffenen Benutzers gültig.

usermod -R sudo karl ... Benutzer karl wird aus der Gruppe sudo wieder entfernt

Anmerkung: Mit dem Terminalprogramm gpasswd kann ein Benutzer aus einer Gruppe oder mehreren Gruppen entfernt werden (siehe auch: /etc/group).

siehe auch: usermod -h, userdel, useradd, gpasswd, groups

unzip

unzip entpackt Zip-Dateien.

unzip <Dateiname> ... benanntes ZIP-Archiv entpacken

unzip *.zip ... alle ZIP-Archive des aktuellen Verzeichnisses entpacken; der Backslash vor dem Asterisk sagt der Bash-Shell, dass der Stern (*) als Zeichen unverändert an unzip zu übergeben ist

siehe auch: gzip, tar, p7zip

unrar

unrar entpackt RAR-Dateien.

unrar <Dateiname>

siehe auch: gzip, tar p7zip

UTF-8

siehe auch: Konvertierung von Textkodierungen, iconv, fromdos, todos, recode

uuencode

Das Programm uuencode ist Bestandteil des Pakets sharutils. Das Programm uuencode bereitet eine Datei auf die Übertragung über einen elektronischen Kanal vor, der sonst das achte Bit (high order bit, Email-Server) der Bytes verstümmeln würde.

Das Programm uudecode führt die gegenteilige Umwandlung durch.

uuencode [-m] [file] name

uuencode -m picture.tar.gz image.tar.gz > picture.txt ... Umwandlung des Archivs picture.tar.gz in den base64-Code (Option: -m) und Speicherung in der Datei picture.txt; image.tar.gz ist der Name den

uudecode für die Konvertierung in den ursprünglichen Bytecode verwendet

uuencode -m picture.tar.gz image.tar.gz | tee picture.txt ... das Ergebnis ist dasselbe wie vorherigen Beispiel; zusätzlich erfolgt aber eine Bildschirmausgabe des base64-Codes

siehe auch: man uuencode, uudecode

uudecode

Das Programm uudecode ist Bestandteil des Pakets sharutils. Das Programm uudecode konvertiert Daten die von uuencode kodiert wurden, wieder in den ursprüngliche Bytecode.

uudecode [-o outfile] [file] ...

uudecode picture.txt ... Daten werden in den ursprünglichen Bytecode konvertiert

siehe auch: man uudecode, uuencode

Version des Kernels

cat /proc/version ... zeigt die Version des Kernels an

uname -a ... zeigt ebenfalls die Kernelversion

siehe auch: `uname --help`, `proc-Dateisystem`

Virens Scanner

Der bekannteste Virens Scanner für Linux ist sicherlich der Open-Source-Scanner ClamAV. ClamAV kann als Bibliothek in eigene Programme eingebunden werden, steht jedoch auch als Dämon und als Kommandozeilenprogramm zur Verfügung. Neben ClamAV gibt es noch weitere Virens Scanner wie beispielsweise den McAfee vscan für Linux. Jedoch sollte man nicht vergessen, dass diese Scanner fast ausschließlich nach Viren für »Fremdbetriebssysteme« suchen und deshalb auch vor allem auf Mail- oder Fileservern eingesetzt werden.

ClamAV installieren Sie ganz einfach über Synaptic. Die jeweils neue Version von ClamAV befindet sich in der Universe-Sektion. Das zu installierende Paket heißt `clamav`. Sie können ClamAV natürlich auch über

sudo apt-get install clamav

installieren. Bitte achten Sie darauf, dass Sie dies per `sudo`, nicht als Root, tun. ClamAv wird als Benutzer im Terminal mit dem Kommando `clamscan` gestartet.

clamscan

Dabei werden die gescannten Verzeichnisse/Dateien angezeigt. Zunächst können folgende einfache Scan-Befehle verwendet werden (alle als normaler User ohne Root-Rechte):

- **clamscan hallo.pdf** ... scannt die Datei `hallo.pdf` im aktuellen Verzeichnis.
- **clamscan /etc** ... scannt das Verzeichnis `/etc` ohne die Unterverzeichnisse.
- **clamscan -r /etc** ... führt einen rekursiven Scan des Verzeichnisses `/etc` und aller Unterverzeichnisse durch.
- **sudo freshclam** ... führt ein Update der Virendefinitionen aus.

Der Befehl

clamscan -ril /home/user/Desktop/clamscan.txt --bell --remove --unrar=/usr/bin/unzip --tgz=/bin/tar /home

scannt das Home-Verzeichnis inklusive Unterverzeichnissen, schreibt eine Logdatei (clamscan.txt) nach /home/user/Desktop, piepst bei einem Virenfund, löscht das Virus und benutzt unzip (für *.zip) und tar (für *.tar.gz).

siehe auch: clamscan -h

Virens Scanner ClamAV von CD-ROM starten

ClamAV ist Bestandteil der Live-CD PartedMagic. Die ISO-Datei finden Sie auf der Webseite **<http://partedmagic.com>**. Die heruntergeladene ISO-Datei können Sie dann mit einem beliebigen Brennprogramm als Abbild (CD-Image) auf CD brennen (**siehe auch:** Links → Linux-Live-Distributionen).

Starten Sie den Rechner mit der PartedMagic-Live-CD. Nach erfolgreichem Start können Sie die CD entfernen, denn PartedMagic befindet sich nun vollständig im Hauptspeicher des Rechners.

Als Erstes ändern Sie das Tastaturlayout durch einen Mausklick auf das entsprechende Icon auf dem Desktop.

ClamAV starten Sie innerhalb eines Terminals mit dem Befehl **clamscan**.

Die aktuelle Datenbasis mit den Virensignaturen finden Sie auf der Webseite **www.clamav.org** bzw. **www.clamav.net**. Die Datenbanken mit den Virensignaturen heißen **main.cvd** und **daily.cvd**.

Falls Sie Probleme haben, mit der PartedMagic Linux-Live-Distribution ins Internet zu kommen, so gehen Sie bitte wie folgt vor.

1. Installieren Sie auf Ihren Ubuntu-Linux-Rechner das Programm ISO-Master (Paketname: isomaster).
2. Erstellen Sie ein Verzeichnis clamav und kopieren Sie die Datenbanken mit den Virensignaturen (main.cvd, daily.cvd) in dieses Verzeichnis. **Anmerkung:** Legen Sie dort evtl. auch eine Textdatei mit einigen Hinweisen und den gesamten ClamAV-Befehl ab. Diesen können Sie dann später in das Kommandozeilen-

Fenster von PartedMagic einfach hinein kopieren.

3. Öffnen Sie mit ISO-Master die ISO-Datei pmagic-4.8.iso. Im oberen Programmfenster bewegen Sie sich zu dem neu erstellten Verzeichnis clamav. Dies Verzeichnis fügen Sie in das untere Programmfenster ein. Es befinden sich dann 3 Verzeichnisse (boot, pmagic, clamav) im unteren Fenster.
4. Speichern Sie die ISO-Datei unter einen neuen Namen (z.B. pmagic-4.8_clamav_08-03-2010.iso) ab.
5. Diese neue ISO-Datei brennen Sie nun mit einem beliebigen Brennprogramm als CD-Abbild (CD-Image) auf eine CD.

Nun können Sie den Rechner mit der angepassten PartedMagic-Live-CD starten. Bei dieser Variante ist die CD nach erfolgten Start, wieder ins CD-Laufwerk einzulegen.

Als Erstes ändern Sie das Tastaturlayout durch einen Mausklick auf das entsprechende Icon auf dem Desktop.

Mounten Sie nun die zu untersuchende Partition und das CD-Laufwerk (Icon »Mount Devices«), alternativ können Sie dies auch mit den Dateimanager erledigen (Icon »My Documents«).

Die Namen der Partition bzw. des CD-Laufwerkes, so wie er in der Befehlszeile einzutragen ist, erfahren Sie durch den Kommandozeilen-Befehl »mount«.

Falls Sie den gesamten Befehl in die vorgenannte Textdatei im Verzeichnis clamav gespeichert haben, können Sie diesen nun in das geöffnete Terminal einfügen. Öffnen Sie dazu die Textdatei, markieren Sie die entsprechende Zeile und kopieren sie ([Strg]+[C] bzw. rechte Maustaste) und fügen ihn in das geöffnete Terminalfenster ein ([Strg]+[Shift]+[V] bzw. rechte Maustaste). Passen Sie evtl. noch die Pfade für database und die Zielpartition an.

```
root@PartedMagic:~# clamscan --database=/media/cdrom1/clamav --infected --recursive --bell /media/hda1
```

```
----- SCAN SUMMARY -----  
Known viruses: 726064  
Engine version: 0.95.3  
Scanned directories: 116  
Scanned files: 250  
Infected files: 0  
Data scanned: 7.36 MB
```


Data read: 6.53 MB (ratio 1.13:1)
Time: 13.898 sec (0 m 13 s)
root@PartedMagic:~#

Falls der Virens Scanner einen Virus gefunden hat, so geben sie einen der folgenden Befehle ein.

```
root@PartedMagic:~# clamscan --database=/media/cdrom1/clamav --log=/root/clamav/clamav_logfile.txt --infected --recursive --remove=yes --bell /media/hda1
```

Sucht und entfernt den Virus und speichert das Ergebnis in eine Log-Datei.

```
root@PartedMagic:~# clamscan --database=/media/cdrom1/clamav --log=/root/clamav/clamav_logfile.txt --infected copy=/root/clamav/quarantaene --recursive --remove=yes --bell /media/hda1
```

Sucht und entfernt den Virus und speichert das Ergebnis in eine Log-Datei. Zusätzlich wird der Virus in das Verzeichnis quarantaene kopiert.

```
root@PartedMagic:~# clamscan --database=/media/cdrom1/clamav --log=/root/clamav/clamav_logfile.txt --infected move=/root/clamav/quarantaene --recursive --bell /media/hda1
```

Sucht und verschiebt den Virus in das Verzeichnis quarantaene und speichert das Ergebnis in eine Log-Datei.

Anmerkung: Die verwendete Linux-Live-Distribution PartedMagic 4.8 benötigt um einigermaßen flüssig zu arbeiten, mindestens einen Rechner mit einem Prozessor von 2GHz und einen Hauptspeicher (RAM) von etwa 500MByte. Bei diesem vorgenannten Beispiel dauert die Virensuche auf einer etwa 10 GByte großen Partition etwa 25 – 30 Minuten.

siehe auch: clamscan -h

Verschlüsselung

* * * * *

Kryptographie mit GnuPG - Artikel Nr. 1

Erzeugen eines primären Schlüsselpaares, das aus einem geheimen und einem öffentlichen Schlüssel besteht:

gpg --gen-key

Nun muss die Frage nach der Art des Schlüssels beantwortet werden. Hier kann der Defaultwert 1 mit **[Enter]** übernommen werden, um 2 Schlüssel (privater und öffentlicher) zu erzeugen. Die Frage nach der Schlüssellänge kann mit dem Defaultwert 1024 beantwortet werden (**[Enter]**).

Die Frage nach dem Verfallsdatum des Schlüssels kann auch mit Enter beantwortet werden (Schlüssel verfällt nie). Um eine eindeutige Benutzer-ID zu erzeugen, wird als nächstes nach Name und Vorname gefragt, anschließend nach einem Kommentar und der e-mail Adresse.

Zum Schluss muss noch ein Mantra (ein Passwort) eingegeben werden.

Die hiermit erzeugten Schlüssel sind im Verzeichnis `~/gnupg/` zu finden (`~` ... Home-Verzeichnis).

Editieren eines Schlüssels - zum Beispiel um weitere User-ID's einzutragen, den Fingerabdruck eines fremden Schlüssels zu prüfen (mit **help** wird eine kurze Syntaxhilfe gezeigt), ...

gpg --edit-key ruwela@web.de

Hier ist die e-mail Adresse `ruwela@web.de` durch die eigene e-mail Adresse zu ersetzen, d.h. die e-mail Adresse die beim Generieren des Schlüsselpaares angegeben wurde.

Nach der Eingabe dieses Befehls verändert sich der Eingabeprompt:

Befehl> help

Mit **help** wird eine kleine Hilfe angezeigt (quit ... Menü verlassen; save ... speichern und Menü verlassen; passwd ... Passphrase ändern; expire ...

Verfallsdatum ändern - bringt aber im Nachhinein nicht mehr so viel; addkeys ... Subkeys hinzufügen; etc.).

Exportieren des geheimen Schlüssels in eine Textdatei mit ASCII Werten. Achtung diese darf keinesfalls weitergegeben werden. Eventuell ausdrucken und an einem sicheren Ort verwahren (Bankschließfach).

gpg --export-secret-keys --armor ruwela@web.de > Geheim.txt

Die e-mail Adresse `ruwela@web.de` durch die eigene e-mail Adresse ersetzen, d.h. die e-mail Adresse die beim Generieren des Schlüsselpaares angegeben wurde.

Exportieren des öffentlichen Schlüssels in eine Textdatei mit ASCII Werten. Diese kann per e-mail, über eine Webseite oder über einen Key-Server (z.B. <http://www.keyserver.net/>) ausgetauscht werden.

gpg --export --armor ruwela@web.de > Schluessel.txt

Die e-mail Adresse `ruwela@web.de` durch die eigene e-mail Adresse ersetzen, d.h. die e-mail Adresse die beim Generieren des Schlüsselpaares

angegeben wurde.

Importieren eines öffentlichen Schlüssels:

gpg --import Schluessel.txt

Auflisten aller öffentlichen Schlüssel an Ihrem »Schlüsselbund« (auch Schlüssel von Kommunikationspartnern die importiert wurden):

gpg --list-keys

Die Textdatei text.txt verschlüsseln:

gpg --encrypt text.txt

GnuPG fragt nun nach der User-ID, das ist der eindeutige Name, Kommentar oder am besten die e-mail Adresse des Empfängers. Falls man das Dokument für sich selbst verschlüsselt, einfach die eigene e-mail Adresse angeben. Nun wird die verschlüsselte Datei text.txt.gpg erstellt.

Die für sich selbst bestimmte Datei text.txt.gpg entschlüsseln:

gpg --decrypt text.txt.gpg > text.txt

Das Verzeichnis Briefe mit tar packen und mit gpg für den Benutzer mit der e-mail Adresse mail@ruwela.de verschlüsseln:

tar -c Briefe | gpg -e -r mail@ruwela.de > Briefe.tar.gpg

Das zuvor erstellte und verschlüsselte Paket wieder entpacken:

gpg -d Briefe.tar.gpg | tar -x

Die Textdatei text.txt Signieren:

gpg --clearsign text.txt

GnuPG fragt nun nach dem Mantra (Passwort) und erstellt anschließend die Datei text.txt.asc. Dieser Datei ist eine verschlüsselte Signatur angehängt.

Die Signatur der Textdatei text.txt.asc überprüfen. Um eine Signatur zu überprüfen benötigt man den öffentlichen Schlüssel des Absenders, d.h. vorher ist der öffentliche Schlüssel zu importieren.

gpg --verify text.txt.asc

GnuPG zeigt nun Absender, Erstellungsdatum und Schlüssel-ID an.

Für die Textdatei text.txt eine abgetrennte Signatur erstellen (in einer eigenen Datei):

gpg --detach-sign text.txt

GnuPG fragt nun nach dem Mantra (Passwort) und erstellt anschließend die Datei **text.txt.sig**. Diese Datei ist die Signatur für text.txt.

Die Textdatei text.txt mithilfe der Signatur text.txt.sig überprüfen. Um eine Signatur zu überprüfen benötigt man den öffentlichen Schlüssel des Absenders, d.h. vorher ist der öffentliche Schlüssel zu importieren.

gpg --verify text.txt.sig

GnuPG zeigt nun Absender, Erstellungsdatum und Schlüssel-ID an.

Info: Asymmetrische Verfahren

Ein offensichtlicher Nachteil symmetrischer Verfahren besteht im Austausch des Geheimwortes, der oft ungesichert erfolgt und damit eine potentielle Gefahrenquelle darstellt.

Eine Lösung hierfür bieten die so genannten Public-Key-Verfahren, die zwei Paare von Schlüssel verwenden. Der eine öffentliche Schlüssel dient zur Chiffrierung der Nachricht und nur mit dem zugehörigen privaten Schlüssel lässt sich aus der Nachricht wieder der Klartext gewinnen. Alle Verfahren nach diesem Schema werden als asymmetrisch bezeichnet.

Das Prinzip der Public-Key-Verfahren beruht auf mathematische Einwegfunktionen. Ein oft bemühtes Beispiel ist die Faktorisierung. Angenommen, Sie multiplizieren zwei relativ große Primzahlen miteinander. Die Rechnung dürfte relativ schnell vonstatten gehen. Nehmen Sie jetzt jedoch ein beliebiges Resultat einer Primzahlenmultiplikation her und versuchen ohne Kenntnis der beiden Faktoren dieselbigen zu berechnen, dann müssten Sie trotz Computerhilfe vermutlich recht viel Zeit investieren. Sind die Faktoren nur groß genug, würden Sie an der Faktorisierung letztlich scheitern. Und genau solche Berechnungsvorschriften, wo die »Hinrechnung« einfach, die »Rückrechnung« allerdings schier unmöglich ist, werden zur

Schlüsselerzeugung für asymmetrische Verfahren eingesetzt.

RSA

Das von Rivest, Shamir und Adleman entwickelte RSA-Verfahren ist wohl der bekannteste Vertreter asymmetrischer Verfahren und ein Beispiel der Anwendung der Faktorisierung zur Erzeugung der Schlüssel.

Vorab wird die Schlüssellänge fest gelegt. Je länger dieser ist, desto sicherer ist das Verfahren. Allerdings wird es auch langsamer, weshalb bspw. die Secure Shell 1024 Bit als Voreinstellung vorschlägt (ssh-keygen). Dieser Wert gilt als sicher und stellt bei heutiger Rechengeschwindigkeit noch keine Bremse dar.

Die gewählte Schlüssellänge beeinflusst die Länge der beiden zu erzeugenden Primzahlen. Jede muss mindestens halb so lang wie die Schlüssellänge sein. Die Erzeugung solcher Primzahlen wird per Zufallsgenerator mit anschließendem Primzahltest vorgenommen.

ElGamal

Der nach seinem Erfinder Taher ElGamal benannten Algorithmus verwendet ein anderes zahlentheoretisches Problem, die Berechnung des diskreten Logarithmus modulo einer »großen« Primzahl.

Umfangreiche (auch deutsche) Dokumentationen zu GnuPG findet man auf folgenden Seiten:

<https://www.gnupg.org/>
www.gnupg.org/documentation/
www.gnupg.org/documentation/faqs.html

* * * * *

Das kleine GnuPG Intro - Artikel Nr. 2

Einleitung

Bei dieser kleinen Einführung zur Verwendung des Kryptographie-Tools »Gnu Privacy Guard« - GPG handelt es sich um einen Auszug aus dem »GNU-Handbuch zum Schutze der Privatsphäre«. Ich werde im Folgenden lediglich die wichtigsten Grundfunktionen von GPG erläutern. Für weitere Details verweise ich auf die Dokumentation sowie den FAQs der GPG-Homepage (<https://www.gnupg.org/>).

Das Grundprinzip von GPG

GPG verwendet ein asymmetrisches Verschlüsselungsprinzip. Dabei wird ein komplementäres Schlüsselpaar verwendet. Der zu chiffrierende Text

wird mit einem öffentlichen Schlüssel verschlüsselt. Zum dechiffrieren des Textes verwendet der Empfänger den zugehörigen geheimen Schlüssel. Dieses Prinzip bietet den Vorteil, dass der öffentliche Schlüssel jedem zugänglich gemacht werden kann, verschlüsselte Nachrichten aber nur vom Empfänger gelesen werden können.

(Zum Vergleich: Symmetrische Verschlüsselung verwendet zum Ver- und Entschlüsseln den gleichen Schlüssel. Dabei stellt die sichere Weitergabe des Schlüssels das größte Problem dar.)

Mehr zur Funktionsweise der asymmetrischen Verschlüsselung findet man unter: <http://www.pgpi.org/doc/pgpintro/>

Generieren des ersten Schlüsselpaares

Damit GPG zum Verschlüsseln, Entschlüsseln und Signieren eingesetzt werden kann, benötigt man ein Schlüsselpaar, mit einem geheimen und einen öffentlichen Schlüssel.

Dazu muss der Befehl

gpg --gen-key

aufgerufen werden. (evtl. muss der Befehl zweimal aufgerufen werden, da beim ersten Mal lediglich das Verzeichnis ~/.gnupg angelegt wird.)

Es werden nun diverse Schlüsseleigenschaften erfragt. Es bietet sich an die Default-Einstellungen zu übernehmen. Danach wird eine Benutzer-ID erstellt, z.B.:

Isaac Newton (Mathetutor) isaac@pool.math.tu-berlin.de

Nun wird der User nach einem so genannten »Mantra« (Passwort, Passphrase) gefragt. Dabei sollten folgende Kriterien beachtet werden:

- nicht zu kurz, man kann ohne weiteres ganze Sätze mit Leerzeichen verwenden (Leerzeichen erhöhen die Sicherheit der Passphrase erheblich!)
- enthält Sonderzeichen,
- ist kein Name und
- ist nicht mit Kenntnis persönlicher Daten des Benutzers leicht zu erraten (wie Telefonnummer, Bankleitzahl, Name und Anzahl der Kinder, ...)

Zum Schluss sollte noch dafür gesorgt werden, dass kein Unbefugter Zugriff auf den privaten Schlüssel erhält:

chmod 700 ~/.gnupg

Achtung! Auch wenn das Mantra möglichst sicher gewählt werden sollte, muss man dafür Sorge tragen, es nicht zu vergessen. Es ist ratsam, sich keine neue Passphrase für den Schlüssel auszudenken, da man diese leicht wieder vergessen könnte. Stattdessen sollte man etwas Bekanntes nehmen, an das man sich auch noch nach einem halben Jahr wieder erinnern kann. Aber Vorsicht: Die Passphrase sollte trotzdem für andere nicht leicht erratbar sein. Wenn der Schlüssel ohne »Verfallsdatum« generiert wird und der öffentliche Schlüssel z.B. auf einen PGP/GPG-Server exportiert wird, kann ohne das Mantra keine »Widerrufsurkunde« erzeugt werden.

Arbeiten mit den Schlüsseln

Die erzeugten Schlüssel werden nun in einem »Schlüsselbund« gespeichert. Im Verzeichnis `~/.gnupg/` befinden sich die Dateien `pubring.gpg` (der Schlüsselbund für die öffentlichen Schlüssel) und `secring.gpg` (der Schlüsselbund für die geheimen Schlüssel).

Um einen Austausch von verschlüsselten Nachrichten zu ermöglichen, müssen nun die öffentlichen Schlüssel »ausgetauscht« werden, d.h. man exportiert den eigenen öffentlichen Schlüssel in eine Datei und stellt ihn dann der Allgemeinheit zur Verfügung (z.B. durch Veröffentlichung auf einem der PGP-Server) und importiert die öffentlichen Schlüssel anderer in das eigene öffentliche Schlüsselbund.

Auflisten der Schlüssel eines Schlüsselbundes

Zum Auflisten der Schlüssel verwendet man den Befehl:

gpg --list-keys

Exportieren des eigenen öffentlichen Schlüssels

Den eigenen öffentlichen Schlüssel kann man auf drei verschiedene Arten exportieren:

- als Binärdatei

gpg --output dateiname.ext --export isaac@pool.math.tu-berlin.de

- als ASCII – Datei

gpg --output dateiname.ext --armor --export isaac@pool.math.tu-berlin.de

- als ASCII – Bildschirmanzeige

gpg --armor --export isaac@pool.math.tu-berlin.de

Nun kann der öffentliche Schlüssel anderen Personen übergeben werden (z.B. per Email, WWW, USB-Stick, PGP-Server...).

Importieren eines anderen öffentlichen Schlüssels

Wenn man einen öffentlichen Schlüssel in Form einer Datei erhalten hat oder eine per Email oder WWW erhaltene ASCII-Zeichenkette eines öffentlichen Schlüssels als Textdatei speichert, kann man den Schlüssel zum eigenen Schlüsselbund hinzufügen:

gpg --import dateiname.ext

Signieren von Dokumenten

Digitale Signaturen sind eine Art Siegel, mit der die Authentizität eines empfangenen Dokuments bestätigt wird. Sie ermöglichen zudem eine einwandfreie Zuordnung des Absenders.

Die digitale Signatur ist eine kurze Datensequenz, die durch eine spezielle mathematischen Funktion (Hash-Funktion) mit Hilfe des privaten Schlüssels, des Mantras und der Zeichenfolge des Dokuments erstellt wird. Der Empfänger kann dann mit dem öffentlichen Schlüssel des Absenders die Authentizität des Dokuments überprüfen. Wenn auch nur ein Zeichen nachträglich im Dokument geändert wird, ist die Signatur nicht mehr gültig und GnuPG liefert eine Fehlermeldung zurück.

Eine Datei signiert man mit dem Befehl:

gpg --output datei2.ext --sign datei1.ext

Dabei wird die Datei (datei1.ext) vor dem Signieren komprimiert und im binären Format gespeichert (datei2.ext).

Der Empfänger prüft und dekomprimiert die Datei mit

gpg --output datei3.ext --decrypt datei2.ext

(Dies geht übrigens nur, wenn der öffentliche Schlüssel des Absenders im Schlüsselbund des Empfängers integriert ist!)

Wenn man signierte Texte unkomprimiert speichern will (damit der Empfänger den Inhalt ggf. auch ohne Überprüfung der Signatur lesen kann) verwendet man den Befehl

gpg --output datei2.ext --clearsign datei1.ext

Dabei wird die Signatur des Textes an das Ende des Dokumentes gestellt:

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA1

Hier steht der zu signierende Text.

-----BEGIN PGP SIGNATURE-----

Version: GnuPG v1.0.6 (GNU/Linux)

Comment: For info see <http://www.gnupg.org>

iD8DBQE8endWKOju0A7sC5ARArFaAJ9OnS9rNRu1gI3kAn044SLzUmFMc
ACeLP4f

FpKlJRwbUDE+go6aE0kzHPU=

=h2gB

-----END PGP SIGNATURE-----

Wurde eine Signatur korrekt verifiziert, so erhält man eine Mitteilung der Form:

gpg: Signature made Thu Feb 28 15:08:05 2002 CET using DSA key ID
5CF428D3

gpg: Good signature from "Isaac Newton (Mathetutor) isaac@pool.math.tu-berlin.de"

Widerrufen des eigenen öffentlichen Schlüssels

Wenn der private Schlüssel die erwünschte Sicherheit nicht mehr gewährleistet oder aus einem anderen Grund nicht mehr verwendet werden soll, kann man den Schlüssel mit Hilfe einer »Widerrufsurkunde« ungültig machen.

Es macht durchaus Sinn diese Widerrufsurkunde direkt nach dem Generieren des ersten Schlüsselpaares zu erzeugen, da der häufigste Grund für einen Widerruf die vergessene Passphrase ist (...ohne das Mantra kann **keine** Widerrufsurkunde erzeugt werden, d.h. es ist dann nicht mehr möglich den Schlüssel zu widerrufen, wenn die Urkunde nicht bereits existiert).

Der Befehl zum Erzeugen der Widerrufsurkunde lautet:

gpg --output revoke.asc --gen-revoke mykey
bzw.

gpg --output revoke.asc --armor --gen-revoke mykey

Dabei steht das Wort **mykey** für irgendeinen Teil der Benutzer-ID (z.B. die Email-Adresse).

Der Schlüssel kann nun jederzeit durch das Veröffentlichen der Widerrufsurkunde (engl. revoke certificate) z.B. auf einem PGP-Server oder dem Versenden der Urkunde per Mail ungültig gemacht werden.

Die Widerrufsurkunde sollte allerdings gut vor unbefugtem Zugriff

geschützt werden!

GPA - Die graphische Oberfläche für GnuPG

Der »Gnu Privacy Assistant (GPA)« ist ein graphisches Interface für GPG. Das Programm erleichtert die oben aufgeführten Funktionen erheblich, da man sich keine langen Kommandozeilen merken muss. Das Programm ist im UNIX-Pool installiert und kann mit dem Befehl **gpa** aufgerufen werden.

Man kann GPA für den privaten PC auf den folgenden WWW-Seiten erhalten.

GnuPG-Homepage ... <https://www.gnupg.org>

Die Homepage des Bundesministeriums für Wirtschaft und Technologie und des Bundesinnenministeriums zum Thema Sicherheit im Netz. Seit Mitte des Jahres 2000 unterstützt die Bundesregierung viele Open-Source Projekte, u.a. das »Gnu Privacy Project«.

Links

<https://www.gnupg.org/> ... GnuPG-Homepage (engl.)

<http://www.pgpi.org/> ... die internationale PGP-Homepage

<http://web.mit.edu/~prz> ... Homepage von Phil R. Zimmermann, dem Erfinder von PGP

* * * * *

Kurzreferenz der gängigsten gpg-Befehle - Artikel Nr. 3

Erzeugen, Export, Import, Widerruf

gpg --gen-key ... Regeln für das Mantra: - nicht zu kurz - enthält Sonderzeichen - kein Name - nicht leicht zu erraten

gpg --export [-a|--armor] [UID] Schlüssel[-bund] ... exportieren

gpg --send-keys [--keyserver servername] [UID] ... Schlüssel zum Schlüsselserver senden (keinen kompletten Schlüsselbund senden!)

gpg --recv-key [KeyID] ... Schlüssel vom Schlüsselserver holen (in Schlüsselbund aufnehmen)

gpg --import [Datei] Schlüssel [aus Datei] ... importieren eines öffentlichen Schlüssels, d.h. den Schlüssel in den eigenen öffentlichen

Schlüsselbund aufnehmen

gpg --gen-revoke ... Widerruf-Zertifikat - benötigt privaten Schlüssel mit Mantra!

Schlüsselbund verwalten

gpg --list-keys ... alle Schlüssel des öffentlichen Schlüsselbundes anzeigen

gpg --list-sigs ... alle Schlüssel des öffentlichen Schlüsselbundes mit Signaturen anzeigen

gpg --fingerprint ... Schlüssel mit Fingerabdrücken anzeigen bzw.

gpg --fingerprint hansmueller@web.de ... Fingerabdruck des öffentlichen Schlüssels von Hans Müller zeigen, dessen UID die Zeichenkette hansmueller@web.de enthält.

gpg --list-secret-keys ... alle Schlüssel des privaten Schlüsselbundes anzeigen

gpg --delete-key [UID] bzw. **gpg --delete-secret-key [UID]** ... Schlüssel aus dem entsprechenden öffentlichen oder privaten Schlüsselbund löschen

gpg --edit-key [UID] ... Schlüssel editieren (Mantra; Verfallsdatum; signieren: **gpg --sign**; ownertrust)

Nach der Eingabe dieses Befehls verändert sich der Eingabeprompt:

Befehl> help

Mit **help** wird eine kleine Hilfe angezeigt (quit ... Menü verlassen; save ... speichern und Menü verlassen; passwd ... Passphrase ändern; expire ... Verfallsdatum ändern - bringt aber im Nachhinein nicht mehr soviel; addkeys ... Subkeys hinzufügen; etc.).

Verschlüsseln und Entschlüsseln

gpg --encrypt Empfänger [Datei] ... Verschlüsseln einer Datei (sinnvoll: auch signieren, **gpg --sign**)

gpg --encrypt --output [Datei.gpg] --recipient Empfänger1 --recipient Empfänger2 --recipient Empfänger3 --recipient Absender [Datei] ... Verschlüsselung einer Datei für mehrere Nutzer - ist beliebig erweiterbar; der Absender sollte mit aufgenommen werden, damit dieser selbst noch in

der Lage ist die verschlüsselte Datei zu entschlüsseln - ansonsten steht er vor verschlossene Türen

gpg [--decrypt] [Datei] ... Entschlüsseln einer Datei

Signieren

gpg --sign [Datei] ... Datei mit dem privaten Schlüssel signieren und komprimieren

Hinweis: Schlüssel können ebenfalls wie jede anderen Datei signiert werden, um durch die digitale »Unterschrift« deren Echtheit und Unversehrtheit zu gewährleisten. Sind Sie sich absolut sicher, dass ein importierter Schlüssel wirklich demjenigen zugeordnet ist, der als Besitzer genannt wird, können Sie Ihr Vertrauen in die Echtheit des Schlüssels durch Ihre Signatur zum Ausdruck bringen.

gpg --clearsign [Datei] ... Datei lesbar belassen

gpg --detach-sign [--armor] [Datei] ... Unterschrift in separater Datei (für Binärdateien)

gpg [-u Sender] [-r Empfänger] [--armor] --sign --encrypt [Datei] ... gleichzeitig signieren und verschlüsseln

gpg [--verify] [Datei] ... Signatur der unverschlüsselten Datei mit öffentlichem Schlüssel prüfen

Löschen

gpg --delete-keys hanspeter@tux.de ... Löschen des öffentlichen Schlüssels mit dem ID-Bestandteil z.B. mit der e-mail Adresse
»hanspeter@tux.de«

gpg --delete-secret-keys hanspeter@tux.de ... Löschen des privaten Schlüssels mit dem ID-Bestandteil z.B. mit der e-mail Adresse
»hanspeter@tux.de«

Quellen

<https://www.gnupg.org>
man gpg

* * * * *

Bekanntmachung des Schlüssels und Widerruf – Artikel Nr. 4

Wenn Sie Ihren eigenen Schlüssel und ein Revocation-Zertifikat generiert haben, dann können Sie Ihren öffentlichen Schlüssel per eMail auf einen Keyserver hochladen.

Siehe auch: aktuelle Liste von Keyserver erhalten aus dem INTERNET

Extrahieren Sie zuerst Ihren öffentlichen Schlüssel. Das geht mit der Option **-a --export <USERID>**. Hier müssen Sie wiederum für <USERID> einen Teil der gewünschten User-ID einfügen. Die Ausgabe dieses Kommandos können Sie z.B. auf das Programm **mail** umleiten, um so den Schlüssel direkt an den Keyserver zu mailen. Es besteht auch die Möglichkeit, den Schlüssel in eine Datei zu speichern und diese dann im Mailclient einzufügen. Die Email muss einfach im Betreff das Wort **add** und im **Body** den **Schlüsselblock** enthalten. Der Bequemlichkeit halber wähle ich immer den direkten Weg:

```
gpg -a --export <USERID> | mail -s "add" pgp-public-keys@keys.ch.pgp.net
```

Kurze Zeit später kriegen Sie eine Bestätigung, dass der Schlüssel hinzugefügt wurde. Wenn Sie später einmal eine neue User-ID hinzufügen oder jemand Ihren Schlüssel unterschreibt, dann können Sie diesen einfach erneut auf den Keyserver hochladen. Er merkt von alleine, dass nur eine User-ID oder eine Unterschrift hinzugekommen ist und datiert den bereits vorhandenen Schlüssel einfach neu.

Ihren Schlüssel können Sie zwar nicht wieder vom Keyserver löschen, aber Sie können ihn für ungültig erklären. Dazu dient das Revocation-Zertifikat, dass Sie vorher erstellt haben. Wenn Sie es anwenden wollen, dann können Sie es genauso auf den Keyserver hochladen, wie den öffentlichen Schlüssel. Falls Sie es nur ausgedruckt haben, dann müssen Sie es zuerst wieder in ein für Computer lesbares Format umwandeln (z.B. per OCR).

In beiden Fällen müssen Sie das Zertifikat per Email an den Keyserver schicken. Dies geht entweder per Cut & Paste oder direkt über das Programm **mail**. Wichtig ist nur, dass im **Subject** das Wort **add** und im **Body** das **Zertifikat** ist. Achtung: Diese Prozedur unumkehrbar!

```
cat gpg-revoke.txt | mail -s "add" pgp-public-keys@keys.ch.pgp.net
```

Fremde Schlüssel holen

So einfach wie das Hinzufügen eines Schlüssel zu dem Datenbestand der Keyserver ist, so gestaltet sich auch das Holen eines öffentlichen Schlüssels. Sie müssen im **Subject** der eMail das Wort **get** gefolgt von der **eMail-Adresse des Besitzers** des öffentlichen Schlüssels eingeben und das ganze dann an einen Keyserver in Ihrer Nähe schicken. Wenn Sie z.B. den öffentlichen Schlüssel vom thomas holen wollen (Email-Adresse ist thomas@trash.de), dann würde der Betreff **get thomas@trash.de** lauten. Dies können Sie auch über Ihren Mail-Client oder das Programm **mail** erledigen;

mail -s "get thomas@trash.de" pgp-public-keys@keys.ch.gpg.net

Kurze Zeit später erhalten Sie eine Antwort vom Keyserver. Diese enthält ein Attachment, in dem entweder der öffentliche Schlüssel oder eine Fehlermeldung, dass der Schlüssel nicht gefunden wurde, enthalten ist. Wenn der Public-Key drin ist, dann kann man ihn von seinem Mailclient in eine Datei speichern lassen und diese dann gpg übergeben, der den Schlüssel schließlich in den Schlüsselbund aufnimmt.

cat pgpkey.txt | gpg --import

Überprüfung eines fremden Schlüssels

Wenn ein Schlüssel einmal importiert ist, sollte er auf Authentizität überprüft werden. GnuPG arbeitet mit einem wirksamen und flexiblen Vertrauensmodell, bei dem Sie nicht jeden Schlüssel persönlich zu authentifizieren brauchen, den Sie importieren.

Einige Schlüssel können dies jedoch erfordern. Ein Schlüssel wird dadurch authentifiziert, dass Sie den Fingerabdruck des Schlüssels überprüfen und dann den Schlüssel unterschreiben, um seine Gültigkeit zu bestätigen. Der Fingerabdruck eines Schlüssels kann mit der Befehlszeilen-Option **--fingerprint** geprüft werden, um aber den Schlüssel zu bestätigen, muss die Option **--edit-key** verwendet werden.

gpg --edit-key Borgert

gpg (GnuPG) 1.0.4; Copyright (C) 2000 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

pub 1024R/9B668109 erstellt: 1998-06-28 verfällt: niemals Vertrauen: -/q
(1) W. Borgert <debacle@knorke.in-berlin.de>

Befehl> **fpr**

pub 1024R/9B668109 1998-06-28 W. Borgert <debacle@knorke.in-berlin.de>
Fingerabdruck: 6F 74 32 AB 53 DD 09 F1 3B 69 E6 3F 79 8A 70 53

Um den Fingerabdruck zu überprüfen, müssen Sie den Eigentümer des Schlüssels kontaktieren und die Fingerabdrücke vergleichen. Sie können persönlich oder per Telefon mit ihm sprechen oder auf beliebigem anderen Wege kommunizieren, solange nur garantiert ist, dass es sich um den rechtmäßigen Eigentümer handelt. Stimmen beide Fingerabdrücke überein, dann können Sie sicher sein, dass Sie eine echte Kopie des öffentlichen Schlüssels haben.

Nach dem Prüfen des Fingerabdrucks können Sie den Schlüssel unterschreiben, um ihn zu authentifizieren. Da die Schlüsselüberprüfung ein Schwachpunkt in der Kryptographie mit öffentlichem Schlüssel ist, sollten Sie äußerste Sorgfalt walten lassen und den Fingerabdruck eines Schlüssels immer gemeinsam mit dem Eigentümer prüfen, bevor Sie den Schlüssel unterschreiben.

* * * * *

Symmetrische Verschlüsselung mit gpg

Die asymmetrische Ver- und Entschlüsselung ist nur möglich auf Rechner, auf denen der private Schlüssel und der öffentliche Schlüssel gespeichert oder verfügbar ist. Ist dieser Rechner nicht verfügbar, kann man sich mit der symmetrischen Verschlüsselung behelfen, die auf jeden Linux-Rechner mit gpg wieder entschlüsselt werden kann.

Falls aber auch kein Linux-Rechner verfügbar ist, kann man sich z.B. mit Knoppix behelfen. Knoppix ist eine komplett von CD lauffähige Zusammenstellung von GNU/Linux-Software mit automatischer Hardwareerkennung. Es ist keinerlei Installation auf Festplatte notwendig. Knoppix ist Open Source und frei erhältlich (GPL).

Bei einer symmetrischen Verschlüsselung ist kein öffentlicher oder privater Schlüssel vonnöten. Die **Verschlüsselung** bzw. die **Entschlüsselung** erfolgt ausschließlich mit **ein und demselben Passwort**, d.h. alle die die verschlüsselte Datei lesen wollen, brauchen auch dieses Passwort.

gpg --output datei2 --symmetric datei1 ... symmetrische Verschlüsselung der datei1 mit dem CAST5-Verfahren (Default-Wert), andere Verfahren müssen explizit über die Option **--cipher-algo** angegeben werden (z.B.

3DES, BLOWFISH, AES256, TWOFISH; **siehe auch:** gpg --help)

gpg --output datei2 --symmetric --cipher-algo BLOWFISH datei1 ...

Bei einer symmetrischen Verschlüsselung mit dem BLOWFISH-Verfahren, sollte ein möglichst langes Passwort verwendet werden.

gpg --output datei1 --decrypt datei2 ... Entschlüsselung der datei2, die Angabe des Verschlüsselungsverfahrens ist hier nicht notwendig

3DES: TripleDES (3DES) ist eine Weiterentwicklung von DES (Schlüssellänge 51 Bit). 3DES wendet den DES-Algorithmus dreifach an, mit einem auf 168 Bit erweiterten Schlüssel.

AES: Der Advanced Encryption Standards (AES) resultierte aus einer vom US National Institute for Standards and Technology (NIST) initiierten Ausschreibung (1977) zur Ablösung von DES als amerikanischen Standard. 2001 wurde das auf einen Kryptologie-Algorithmus von Rijndael zurückgehende Verfahren offiziell als Nachfolger von DES bestätigt. Im Unterschied zu DES unterstützt das neue Verfahren wahlweise Schlüssellängen von 128, 192 oder 256 Bit (AES, AES192, AES256), so dass Brute-Force-Angriffe auch auf lange Sicht unwahrscheinlich werden. Außerdem unterliegt das Verfahren keinerlei Lizenzbestimmungen und kann von jedem implementiert werden.

CAST5: Schlüssellänge 128 Bit

BLOWFISH: Schlüssellänge 128 Bit

TWOFISH: Schlüssellänge 256 Bit

Hinweis: Wem Programme mit grafischer Oberfläche mehr liegen, kann das KDE-Programm **KGpg** verwenden. Bei einigen Distributionen muss es erst installiert werden. Mit [Strg] + [F2] das Schnellstartfenster aufrufen und dort **KGpg** eingeben und bestätigen.

* * * * *

Schnellanleitung Ver- und Entschlüsselung – Artikel Nr. 5

1. asymmetrische Verschlüsselung

Es wird an dieser Stelle davon ausgegangen, dass bereits ein privater und öffentlicher Schlüssel erzeugt wurde und das mindestens ein fremder öffentlicher Schlüssel von einem **Empfänger** der verschlüsselten Nachricht (Email-Adresse: **susischoen@domain.net**) importiert wurde. Die Email-Adresse des **Absenders** lautet **hansotto@domain.de**. Analog gelten diese

Voraussetzungen auch für den Empfänger der Nachricht - hier Susi Schön.

Verschlüsselung

- erstellen einer Datei, die verschlüsselt werden soll - Dateiname z.B. geheim.txt
- **gpg --encrypt --output datei.gpg --recipient susischoen@domain.net geheim.txt**
- nun kann die verschlüsselte Nachricht (**datei.gpg**) z.B. als Email-Anhang an Susi Schön gesendet werden

Entschlüsselung

- Susi Schön hat die Email mit dem Anhang erhalten und will nun die Nachricht entschlüsseln
- **gpg --output geheimertext.txt --decrypt datei.gpg**
- nun wird sie nach dem Passwort **ihres** privaten Schlüssels gefragt
- die entschlüsselte Nachricht wird in die Datei **geheimertext.txt** exportiert

2. symmetrische Verschlüsselung

Voraussetzungen sind hier das der Absender und der Empfänger über einen Rechner mit Linux verfügen, auf dem GPG installiert ist - dies ist normalerweise der Standard. Die symmetrische Verschlüsselung benötigt keinen privaten und öffentlichen Schlüssel. Absender und Empfänger benutzen ein und dasselbe Passwort zum verschlüsseln und entschlüsseln - hier im Beispiel wird das Passwort **schleichender Tiger 2046** benutzt.

Verschlüsselung

- erstellen einer Datei, die verschlüsselt werden soll - Dateiname z.B. geheim.txt
- **gpg --output datei.gpg --symmetric --cipher-algo BLOWFISH geheim.txt**
--force-mdc aktiviert den »Modification Detection Code« für den Integritätstest (nur für BLOWFISH); bei der Entschlüsselung kann dadurch --no-mdc-warning entfallen

Hinweis: Unter den unter gpg verfügbaren Verschlüsselungs-Algorithmen ist BLOWFISH der schnellste, TWOFISH der wahrscheinlich sicherste und AES256 der am meisten

verwendete Verschlüsselungs-Algorithmus.

- nun kann die verschlüsselte Nachricht (**datei.gpg**) z.B. als Email-Anhang an Susi Schön gesendet werden

Entschlüsselung

- Susi Schön hat die Email mit dem Anhang erhalten und will nun die Nachricht entschlüsseln
- **gpg --output geheimertext.txt --no-mdc-warning --decrypt datei.gpg**
- nun wird sie nach dem Passwort gefragt, nach der Eingabe von **schleichender Tiger 2046** wird die Datei entschlüsselt und als **geheimertext.txt** exportiert; --no-mdc-warning ist optional, unterdrückt nur eine Warnung

* * * * *

vi

vi ist ein Editor der auf der Kommandozeilenebene bedient wird. Die Bedienung von vi ist am Anfang sehr gewöhnungsbedürftig, aber auf **Rettungssystemen / »Rescue«-Systemen** (DVD, CD-ROM: meist 1. oder 2. Installationsmedium) ist **vi** meist der einzige verfügbare Editor.

Der Editor vi kennt drei Modi: den Befehlsmodus – jeder Tastendruck wird als Teil eines Befehls interpretiert; den Einfügemodus – Tastatureingaben werden als Text interpretiert; den komplexen Befehlsmodus – Befehle werden in der letzten Zeile eingegeben, sie werden durch einen Doppelpunkt [:] eingeleitet.

vi ... startet den Editor vi; danach **e: [datei]** eingeben, um die benannte Datei zu öffnen

vi [datei] ... öffnet die benannte Datei im Editor (z.B. vi /etc/shadow)

i oder **[Einf]** ... wechselt in den Einfügemodus (insert)

[Esc] ... wechselt vom Einfügemodus in den Befehlsmodus; wird danach ein Doppelpunkt eingegeben, so wird in den komplexen Befehlsmodus gewechselt

o ... es wird eine neue Zeile eingefügt, nach dem aktuellen Cursor (nur im Befehlsmodus)

x ... löscht das aktuelle Zeichen (nur im Befehlsmodus)

/<Suchstring> ... sucht den betreffenden String, mit **[n]** wird der nächste Suchtreffer angezeigt

dd ... löscht die aktuelle Zeile (nur im Befehlsmodus)

:d ... löscht die Zeile in der der Cursor sich gerade befindet (nur im komplexen Befehlsmodus)
:r [datei] ... fügt den Inhalt einer Datei nach dem aktuellen Stand des Cursors ein (nur im komplexen Befehlsmodus)
u ... letzte Änderung widerrufen, Undo-Funktion (nur im Befehlsmodus)
:u ... letzte Änderung widerrufen, Undo-Funktion (nur im komplexen Befehlsmodus)
[Strg] + [u] ... letzte Änderung widerrufen, Undo-Funktion (innerhalb des Einfügemodus)
:q! ... vi verlassen, ohne die Änderungen zu speichern (nur im komplexen Befehlsmodus)
:w [datei] ... aktueller Inhalt in einer anderen Datei abspeichern (nur im komplexen Befehlsmodus)
:x ... speichert die aktuelle Datei und verlässt den Editor (nur im komplexen Befehlsmodus)
:wq! ... Änderungen speichern und vi verlassen (nur im komplexen Befehlsmodus)

Allen Befehlen kann eine Zahl vorangestellt werden, die angibt, auf wie viele Objekte sich der folgende Befehl beziehen soll. Durch Angabe von z.B. **10x** erreicht man das löschen von 10 Zeichen ab der Cursorposition, **20dd** löscht 20 Zeilen.

siehe auch: man vi bzw. man vim

vobcopy

vobcopy kopiert von einer Video-DVD vob-Dateien (Video-Object-Dateien) auf die Festplatte. Die kopierten Dateien erhalten den gleichen Dateinamen, wie die vob-Dateien auf der DVD. vobcopy prüft die Festplatte (Ziel-Partition) auf ausreichend vorhandenen freien Festplattenplatz und vergleicht die Größe der kopierten Dateien mit der Größe auf der DVD. Falls die kopierten Dateien sich in der Größe mit den Dateien auf der DVD unterscheiden, behalten die Dateien ihre temporäre Endung .partial.

Hinweis: vobcopy befindet sich teilweise noch in einem frühen Entwicklungsstadium (Stand: 2014).

vobcopy [-i Eingabeverzeichnis] [-n Titel-Nummer] [-M] [-l] [-o Ausgabeverzeichnis]

-i, --input-dir EINGABE-VERZEICHNIS ... gibt das Eingabeverzeichnis an - das Verzeichnis, auf dem die DVD eingehangen, gemountet ist

-l, --large-file ... schreibt alle Daten in eine große Datei (> 2GB)

-M, --longest ... wählt den Titel mit der längsten Spielzeit; Bei manchen

DVDs wird der Hauptfilm besser gefunden als mit der Methode der meisten Kapitel (vobcopy .. ohne Optionen), bei manchen schlechter.

-m, --mirror ... spiegelt die gesamte DVD auf die Festplatte; Es wird ein Unterverzeichnis mit dem DVD-Namen erzeugt und die gesamte Dateistruktur des Videoteils wird darunter kopiert.

-n, --title-number TITEL-NUMMER ... gibt an, welcher Titel kopiert werden soll; Auf der DVD heißt der erste Titel vts_01_x.vob (-n 1, meist der Hauptfilm).

-o, --output-dir AUSGABE-VERZEICHNIS ... gibt das Ausgabeverzeichnis für die Kopien an

-I, --info ... gibt Informationen über die Titel, Kapitel und Teile der DVD aus

Beispiele:

vobcopy --info ... Informationen über die eingelegte DVD ausgeben

vobcopy --title-number 5 ... kopiert den 5. Titel (VTS_05_1.VOB) auf der DVD in das aktuelle Verzeichnis

vobcopy ... Beim Aufruf ohne jegliche Optionen wird der Titel mit den meisten Kapiteln (i.d.R. der Hauptfilm) in mehreren Dateien mit einer Größe von jeweils 2 GByte in das aktuelle Verzeichnis kopiert.

vobcopy --large-file ... der Hauptfilm (Titel mit dem meisten Kapiteln) wird in eine einzige große Datei kopiert

vobcopy --longest --large-file ... der Hauptfilm (Titel mit der längsten Spielzeit) wird in eine einzige große Datei kopiert

siehe auch: man vobcopy, mplayer, avconv, dd, mount

W

W3M - ein textbasierter Browser

w3m -dump http://192.168.1.5/tux3/agb.htm

Zeigt die angegebene HTML-Seite vollständig an. Ohne die Option -dump wird die HTML-Seite Seitenweise angezeigt.

w3m -dump_source http://192.168.1.5/tux3/agb.htm >

/home/<Benutzer>/quelltext ... Speichert den Quelltext der angegebenen HTML-Seite in der Datei quelltext.

ls -l | w3m ... zeigt das Ergebnis im W3M-Browser an; mit [q] wird W3M beendet

Konvertierung:

Konvertierung einer HTML-Seite zu normalen Text mit W3M

w3m -dump <Dateiname.html> > <Dateiname.txt>

bzw.

w3m -dump <Dateiname.html> | less

siehe auch: lynx

Wave

siehe auch: Sound

w

Mit dem Kommando »w« können Sie feststellen, wer auf dem System eingeloggt ist und was er tut. Sollten sich Benutzer von anderen Systemen remote eingeloggt haben, dann kann man mit der Option **-f** anzeigen lassen, von welchem Rechner aus diese die Verbindung aufgebaut haben.

wc

Wörter, Zeichen bzw. Zeilen zählen.

-m ... Zeichenanzahl ausgeben

-l ... Zeilenanzahl ausgeben

-w ... Wortanzahl ausgeben

Beispiele:

ls *.j{p,pe}g | wc -w ... Alle JPEG-Dateien in einem Verzeichnis zählen.

***** ... beliebige Zeichenkette

? ... genau ein beliebiges Zeichen

[abc] ... genau eines der genannten Zeichen
[^abc] ... genau ein nicht genanntes Zeichen oder **[! abc]**
{abc,bcdf} ... genau eine der Zeichenketten

ls | wc -w ... Alle Einträge eines Verzeichnisses - Dateien und Verzeichnisse - zählen.

ls -R | wc -w ... Alle Einträge eines Verzeichnisses und seiner Unterverzeichnisse - Dateien und Verzeichnisse - zählen.

siehe auch: `wc --help`, `ls`, Umleitung von Befehle, Anhang: Einführung in die Shellprogrammierung

wget

wget ist ein praktisches Tool, um Dateien aus dem Web zu laden. Sogar abgebrochene Downloads können wieder aufgenommen werden. Es werden Dateien und Seiten sowohl per http, als auch per ftp geladen.

Mit wget können nicht nur einzelne Dateien gespeichert, sondern auch ganze Seiten gespiegelt werden. Die dafür zu verwendende Option ist `-r`. Damit wird bei der angegebenen Seite rekursiv den Links gefolgt. Die Standardeinstellung für `-r` folgt fünf Ebenen von Links.

Dabei wird nicht unterschieden, ob die Seiten vom gleichen Server stammen, oder nicht.

Bei FTP-Adressen werden dementsprechend fünf Unterverzeichnisebenen heruntergeladen.

Für das Spiegeln von Webseiten geben Sie folgenden Befehl ein:

wget -r -l 2 --quota=2m www.selflinux.org

Speichert alle Dateien (rekursiv), die über eine andere Datei verlinkt (Linktiefe 2) sind, im Verzeichnis `www.selflinux.org` (max. 2 MByte). Die gefundene Verzeichnisstruktur wird dabei übernommen.

siehe auch: Sitecopy, `wget --help`

whereis

Findet den Pfad des Verzeichnisses in dem das angegebene Programm gespeichert ist.

`whereis <Programmname>`

siehe auch: man which

whatis

Innerhalb jeder Manualseite ist eine Kurzbeschreibung vorhanden. **whatis** sucht **Schlüsselwort** in den Kurzbeschreibungen der Indexdatenbank.

whatis <Schlüsselwort>

Falls es eine solche nicht im Manualpfad gibt, durchsucht es die **whatis**-Datenbank nach **Schlüsselwort**. **whatis** kann Schlüsselwörter suchen, die Wildcards oder reguläre Ausdrücke enthalten.

whatis -w <*begriff*> ... der Suchbegriff enthält Wildcards

siehe auch: apropos, which

which

Gibt den absoluten Pfad eines Programms aus. Das Programm **which** ist besonders innerhalb von Shellskripten nützlich.

which <Schlüsselwort>

which ls ... gibt den absoluten Pfad des Programms **ls** aus: /bin/ls

siehe auch: man which, **whatis**, apropos

whois

Mit dem Befehl **whois** können Informationen aus dem Internet zu einem Domain-Namen oder einer IP-Adresse direkt von den Datenbanken der Registrierungsstellen (NIC) abgerufen werden. Mit **whois** können auch Verbindungen untersucht werden, bei denen sie nicht wissen, welches Programm sie hergestellt hat und wer sich hinter der Gegenstelle verbirgt.

NIC (Network Information Centre) ... Ein Network Information Center (NIC) oder eine Domain Name Registry (Registrierungsstelle) verwaltet eine oder mehrere Top-Level-Domains im Domain Name System. Zu den Aufgaben gehört der Betrieb der Nameserver, die Verwaltung des Namensraums und der Betrieb der Whois-Server mit Kontaktdaten der Domaininhaber.

whois [Optionen] <Domain-Name bzw. IP-Adresse>

-h Registrierungsstelle ... Host, welcher die Identifikationsinformation in

seiner Datenbank bereithält

-p port ... anstelle des Standard-Ports (43), mit dem angegebenen Port verbinden

-a ... alle Datenbanken durchsuchen

Beispiele:

whois pcwelt.de ... die Vorgabe-Registrierungsstelle und den Standardport für die Anfrage benutzen

whois 62.146.91.235 ... die Vorgabe-Registrierungsstelle und den Standardport für die Anfrage benutzen

whois -h whois.networksolutions.com yahoo.com ... die Registrierungsstelle whois.networksolutions.com ist für internationale Domains und IP-Adressen zuständig (whois-Vorgabe)

whois -h whois.denic.net pcwelt.de ... die Registrierungsstelle whois.denic.net ist für deutsche Domains und IP-Adressen zuständig

WHOIS-Server:

whois.networksolutions.com ... internationale Domains

whois.afiliat.net ... INFO-Domains (.info)

whois.dotasia.net ... ASIA-Domains (.asia)

whois.internic.net ... COM-Domains (.com)

whois.eu ... EU-Domains (.eu)

whois.internic.net ... NET-Domains (.net)

whois.pir.org ... ORG-Domains (.org)

whois.denic.net bzw. whois.denic.de ... Deutschland (de)

whois.nic.at ... Österreich (at)

whois.nic.ch ... Schweiz (ch)

whois.nic.fr ... Frankreich (fr)

whois.cnnic.net.cn bzw. whois.centralnic.com ... China (cn, cn.com)

whois.nic.or.kr ... Korea (kr)

whois.tcinet.ru ... Russland (ru)

whois.nic.uk bzw. whois.centralnic.com ... Großbritannien (uk, uk.com)

[...]

siehe auch: man whois, www.nirsoft.net/whois_servers_list.html

whoami

Gibt den aktuellen Benutzernamen aus. Dies ist eigentlich nur interessant, wenn man den Benutzernamen an der Kommandozeile - z.B. mit su - häufiger wechselt oder wenn einige Aktionen in Shellskripts nur von bestimmten Benutzern ausgeführt werden sollen.

when

Das Perl-Programm when ist ein schlanker Terminkalender der ohne Abhängigkeiten – wie Webserver oder Datenbanken – auskommt. Der Kalender when kann somit auch über eine SSH-Verbindung genutzt werden oder auf USB-Stick gespeichert werden. Perl ist i.d.R. auf jeden Unix- und Linux-Rechner standardmäßig installiert.

Beim 1. Aufruf von when müssen sie ihren Standardeditor angeben – z.B. mcedit (Bestandteil des Paketes mc).

Standardmäßig zeigt when die Termine der nächsten 2 Wochen an. Sämtliche Termine und die Konfigurationseinstellungen speichert when in einem versteckten Unterverzeichnis des Home-Verzeichnisses.

Mit wiederkehrende Ereignisse – wie Geburtstage – kann when ebenfalls umgehen (siehe: Beispiel und man when).

when ... Termine der nächsten 2 Wochen anzeigen

when w ... Termine der aktuellen Woche anzeigen

when m ... Termine der nächsten 30 Tage anzeigen

when y ... Termine eines Jahres anzeigen

when c ... Kalender für 3 Monate anzeigen

when cm ... Kalender für 3 Monate und die Termine eines Monats anzeigen

when --past=0 --future=365 c ... Jahreskalender ohne die Termine anzeigen

when e ... Editor aufrufen

Syntax der Termineinträge:

Jahr Monat Tag, beliebiger Text - Uhrzeit

Beispieleinträge im Kalender:

* Jan 01, Neujahr - Feiertag

m=Mar & b=1 & w=Son, Beginn der Sommerzeit

w=Fr & e=2, Karfreitag - Feiertag

e=0, Ostersonntag - Feiertag

e=0-1, Ostermontag - Feiertag

w=Don & e=0-39, Christi Himmelfahrt - Feiertag

w=Son & e=0-49, Pfingstsonntag - Feiertag

w=Mon & e=0-50, Pfingstmontag - Feiertag

* Mai 01, Maifeiertag

* Okt 03, Tag der deutschen Einheit – Feiertag

m=Okt & b=1 & w=Son, Ende der Sommerzeit

* Okt 31, Reformationstag - Feiertag in Sachsen

m=Nov & w=Mit & b=2, Buß- und Bettag - Feiertag in Sachsen

m=Nov & d>26 & w=Son, 1. Advent

m=Dez & d<4 & w=Son, 1. Advent

m=Dez & d>3 & d<11 & w=Son, 2. Advent

m=Dez & d>10 & d<18 & w=Son, 3. Advent

m=Dez & d>17 & d<25 & w=Son, 4. Advent

* Dez 24, Heiligabend - Feiertag

* Dez 25, 1. Weihnachtstag - Feiertag

* Dez 26, 2. Weihnachtstag - Feiertag

1920* Aug 29, Charlie Parker wird \a - geboren \y

* Feb 06, Geburtstag von Sonja

2014 Feb 01, Hallo - wichtiger Termin Nr.1 um 14:30 Uhr

Die Reihenfolge der Termine spielt keine Rolle, da when sie bei der Ausgabe nach Tagen sortiert. Bereits abgelaufene Termine blendet when aus.

Beispiel – Datei: /home/<Benutzername>/.when/preferences

calendar = /home/<Benutzername>/.when/calendar

editor = mcedit

ampm = 0

Anmerkung:

Die Navigation bei etwas längeren Bildschirmausgaben erfolgt entweder über die Leertaste oder den Pfeiltasten.

Eine nicht vollständig abgeschlossene Bildschirmausgabe, wird über die Taste [Q] beendet.

Bei einigen Linux-Distributionen hat when Probleme mit den landestypischen Sonderzeichen (statt **März** besser **Mar** oder **March** verwenden).

siehe auch: man when

Wine

Wine ist ein »Windows-Emulator«, mit dem es möglich ist einigen

Windowsprogrammen auch unter Linux Leben einzuhauchen.

Einfache Programme die nur aus einer Programmdatei (.exe) bestehen oder nur DLL-Dateien bzw. andere Hilfsdateien nutzen die sich im selben Stammverzeichnis wie die Programmdatei befinden, können mit Wine meistens recht unkompliziert zum Laufen gebracht werden.

Andere komplexere Programme die viele Dateien in mehreren Systemordnern speichern und die sich massiv in der Registry von Windows breit machen, haben mit Wine in der Regel schlechte Karten auch unter Linux zu laufen.

Mit **cd** in das Verzeichnis mit dem Windows-Programm wechseln und mit

wine programm.exe

das Programm aufrufen.

siehe auch: man wine



XAMPP für Linux

Allgemeiner Hinweis: Zwei gleichartige Server sollten niemals gleichzeitig aktiv sein.

Beispiel: Es wurde der Apache-Webserver über die Paketquellen installiert, gleichzeitig wurde aber auch das Paket von XAMPP installiert. In dem XAMPP-Paket ist ebenfalls ein Webserver enthalten. Bevor XAMPP über »sudo /opt/lampp/lampp start« gestartet wird, muss der andere Webserver mittels »rcapache stop« (OpenSuse) oder »sudo apachectl stop« (Debian, Ubuntu) angehalten werden. Dies gilt analog auch für die FTP-, Mail-, MySQL- und Samba-Server.

Der gleichzeitige Betrieb von zwei gleichwertigen Servern ist im Einzelfall zwar möglich, dies bleibt aber immer ein Experiment und ist für den normalen Betrieb nicht zu empfehlen.

Das Serverpaket XAMPP bringt all das mit, was des Webmasters Herz begehrt: den Webserver Apache, das PHP- und Perl-Modul, den FTP-Server ProFTPD und die Datenbank MySQL. Fertig konfiguriert und speziell auf Anfänger zugeschnitten. Zur Installation genügt ein Kommando. Das Serverpaket kann man von der Webseite <http://www.apachefriends.org> herunterladen und anschließend installieren.

A. Installation aus TAR-GZ-Archiven:

Mit dem Befehl **tar -xvzf xampp-linux-1.7.3.tar.gz -C /opt** wird das Paket im Verzeichnis /opt entpackt. Den XAMPP-Server findet man danach im neu angelegten Verzeichnis /opt/lampp.

B. Installation mittels eines Installationspaketes:

Das Installationspaket kann in einer 32-bit oder in einer 64-bit Version heruntergeladen werden.

chmod 755 xampp-linux-x64-1.8.3-3-installer.run ... Installationspaket als Programm ausführbar machen

sudo ./xampp-linux-x64-1.8.3-3-installer.run ... Installationsroutine aufrufen

Der XAMPP-Server wird hier ebenfalls im Verzeichnis /opt/lampp

installiert.

C. Aufruf des installierten XAMPP-Servers:

Anschließend kann der XAMPP-Server innerhalb eines Terminals mit Root-Rechten gestartet werden:

sudo /opt/lampp/lampp start

Dieser Befehl muss nach jedem Neustart des Systems erneut eingegeben werden. Es werden nun alle Dienste konfiguriert und gestartet. Danach im Browser **http://localhost** eingeben - es sollte die Startseite von XAMPP erscheinen.

Anmerkung:

Für den Start bzw. Stopp des Servers kann auch die grafische Oberfläche des XAMPP-Managers verwendet werden.

sudo /opt/lampp/manager-linux.run (bzw. **manager-linux-x64.run**)

Nach der Standardinstallation ist allein der FTP-Server mit einem Passwort geschützt, der Benutzer-Account heißt **nobody**, das Passwort heißt **lampp**. Mit **ftp://nobody:passwort@ihre-ip-adresse** kann mit einem Browser oder FTP-Programm auf den Server zugegriffen werden.

XAMPP-Befehle aufrufen: **/opt/lampp/lampp <Aktion>** , das sind start, startapache, startssl, startmysql, startftp, stop, stopapache, stopssl, stopmysql, stopftp, reload, reloadapache, reloadmysql, reloadftp, restart, security, php5, phpstatus, backup und weitere.

sudo /opt/lampp/lampp ... Kurzhilfe aufrufen

sudo /opt/lampp/lampp start ... XAMPP-Server starten

sudo /opt/lampp/lampp stop ... XAMPP-Server stoppen

sudo /opt/lampp/lampp backup ... Backup der Konfiguration, Logs und aller Daten

sh /opt/lampp/backup/xampp-backup-datum.sh ... Restore; ein Backup zurückspielen

Dienst	Benutzername	Passwort
XAMPP-Seiten	lampp	-
FTP	nobody	lampp
PHPMyAdmin	pma	-
MySQL	root	-

Hinweis: Dies ist der Zustand nach einer Standardinstallation. Der Benutzername **root** für **MySQL** ist **nicht** identisch mit dem Systemadministrator **root** für **Linux**. Mit dem Start des Skripts **/opt/lampp/lampp security** können die Passwörter gesetzt werden, falls dies für ein Entwicklungssystem z.B. für PHP-Programme überhaupt erforderlich ist.

User-Directory (public_html) aktivieren:

In `/opt/lampp/etc/httpdconf` ist das Kommentarzeichen vor dem Eintrag `Include etc/extra/httpd-userdir.conf` zu entfernen. Anschließend ist XAMPP neu zu starten.

Der Eintrag befindet sich ziemlich am Schluss der Konfigurationsdatei `httpd.conf`.

```
[...]
# User home directories
Include etc/extra/httpd-userdir.conf
[...]
```

Anmerkung: Das Verzeichnis `public_html` sollte bereits im Home-Verzeichnis der Benutzer existieren.

siehe auch: Skript-Listings: Start- und Stop-Skript für XAMPP

xkill

Mit `[Strg] + [F2]` das Schnellstartfenster aufrufen und dort `xkill` eingeben und bestätigen. Der Mauszeiger auf der grafischen Oberfläche verwandelt sich in einen Totenkopf oder in ein Kreuz. Wählt man mit diesem Totenkopf ein Programmfenster mit einem Mausklick aus, welches auf Benutzereingaben nicht mehr reagiert, so wird dieses augenblicklich beendet und geschlossen. Anschließend kann es einige Sekunden dauern, bis sich `xkill` selbstständig wieder beendet.

siehe auch: `skill`, `xkill`

xrandr

Mit `xrandr` wird ermittelt, an welchem Anschluss sich der Monitor befindet und welche Bildschirmauflösungen für diesen Monitor zur Verfügung stehen.

siehe auch: `man xrandr`

xset

Das Tool kann verschiedene Parameter des X-Servers (Bildschirmausgabe für Tastaturbefehle, Mauszeiger, etc.) setzen und verändern.

`xset [-display host:dpy] option ...`

Einige Medienplayer (z.B. VLC) versuchen von sich aus, den Start des Bildschirmschoners zu verbieten. Diese Option hat aber keine Auswirkungen auf die üblichen Stromsparfunktionen, die den Bildschirm über DPMS (Display Power Management Signaling) abschalten.

Hinweis: Der hier vorgeschlagene Weg (`xset s off -dpms`) funktioniert auf einigen Rechnern nicht.

`xset s off -dpms` ... Bildschirmschoner (s off) und DPMS (Display Power Management Signaling – Stromsparfunktionen; -dpms) abschalten

`xset s on +dpms` ... Bildschirmschoner (s on) und DPMS (Display Power Management Signaling – Stromsparfunktionen; +dpms) wieder einschalten

Werden die beiden Befehle häufiger benötigt, so können sie auch in ein Shellskript verpackt werden und im Verzeichnis `/usr/local/bin` abgelegt werden. Um den Aufruf zu vereinfachen, kann dem Shellskript über die Systemeinstellungen des Desktops eine noch freie Tastenkombination zugeordnet werden.

siehe auch: `man xset`

yapet

Yapet ist ein Passwortmanager, der die Daten mit einem 448-Bit-Schlüssel nach dem Blowfish-Algorithmus verschlüsselt. Mit Yapet werden nicht nur die Daten auf der Festplatte verschlüsselt, sondern auch die im RAM befindlichen Passwörter. Die übersichtliche Bedienoberfläche ist auch in einer SSH-Session lauffähig.

Nach der Installation von Yapet sind im Home-Verzeichnis die beiden versteckten Dateien **.yapet.pet** und **.yapet** zu erstellen. Die Zugriffsrechte dieser beiden Dateien sind mit `chmod` auf 0600 zu ändern. In der Datei **.yapet.pet** werden die verschlüsselten Passwörter abgelegt. In der Datei **.yapet** können einige Parameter (z.B. die Datei die beim Aufruf von yapet geladen wird) dauerhaft gespeichert werden. Yapet kann mehrere Passwortdateien verwalten.

Hinweis: Yapet unterstützt in der aktuellen Version nur den englischen Zeichensatz, d.h. deutsche Umlaute und andere landestypische Zeichensätze werden von Yapet zur Zeit nicht unterstützt.

Öffnet Yapet einen gespeicherten Datensatz im **Nur-Lesen-Modus** (Schutz des Passwortes: zufälliges Lesen des Passwortes durch Dritte), so wird das gespeicherte Passwort nicht angezeigt. Der Schreibmodus wird mit der Tastenkombination **[Strg] + [E]** aktiviert. Danach ist das gespeicherte Passwort sofort wieder lesbar und bearbeitbar.

chmod 0600 .yapet.pet

chmod 0600 .yapet

Datei: .yapet

`load=/home/<benutzername>/.yapet.pet`

siehe auch: man yapet

ZIP

siehe auch: gzip, tar, unrar, unzip, p7zip

Zugriffsrechte

Um die Zugriffe auf Dateien und Verzeichnisse zu regulieren, werden diesen Eigentümer zugeordnet, die allein über die Zugriffsrechte (Permissions) der anderen Benutzer bestimmen können.

Zur flexibleren Gestaltung dieser Regulierung existieren neben den Eigentümer (owner) noch zwei Kategorien für jede Datei bzw. Verzeichnis - Benutzergruppe (group) und übrige Anwender (other ... »Rest der Welt«).

Jeder Systembenutzer gehört mindestens einer Gruppe an und kann gleichzeitig Mitglied mehrerer Benutzergruppen sein. Indem der Eigentümer die Zugriffsrechte für eine Gruppe und für die übrigen Anwender festlegt, kann er den Kreis der berechtigten Personen für jede Datei und für jedes Verzeichnis individuell bestimmen.

Die Rechte werden bei einem langen Listing mit **ls -l** im ersten Feld angezeigt. Die erste Stelle gibt den Dateityp an (d ... directory = Verzeichnis, - ... normale Datei, l ... Link). Der Rest des Feldes besteht aus drei Gruppen zu je drei Stellen. Hier bedeuten:

r ... read, Leseberechtigung (bei Verzeichnissen das Recht, den Inhalt des Verzeichnisses anzuzeigen)

w ... write, Schreibberechtigung (bei Verzeichnissen das Recht, eine Datei oder ein Unterverzeichnis anzulegen und auch zu löschen)

Bei einer Datei mit Schreibberechtigung, kann die Datei vollständig geleert werden. Sie kann aber nur gelöscht werden, wenn der Benutzer für das Verzeichnis auch eine Schreibberechtigung besitzt.

x ... execute, Ausführungsberechtigung (bei Verzeichnissen die Möglichkeit, in dieses Verzeichnis zu wechseln)

Bei interpretierten Programmen wie z.B. Shellskripts oder Perl-Programmen, wird für diese Datei neben der Ausführungsberechtigung auch Leserecht benötigt, um sie ausführen zu können.

Beispiel:

Typ	Eigentümer	Gruppe	Andere
d	r w x	r - x	r - x

Typ	Eigentümer	Gruppe	Andere
<hr/>			
	4 + 2 + 1	4 + 2 + 1	4 + 2 + 1

Dies Verzeichnis (d) besitzt die Zugriffsrechte 755 (4+2+1=7, 4+1=5, 4+1=5), d.h. der Eigentümer darf in diesem Verzeichnis lesen + schreiben, die Gruppe und der Rest der Welt nur lesen.

Typ	Eigentümer	Gruppe	Andere
-	r w -	r w -	r - -
<hr/>			
	4 + 2 + 1	4 + 2 + 1	4 + 2 + 1

Diese normale Datei (-) besitzt die Zugriffsrechte 664 (4+2=6, 4+2=6, 4), d.h. der Eigentümer und die Mitglieder der Gruppe können diese Datei lesen und beschreiben (verändern), während die anderen Benutzer die Datei nur lesen können.

Bedeutung des Zugriffsmodus bei Verzeichnissen

Verzeichnisse werden in vielerlei Hinsicht wie die anderen Dateitypen behandelt, die Bedeutung der Modi für die Zugriffsrechte können aber nicht einfach übernommen werden. Die Übersetzung der Wirkung von Zugriffsrechten auf Verzeichnisse sieht folgendermaßen aus:

lesbar: Bei Verzeichnissen bedeutet die Lesbarkeit, dass die berechtigten Personen den Inhalt des Verzeichnisses sehen können. Ohne dieses Recht können Programme wie ls kein Listing des Verzeichnisses anzeigen.

schreibbar: Wenn ein Verzeichnis schreibbar ist, können Dateien darin angelegt, umbenannt und gelöscht werden. Das Löschen einer Datei ist auch erlaubt, wenn diese Datei selbst nicht verändert werden darf.

Ausführbar: Die Ausführbarkeit eines Verzeichnisses erlaubt es den berechtigten Benutzern, in dieses Verzeichnis als aktuelles Verzeichnis zu wechseln und auf die Dateien darin zuzugreifen. Wenn ein Verzeichnis ausführbar, aber nicht lesbar ist, kann auf die Dateien oder Unterverzeichnisse «blind» zugegriffen werden.

Eigentum an Dateien

Linux unterscheidet sechs verschiedene Dateitypen, die im Dateisystem gespeichert werden: normale Dateien, Verzeichnisse, Gerätedateien, Sockets, FIFOs und Links. Alle diese Objekte werden bei ihrer Erzeugung in den Datenstrukturen des Dateisystems eingetragen. In diesen Einträgen

(den I-Nodes) wird unter anderem die Information über den Eigentümer und die Benutzergruppen, die der Datei zugeordnet wird, festgehalten. Jede Datei kann nur einem Eigentümer und einer Benutzergruppe gehören. Das Programm ls zeigt Eigentümer und Gruppe jeder Datei an, wenn es mit der Option -l aufgerufen wird. Die dritte Spalte zeigt den Namen des Eigentümers, die vierte Spalte zeigt den Namen der Gruppe.

Dateitypen:

Normale Dateien: Reguläre Dateien (regular files) werden mit einem Bindestrich (-) dargestellt. Sie können ebenso gut lesbare Gedichte wie ausführbare Maschinenprogramme enthalten. Sie belegen normale Datenblöcke auf der Festplatte.

Verzeichnisse: Verzeichnisse (directories) sind genauer betrachtet, Dateien, in denen auf eine spezielle Art weitere Dateien enthalten sind. Auch sie belegen Datenblöcke auf der Festplatte. Verzeichnisse werden mit einem d dargestellt.

Geräte-dateien: Gerätedateien sind Bindeglieder zwischen den Hardwarekomponenten und Geräten (devices) am Rechner bzw. den Gerätetreibern im Kernel auf der einen Seite und der Software im Laufzeitsystem, also den Anwenderprogrammen, auf der anderen. Gerätedateien werden mit einem b (blockorientierte Gerätedatei) oder mit einem c (zeichenorientierte Gerätedatei) dargestellt.

Sockets: Sockets sind Spezialdateien aus dem Bereich der TCP/IP-Vernetzung, mit denen der Datenaustausch zwischen zwei lokal laufenden Prozessen über das Dateisystem realisiert werden kann. Sockets werden mit einem s dargestellt.

FIFOs: FIFOs (named pipes) stellen eine zweite, einfachere Methode des Datentransports zwischen zwei Prozessen über das Dateisystem dar. Wie die Pipelines in der Shell können FIFOs Daten nur in einer Richtung transportieren. FIFOs werden mit einem p dargestellt.

Link: Links sind zusätzliche Namen (Verzeichniseinträge) für existierende Dateien. Es werden symbolische und Hardlinks unterschieden. Während Hardlinks vollkommen gleichwertige Verzeichniseinträge für existierende Dateien sind, bestehen symbolische Links aus einer Spezialdatei, deren Inhalt ein Zeiger auf eine andere Datei ist (in der Windowswelt Verknüpfung genannt). Links werden mit einem l dargestellt.

Eigentum an Prozessen

Linux betrachtet nicht nur die mehr oder weniger »festen« Daten auf einem dauerhaften Speichermedium als Objekte, die einem Eigentümer zugeordnet werden. Der Eigentumsbegriff wird in gewisser Weise auch auf die im Speicher befindlichen Daten und die laufenden Prozesse angewendet. Jedes Kommando, das ein User über die Tastatur eingibt, erzeugt einen Prozess im Arbeitsspeicher des Rechners. Im normalen Betrieb befinden sich immer mehrere Prozesse gleichzeitig im Speicher, die vom Betriebssystem streng voneinander abgegrenzt werden. Die einzelnen Prozesse werden mit allen Daten, die in ihrem virtuellen Adressraum enthalten sind, einem Benutzer als Eigentümer zugeordnet. Die Eigentümer der Programme werden vom Programm `ps` angezeigt, wenn es mit der Option `-u` aufgerufen wird.

SUID-Bit

Jeder Prozess der gestartet wird, hat einen Besitzer, der normalerweise derjenige Benutzer ist, welcher das Programm aufruft.

Wenn der Benutzer keine Zugriffsberechtigungen auf bestimmte Ressourcen hat, darf auch die von ihm gestartete Anwendung nicht darauf zugreifen. Dies passiert, weil das Programm die Arbeitsumgebung und somit die darin enthaltenen Benutzer- und Gruppen-IDs vererbt.

In den meisten Fällen funktioniert diese Lösung gut, einige Anwendungen erfordern jedoch höhere Berechtigungen als die der gewöhnlichen Benutzer.

Als Beispiel kann der Befehl `passwd` dienen, mit dem die Benutzerpasswörter geändert werden. Ein normaler Benutzer hat keine Schreibrechte für die Datei `/etc/shadow`, wo standardmäßig verschlüsselte Passwörter aufbewahrt werden, und doch kann jeder sein Passwort ändern und somit `/etc/shadow` modifizieren.

Das Problem wird gelöst, indem für die Dauer der Ausführung dem Programm eine andere Benutzer-ID (womit die Zugriffsberechtigungen festgelegt werden) zugewiesen wird, und zwar die seines Besitzers. Wenn der Besitzer von `passwd` also `root` ist, verfügen wir über Administrator-Rechte, während wir mit dem Programm arbeiten. So eine Zuweisung ist möglich, wenn das Programm dazu berechtigt ist, und darüber entscheidet das SUID-Bit («set-user-id» oder «Setuid-Bit»).

Ist für eine Programmdatei das SUID-Bit gesetzt, wird sie in der Ausgabe des Befehls `ls -l` nicht als gewöhnliche ausführbare Datei, sondern als Programm das seine effektive ID ändern kann gelistet, und zwar indem der Buchstabe `s` an der Stelle von `x` ausgegeben wird:

`ls -l /usr/bin/passwd`

`-rwsr-xr-x 1 root root 33924 passwd`

Hinweis: Ist das SUID-Bit gesetzt, steht hier statt des x ein S oder falls der aktuelle Benutzer nicht über Ausführrechte verfügt - ein kleines s.

Das SUID-Bit kann vom Programmbesitzer oder vom Superuser gesetzt werden. Wir tun das, indem wir den Befehl `chmod` mit dem Buchstaben `s` statt `x` als Parameter aufrufen, zum Beispiel:

`chmod u+s <Programmname>`

bzw.

`chmod 4755 <Programmname>`

Dabei sollten wir immer daran denken, vorsichtig mit diesem Bit umzugehen. Vom Standpunkt der Systemsicherheit bildet jedes Programm mit gesetztem SUID-Bit eine Möglichkeit für Unbefugte, höhere Berechtigungen zu erlangen.

Ein »Set-User-ID Programm« bildet auf diese Weise ein »intelligentes Tor« zwischen den ansonsten hermetisch voneinander abgeriegelten Bereichen zweier User.

Das Angebot eines solchen Tor's setzt das Vertrauen des Eigentümers in die Zuverlässigkeit des Programms voraus, das mit seinen eigenen Rechten laufen soll. Der größte anzunehmende Unfall bei der Verwendung des SUID-Bits würde eintreten, wenn der Benutzer eine interaktive Shell mit den Rechten des Dateieigentümers bekommen würde. Insbesondere bei Fehlfunktionen des Programms Schaden an Systemdaten oder an den Dateien anderer User anrichten.

Systemprogramme, die bei der Installation von Linux in den SUID-Modus gesetzt werden, sind gut getestet und nach dem gegenwärtigen Wissenstand zuverlässig und sicher.

SGID-Bit

Es gibt eine weitere Ausnahme von der Zuordnung des Eigentums an Dateien nach dem Verursacherprinzip: der Eigentümer kann bestimmen, dass die in diesem Verzeichnis erzeugten Dateien der gleichen Benutzergruppe gehören wie das Verzeichnis selbst. Das geschieht, indem das Verzeichnis den S-Modus für die Gruppe (»Set-Group-ID« oder »Setgid-Bit«) bekommt.

Die Zugriffsrechte auf ein Verzeichnis werden durch das SGID-Bit nicht verändert. Um eine Datei in einen solchen Verzeichnis anzulegen, muss ein User das Schreibrecht in der für ihn zutreffenden Kategorie (Eigentümer, Gruppe, andere User) haben. Wenn ein User z.B. weder der Eigentümer

noch Mitglied der Benutzergruppe eines SGID-Verzeichnisses ist, muss das Verzeichnis für die anderen User beschreibbar sein. Die in dem SGID-Verzeichnis erzeugte Datei gehört dann der Gruppe des Verzeichnisses, auch wenn der User selbst dieser Gruppe nicht angehört.

Das SGID-Bit verändert nur das Verhalten des Betriebssystems beim Erzeugen neuer Dateien. Der Umgang mit bereits existierenden Dateien ist in diesen Verzeichnissen völlig normal. Das bedeutet beispielsweise, dass eine Datei, die außerhalb des SGID-Verzeichnisses erzeugt wurde, beim Verschieben dorthin ihre originale Gruppe behält - wohingegen sie beim kopieren die Gruppe des Verzeichnisses bekommen würde.

Auch das Programm `chgrp` arbeitet in SGID-Verzeichnisse völlig normal: der Eigentümer einer Datei kann sie jeder Gruppe zueignen, der er selbst angehört. Gehört der Eigentümer nicht zu der Gruppe des Verzeichnisses, kann er die Datei mit `chgrp` nicht dieser Gruppe zueignen - dazu muss er sie in dem Verzeichnis neu erzeugen.

Es ist zwar möglich, auch bei einem Verzeichnis das SUID-Bit zu setzen, diese Einstellung hat aber keine Wirkung. Das Betriebssystem erlaubt es Usern nicht, Dateien an andere User »zu verschenken«.

Beispiel:

Die Datei `backup` befindet sich im Verzeichnis `test`.

```
drwxrwxr--  2 root archive 48 Nov 19 17:12 backup
```

Mit dem Befehl **`chmod g+s ./test`** bzw. **`chmod 2774 ./test`** wird das SGID-Bit gesetzt.

Die Zugriffsrechte sehen danach wie folgt aus:

```
drwxrwsr--  2 root archive 48 Nov 19 17:12 backup
```

Sticky-Bit

Zusätzlich zu dem SUID-Bit und dem SGID-Bit gibt es noch das so genannte Sticky-Bit (T-Bit). Hierbei muss man unterscheiden, ob es einem ausführbaren Programm oder einem Verzeichnis angehört. Für Dateien ist dieses Bit heute nicht mehr weit im Gebrauch und hat nur noch historische Bedeutung.

Wird dagegen einem Verzeichnis dieses Attribut zugewiesen, verhindert dies, dass Benutzer sich ihre Dateien gegenseitig löschen. In Verzeichnissen mit Sticky-Bit dürfen Benutzer nur Dateien entfernen, die sie selbst besitzen. Typische Beispiele sind die Verzeichnisse `/tmp` und `/var/tmp`.

`chmod 1777 /tmp`

```
drwxrwxrwt  2 root root 1160 2002-11-19 17:15 /tmp
```

Hinweis: Ist das Sticky-Bit gesetzt, steht hier statt des x ein T oder falls der

aktuelle Benutzer nicht über Ausführrechte verfügt - ein kleines t.

siehe auch: chmod, chown, chgrp

* * * * * * * * * *

Anhang

Software-Empfehlungen

HTTrack

HTTrack kopiert komplette Webseiten aus dem Internet.

siehe auch: www.httrack.com

Portfwd

<http://portfwd.sourceforge.net>; Portfwd (Port Forwarding Daemon) ist ein frei konfigurierbarer Redirector der eingehende Signale (TCP-, UDP-Pakete) auf einen anderen Rechner umleiten kann.

vlc

VideoLAN ist ein einfacher portierbarer Multimedia-Player für verschiedene Audio- und Videoformate (MPEG-1, MPEG-2, MPEG-4, DivX, mp3, ogg, ...), DVDs, VCDs und verschiedenen Streaming-Protokolle.

Inkscape

Inkscape ist ein vektororientiertes Grafikprogramm für verlustfreie Vergrößerungen und Verkleinerungen. Das Programm unterstützt eine Vielzahl von Ein- und Ausgabeformaten wie svg, ps, pdf, pov oder epsi - um nur einige zu nennen.

siehe auch: www.inkscape.org

Darktable

Einige Digitalkameras speichern die Fotoaufnahmen die während der Aufnahme entstehen im RAW-Format (Rohdaten). Erst nach der Bearbeitung der Rohdaten werden die eigentlichen Foto-Dateien exportiert. Die Originale im RAW-Format bleiben wie im vordigitalen Zeitalter des Films unangetastet. Die Fotobearbeitung Darktable eignet sich für die Verwaltung und Bearbeitung umfangreicher Sammlungen solcher RAW-Dateien.

Xpdf

Xpdf ist ein schneller PDF-Betrachter - schneller als der Acrobat Reader. Die grafische Bedienoberfläche ist bei Xpdf auf das notwendigste reduziert.

Nachteil: Es werden nicht alle PDF-Funktionen unterstützt - einfach ausprobieren.

Okular

Der PDF-Betrachter Okular (KDE-Programm) kommt vor allem mit

mehrspaltigen PDF-Dokumenten besser zurecht. Im Auswahlmodus kann man in Okular per Maus einen beliebigen rechteckigen Bereich auswählen und für die weitere Bearbeitung als Text in die Zwischenablage kopieren.

Diffuse

Das Python-Skript (<http://diffuse.sourceforge.net/>) vergleicht bis zu 3 Dateien miteinander. Die Unterschiede werden farblich hervorgehoben.

Focuswriter

Focuswriter ist eine auf das wesentliche reduzierte Textverarbeitung. Das Programmfenster von Focuswriter ist stets bildschirmfüllend. Die Menüliste zeigt sich erst, wenn der Mauszeiger an den oberen Bildschirmrand gesetzt wird.

Xnviewmp

Xnviewmp ist Bildbetrachter der auch mit exotischen Bildformaten zurechtkommt. Das Installationspaket muss allerdings von der Herstellerseite (www.xnview.com) heruntergeladen werden.

Calibre

Mit Calibre können EPUB-Dateien (EPUB ... elektronisches Buchformat) geöffnet und gelesen werden. Für das gelegentliche Lesen reichen aber Erweiterungen für die Internet-Browser Firefox oder Chrome aus.

Playonlinux

Playonlinux ist ein grafisches Werkzeug für Wine und bringt bei der Installation ein aktuelles Wine gleich mit. Mit Wine können eine Vielzahl von Windows-Programme unter Linux installiert werden.

dvdbackup

Mit dvdbackup können 1:1 Kopien von DVDs über die Kommandozeile angefertigt werden. Alternativ kann man den Hauptfilm, Titelzusammenstellungen und Einzeltitel sichern.

Audacity

Audacity ist ein Mehrspur-Audio-Editor für Linux/Unix, MacOS und Windows. Er wurde entwickelt für einfaches Aufnehmen, Abspielen und Bearbeiten von digitalen Audiodaten. Audacity enthält Digitaleffekte und Werkzeuge zur Spektralanalyse. Unterstützte Dateiformate sind Ogg Vorbis, MP2, MP3, WAV, AIFF und AU.

SoundConverter

Wer auf ausufernde Detailsinstellungen verzichten kann und möglichst

unkompliziert Audiodateien konvertieren will, wird mit dem Audio-Programm SoundConverter zufrieden sein.

OpenShot

OpenShot ist ein einfach zu bedienendes Video-Schnittprogramm. Die aufgeräumte Oberfläche orientiert sich an den gängigen Standards für Schnittprogramme.

siehe auch: <http://www.openshotusers.com/help/1.3/de/>;

<http://wiki.ubuntuusers.de/OpenShot>; OpenShot-Dokumentation

installieren: `sudo apt-get install openshot-doc`

(`/usr/share/gnome/help/openshot/de/openshot.xml`; die Datei `openshot.xml` mit LibreOffice öffnen und im Home-Verzeichnis als PDF-Datei speichern)

photofilmstrip - Slideshow creator mit den »Ken Burns Effekt«

photofilmstrip erstellt aus einer Bilderserie einen Videofilm. Der Videofilm kann mit einem Background-Sound unterlegt werden. Hinweis:

photofilmstrip befindet sich teilweise noch in einem experimentellen Stadium (Stand: 2014).

Moovida-Media-Center

Moovida ist ein Medienabspieler (Media-Center). Es erzeugt automatisch eine digitale Bibliothek, die mit der Fernbedienung durchsucht werden kann. Die einfache Benutzerschnittstelle zeigt automatisch passende Bilder und erlaubt den Zugriff auf Filmzusammenfassungen und Künstlerinfos.

DVDStyler

DVDStyler ist eine Plattform-unabhängige Anwendung zum Erstellen von DVDs für Video-Enthusiasten, um professionell erscheinende Video-DVDs zu erzeugen.

Die Hauptmerkmale sind:

- Erstellen einer Video-DVD mit interaktiven Menüs
- Unterstützung für AVI, MPEG, VOB und andere Dateiformate
- Direktes Verschieben von Videodateien per Drag und Drop
- Importieren von Bilddateien für den Menühintergrund
- Schaltflächen, Text, Bilder und grafische Objekte können überall auf dem Menü-Bildschirm angeordnet werden und vieles mehr.

Dvdauthor - erstellt DVD-Video-Dateisysteme

Dvdauthor ist ein Programm, das aus einem gültigen mpeg2-Stream einen DVD- Film erstellt. Dieser sollte von einem DVD-Player abgespielt werden können.

videotrans - DVD-Erstellungswerkzeuge

videotrans ist eine Sammlung von Utilities für die Erstellung von DVDs.

Die Werkzeuge sind:

- movie-to-dvd ... Konvertierung von MPEG2 + MP2 oder AC3
- movie-title ... kombiniert Video- und Titelsequenzen in einem Menü
- movie-make-title ... erstellt ein Background-Video für das DVD-Menü
- movie-make-title-simple ... erstellt ein Background-Bild + Audiosound für das DVD-Menü und vieles mehr

EncFS

Die Verschlüsselungs-Software EncFS ist in den Standard-Repositories aller namhaften Linux-Distributionen enthalten. Für EncFS gibt es auch einige Programme mit grafischer Oberfläche, die intern auf das Terminalprogramm Encfs zugreifen.

Dukto R6

Mit Dukto ist ein einfacher Peer-to-Peer-Dateiaustausch im Netzwerk möglich. Der kleinste gemeinsame Nenner zwischen Linux, Windows und Mac-OS X ist Samba. Dukto R6 geht einen anderen Weg: Anwender können sich damit Dateien, Nachrichten oder die Zwischenablage schicken. Die Projektwebseite bietet Dukto für Linux, Android, Windows und Mac-OS X.

Webseite: www.msec.it/blog/?page_id=11

Gigolo

Gigolo ist ein Laufwerksmanager für Netzwerkressourcen. Das Programm arbeitet mit dem Gnome Virtual File System (GVFS) und spricht alle Protokolle, die auch Gnome spricht: SSH, FTP, Samba (Windows-Netzwerk) und Web-DAV. Diese Verbindungen können mit Gigolo als Laufwerke eingebunden werden. Unter Debian, Ubuntu und Linux-Distributionen die auf Ubuntu basieren, wird Gigolo wie folgt installiert:

sudo apt-get install gigolo gvfs-backends gvfs-fuse fuse-utils

In der Linux-Distribution Xubuntu ist Gigolo standardmäßig vorinstalliert.

aunpack

Das Kommandozeilen-Tool unpack ist Bestandteil des Paketes **atool**. Die Packer und Entpacker haben eine mehr oder weniger unterschiedliche Syntax. Das Programm unpack sorgt für eine identische Syntax für alle gebräuchlichen Entpacker (ZIP, TAR.GZ, TAR.BZ2, lzma, 7z, ...):

aunpack -e *.*[Archivtyp]* (z.B. **aunpack -e *.zip**)

Unetbootin

Unetbootin gibt es für Linux, Windows und Mac-OS X. Es hat sich zum Klassiker für das Erstellen bootfähiger USB-Sticks entwickelt und ist für Linux in den Paketquellen vieler Distributionen enthalten. Um ein ISO-Image bootfähig auf USB-Stick zu befördern, ist dieses vorher mit dem Dateisystem FAT32 zu formatieren. Unetbootin kennt die meisten populären Distributionen und Live-Systeme und kann diese auf Wunsch aus dem Internet herunterladen. Es können aber auch ISO-Abbilder die bereits auf der Festplatte liegen ausgewählt werden.

Nach der Auswahl des ISO-Images (»Distribution« oder »Abbild«), ist für »Typ« USB-Laufwerk und für »Laufwerk« die Laufwerksbezeichnung des USB-Sticks (z.B. /dev/sdd1) auszuwählen. Anschließend kann der Kopiervorgang gestartet werden.

Hinweis: Das Kopieren des ISO-Images und die Erstellung des Bootloaders ist streng genommen nur dann zuverlässig möglich, wenn das System bekannt ist. Unetbootin verwendet einfach einen universellen Standard-Bootloader, der zwar meistens, aber nicht überall funktioniert.

Yumi

Yumi (Your Universal Multiboot Installer) gibt es Debian-basierte Linux-Distributionen (Debian, Ubuntu, Linux Mint, ...) und für Windows. Yumi kann im Gegensatz zu Unetbootin mehrere Linux-Distributionen auf einen bootfähigen USB-Stick befördern. Und beim Booten in einem Auswahlmenü anbieten.

Die Linux-Variante ist in den Repositories nicht enthalten, das deb-Paket kann aber nach dem Download mit der jeweiligen Paketverwaltung installiert werden (Kontextmenü über die rechte Maustaste aufrufen).

Schritt 1: Ziellaufwerk auswählen (USB-Stick)

Schritt 2: Linux-Distribution auswählen

Schritt 3: ISO-Image auswählen

Beim Booten des USB-Datenträgers erscheint der Yumi-Bootloader und bietet unter »Linux Distributions« die eingerichteten Systeme an. Standardmäßig lädt er nach 30 Sekunden Wartezeit das System der ersten Festplatte.

Hinweis: Das Kopieren des ISO-Images und die Erstellung des Bootloaders ist streng genommen nur dann zuverlässig möglich, wenn das System bekannt ist. Yumi lässt daher den Schritt 3 (ISO-Image auswählen) erst zu, wenn vorher die Distribution ausgewählt wurde.

Download: www.pendrivelinux.com/yumi-multiboot-usb-creator/

Suchmaschinen

Spezielle Schlüsselwörter: Google

site: die Suche wird auf bestimmte Webseiten oder Domains eingegrenzt - z.B. linux site:www.domainname.de

daterange: sucht nur in bestimmten Datumsbereichen, dabei bezieht sich das Datum auf das Indizierungsdatum bei Google (Google verwendet den julianischen Kalender) und nicht auf das Erstellungsdatum der Webseite - z.B. "Karl Marx" daterange: 2452389-2452389

intitle: beschränkt die Suche auf den Seitentitel - der Titel der im HTML-Code zwischen <title></title> steht - z.B. intitle: "Linux"

inurl: sucht nur in den URLs der Webseiten - z.B. inurl:linux

intext: sucht nach Dokumenten, bei denen der oder die Suchbegriffe nur im Text der Datei vorkommen; Beispiel: intext:"Bearbeiten von Google"

filetype: ermöglicht die Suche nach bestimmten Dateieendungen - z.B. "Internet Pro" filetype:pdf

ext: wie filetype; Google kann derzeit in folgende Formate »reinschauen«: PS, PDF, AI, DOC, PPT, XLS, SWF, sowie alle textbasierten Dateien wie RTF, TXT, ASP, PHP, CGI, HTML, LOG, INI, JS, usw.

info: gibt eine Seite zurück, die Links zu näheren Informationen über einen bestimmten URL enthält; Beispiel: "Max Mustermann" inurl:impressum

link: liefert eine Liste mit Seiten zurück, die über einen Link auf den angegebenen URL verweisen (Linkpopularität, Page Rank, verwandte Seite) - z.B. link:www.apachefriends.org

inanchor: sucht nur in Links nach den Begriffen; Oftmals führt ein Link mit einer bestimmten Bezeichnung präziser zu einem Ziel, als wenn die Bezeichnung irgendwo im Text vorkommt.

related: sucht nach »verwandten« Seiten aus dem gleichen Themenbereich

cache: findet eine Kopie einer Seite, die von Google indiziert wurde, selbst wenn diese nicht mehr unter der ursprünglichen URL erreichbar ist; die Inhalte werden über einen bestimmten Zeitraum noch bei Google

gespeichert und können von dort aufgerufen werden

define: durchsucht Internet-Enzyklopädien wie Wikipedia und andere nach einer Definition des Suchbegriffes

Links

Linux-Distributionen:

- <http://www.linuxmint.com>; Linux Mint
- <http://www.ubuntu.com>; Ubuntu
- <http://centos.org>; CentOS
- <http://www.debian.org>; Debian
- <http://fedoraproject.org/de>; Fedora
- <http://www.opensuse.org>; OpenSuse
- <http://xubuntu.org>; Xubuntu
- <http://siduction.de>; Siduction
- <http://www.mageia.org>; Mageia
- <http://manjaro.org>; Manjaro
- <http://www.bodhilinux.com>; Bodhi Linux

Linux Dokumentationen:

- <http://www.howtoforge.com>; detaillierte Anleitungen zur Server-Konfiguration
- <http://www.debian.org/doc>; Portal zur offiziellen Debian-Dokumentation
- <http://wiki.hetzner.de>; Hoster-Wiki zu anspruchsvollen Themen
- <http://www.administrator.de>; Community-Seite mit Fragen und Antworten
- <http://www.linuxwiki.org>; deutschsprachiges Wiki mit gesammeltem Wissen
- <http://www.thomas-krenn.com/wiki>; Admin-Know-how eines Hosters zum Nachschlagen
- <http://www.selflinux.org>; Linux-Dokumentation
- <http://debiananwenderhandbuch.de>; Debian-Online-Handbuch
- <http://www.linux-community.de>; Linux-Community - Artikel von Zeitschriften
- <http://wiki.ubuntuusers.de>; Ubuntu-Wiki

Linux News:

- <http://www.pro-linux.de>; deutschsprachiges Portal mit Linux-Neuigkeiten
- <http://www.linux-magazin.de/NEWS>; NEWS - Linux-Magazin
- <http://planet.ubuntuusers.de>; Ubuntu News

Hardware: Linux-Nutzer sind in der Wahl der Hardware nicht so frei wie Windows-Anwender. Gerade neue Geräte werden von Linux oft erst mit einer gewissen Verzögerung unterstützt. Vor dem Kauf eines Gerätes sollte man sich daher als Linux-Anwender informieren.

Hardware allgemein:

- <http://linuxwiki.de>; Tippsammlung zu Linux in Wiki-Form
- <http://www.linux-fuer-alle.de>; Tipps zur Einrichtung unterschiedlicher Hardware unter Linux
- <http://www.linux-usb.org>; Informationen über den Betrieb von USB-Geräte unter Linux
- <http://www.tuxhardware.de>; Händler von mit Linux getesteter Hardware
- <http://www.linux-usb.org/usb-ids.html>; aktuelle Liste mit Gerätenamen von USB-Geräten; Ermittlung eines Gerätenamens anhand der USB-ID (siehe auch: `lsusb`)

Drucker:

- <http://www.turboprint.de>; Treiber für Tintenstrahldrucker, kostenpflichtig

Test und Analysen:

- <http://browsercheck.pcwelt.de>; Internet-Browsercheck von der Zeitschrift PC-Welt (u.a. Firewall-Check, Portscanner)
- <https://panopticlick.eff.org/>; Panopticlick - im Internet ist jeder einzigartig und auch erkennbar; Panopticlick zeigt was jeder Internet-Benutzer von seinem Rechner und Internet-Browser im allgemeinen preisgibt – die Internet-Benutzer werden fast so einzigartig, wie ihre Fingerabdrücke
- <http://myip.is>; die eigene, aktuelle IP-Adresse ermitteln

Links - Sonstiges:

- www.nocrew.org/software/httptunnel.html; HTTP-Tunnel
- <http://portfwd.sourceforge.net>; Portfwd (Port Forwarding Daemon) ist ein frei konfigurierbarer Redirector der eingehende Signale (TCP-, UDP-Pakete) auf einen anderen Rechner umleiten kann; siehe auch: Linux-Magazin 08/2005 - Artikel: Aus dem Alltag eines Sysadmin: Portfwd
- www.andre-simon.de; Highlight ist ein Programm mit dem man Quellcode einfärben kann, z.B. um ihn in Textdokumenten (PDF) weiter zu verarbeiten
- www.nwlab.net; Nmap Tutorial von Mirko Kulp
- <http://openvpn.sourceforge.net>; OpenVPN
- <http://www.linuxfibel.de>; Dokumentationen
- <http://thekelleys.org.uk/dnsmasq/doc.html>; DNSmasq ist ein schlanker Domain Nameserver
- <http://fcron.free.fr>; Fcron ist eine Alternative zu den allgegenwärtigen Cron
- <http://www.easylinux.de>; Linux-Zeitschrift »easy-Linux«
- <http://www.linux-user.de>; Linux-Zeitschrift »Linux-Magazin«

- <http://www.linux-magazin.de>; Linux-Zeitschrift »Linux-User«
- <http://www.pcwelt.de>; Linux-Zeitschrift: »Linux Welt«
- <http://pkgs.org>; Suchmaschine für Programmpakete für viele Linux-Distributionen
- <http://www.apachefriends.org>; Serverpaket XAMPP: Webserver Apache, das PHP- und Perl-Modul, den FTP-Server ProFTPD und die Datenbank MySQL
- <http://www.iana.org/assignments/port-numbers>; aktuelle Liste der Portnummern

Linux-Live-Distributionen:

- <http://www.knopper.net/knoppix-mirrors/> → Knoppix ist eine Live-CD bzw. DVD mit sehr vielen Programmen zum testen, wie auch ein fast unverzichtbares Werkzeug für Administratoren
- <http://partedmagic.com> → PartedMagic eine Live-CD mit einem Partitionsmanager, Ghost for Linux G4L (Backup von Partitionen etc.) und vieles mehr; für den Download von Parted Magic ist eine kleine Spende erforderlich
- <http://sourceforge.net/projects/g4l/files/g4l%20ISO%20images/> → Ghost for Linux (G4L) als Live-CD; Backup und Restore von Partitionen oder ganzen Festplatten; Hinweis: An der Eingabeaufforderung ist **g4l** einzugeben.
- <http://blog.lesslinux.org> → LessLinux Search and Rescue ist hilfreich bei vielen Wartungs- und Rettungsarbeiten an Windows- und Linux-Rechnern
- <http://pogostick.net/~pnh/ntpasswd/> → Live-CD mit den NT Password Changer
- <http://www.finnix.org/Download> → Finnix ist eine kleine Linux-Live-Distribution für Administratoren; Finnix enthält viele kleine Programme u.a. auch gpg; Finnix wird ausschließlich im Textmodus (ohne grafische Oberfläche) ausgeführt
- <http://www.geexbox.org/download/> → der Medienplayer GeeXboX ist Live-System das auf den Mplayer basiert; GeeXboX spielt DVDs, MP3-, OGG-, WAV-, Real-Player-Dateien und einige andere Multimedia-Formate
- <http://www.mandalka.name/privatix/> → Privatix von Markus Mandalka ist ein umfangreiches Live-System zum anonymen Surfen
- www.dban.org oder <http://sourceforge.net/projects/dban/files> → DBANs einzige Aufgabe ist das unwiderrufliche Löschen von Festplatten oder anderen beschreibbaren Datenspeichern
- <http://www.slax.org> → Slax ist eine auf Slackware basierende GNU/Linux-Distribution, die sich als Live-CD direkt von einer CD starten lässt; die Funktionalität des Systems kann mit optionale Module

erweitert werden

- <http://www.supergrubdisk.org> → Super-GRUB2-Disk ist ein Live-System mit dem ein nicht mehr über den internen GRUB-Bootloader startendes System gestartet werden kann
- <http://sourceforge.net> und dort nach **Rescatux** suchen → Rescatux ist ein spezialisiertes Rettungssystem für den Grub-Bootloader in den Versionen 1 und 2

Kommerzielle Software:

- **scVENUS** ist die Software für intelligentes Systemmanagement homogener und heterogener Unix-, Linux- und Windows-Netzwerke (Softwareinstallation, Benutzer- und Dateisystemverwaltung, Monitoring etc.). → scVENUS: <http://www.science-computing.de/software/system-management.html>
- **BenHur**: Kommunikationsserver -> Email-Server, Internet-Access-Server, VPN, Firewall und noch einiges mehr; Hardware plus Betriebssystem plus Software; BenHur ist schon konfiguriert und kann bequem über jeden normalen Internet-Browser angepasst werden; <http://www.pyramid.de>; <http://www.intertelsec.net/article.php?sid=25>
- **Kaspersky**: Virenschanner, Firewall etc.; www.kaspersky.com/de/

Literaturverzeichnis

- »Linux Online-Anwenderbuch« der Autoren Sebastian Hetze, Dirk Hohndel, Olaf Kirch und Martin Müller - © LunetIX 1997 - in einer Überarbeitung des Dozenten Frank Viehweger - Dresden 2000
- »Linux Grundlagen und Installation« vom Dozenten Michael Stibane - Johnsbach 2000
- »Linux im Windows-Netzwerk« der Autoren Bernd Burre, Uwe Debacher, Bernd Kretschmer, Dirk von Suchodoletz, Carsten Thalheimer, Franzis Verlag 2004
- Zeitschrift: »CHIP Linux Spezial 6/2000«
- Zeitschrift: »Internet Professionell« einige Ausgaben der Jahre 2003 und 2004
- Zeitschrift: »easy Linux« einige Ausgaben der Jahre 2004 bis 2006; www.easylinux.de
- Zeitschrift: »Internet intern 3/2004«
- Zeitschrift: »hakin9« Ausgabe März/April 2004; www.hakin9.org
- Zeitschrift: »PC-Magazin Kreativ 6« 2001
- Internetseite: Unix-Pool - Institut für Mathematik; <http://www-pool.math.tu-berlin.de/public/maildocu/gpg-intro/gpg-intro.html>
- Internetseite: Linux-Magazin 12/1999; GnuPG - Gnu Privacy Guard; Geheimsache von Thomas Bader; <http://www.linux-magazin.de>
- Internetseite: www.linux-fuer-alle.de von Jens Neppe
- Internetseite: Shellskripte; www.linux-services.org/shell/shell.html von Pawel Slabiak
- Internetseite: Shellskripte; www.keipke.de
- Buch: Knoppix - Linux einfach ausprobieren! von Ute Herzog
- Zeitschrift: LinuxUser - Das Magazin für die Praxis - LinuxUser 04/2002: su, sudo - Neue Identität von Heike Jurzik
- Zeitschrift: LinuxUser - Das Magazin für die Praxis - LinuxUser 08/2005; www.linux-user.de
- Zeitschrift: LinuxMagazin - Die Zeitschrift für Linux-Professionells - LinuxMagazin 08/2005; www.linux-magazin.de
- Zeitschrift: LinuxMagazin 04/2005 Sonderheft: BEST OF 1994-2004; www.linux-magazin.de
- Zeitschrift: LinuxMagazin - Die Zeitschrift für Linux-Professionells - LinuxMagazin 10/2006: Recht und Ordnung von Anke Börnig
- Zeitschrift: LinuxMagazin - Die Zeitschrift für Linux-Professionells - LinuxMagazin 11/2006; www.linux-magazin.de
- Rechner-Netzwerke: tcpdump von Christian Kauhaus; Friedrich-Schiller-Universität Jena; 2004
- Automatische Backups mit Linux und rsync - von Jürgen Donauer
- Zeitschrift: LinuxUser - Das Magazin für die Praxis - LinuxUser

06/2009

- Zeitschrift: »Linux intern« einige Ausgaben des Jahre 2010 bis 2013
- Zeitschrift: »LinuxUser« einige Ausgaben des Jahre 2010 bis 2013;
www.linux-user.de
- Zeitschrift: »Linux Welt« PC-Welt einige Ausgaben des Jahre 2013 und 2014; www.pcwelt.de
- Internetseite: www.wikipedia.de; Wikipedia
- Zeitschrift: ct-magazin; Bilder S. 59, 197, 307
- sowie: Manuals, Info-Dateien - Bestandteil der Linux-Distributionen Linux Mint und Ubuntu

Skript-Listings

Skript: cronjobs

siehe auch: ls, lsuf

* * * * *

Listings-Nr. 1

Gesetzte S-Bit's (**siehe auch:** Zugriffsrechte) sind wahre Geschenke für Hacker, Trojaner & Co ... Eine komplette Aufstellung aller Dateien mit gesetzten S-Bit, die in regelmäßigen Abständen kontrolliert werden, wäre also sehr wünschenswert.

Die Basisliste mit allen Dateien, bei denen das S-Bit gesetzt ist, erhalten Sie durch den Befehl:

```
find / -perm +4000 > ~/sbits.base
```

Die Liste wird im Home-Verzeichnis in der Datei sbits.base gespeichert. Damit Sie auch noch die anderen Dateieigenschaften wie Größe, Datum und so weiter erhalten, betten Sie das find-Kommando in ls ein:

```
ls -l | find / -perm +4000 > ~/sbits.base
```

Diese Aufstellung ist schon eine gute Baseline, die Sie als Schablone für den »Normalzustand« Ihres Systems verwenden können. Die Baseline kann jetzt immer mit den aktuellen Status Ihres Systems über ein diff-Aufruf verglichen werden:

```
ls -l | find / -perm +4000 > ~/sbits.base | diff ~/sbits.base -
```

Getreu dem Unix-Motto »no news are good news«, liefert dieser Aufruf keine Meldungen, wenn alles in Ordnung ist, also keine Änderungen in den S-Bits vorliegt. Haben Sie selbst diese Änderungen vorgenommen, ist alles in Ordnung. In diesem Fall müssen sie die Baseline aktualisieren. In allen anderen Fällen, heißt es »Alarmstufe rot« und es sind dringend Sicherungsmaßnahmen zu ergreifen.

Das Ganze kann mit einem Cronjob automatisiert werden, dafür ist der crontab-Editor aufzurufen (**siehe auch:** crontab):

```
0 * * * * ls -l | find / -perm +4000 > ~/sbits.base | diff ~/sbits.base -
```

Der Cronjob wird zu jeder vollen Stunde aufgerufen.

Ergänzung: Die nachfolgende Anwendung des Kommandos **find** findet alle Programme, die das SUID-Bit (Eigentümer) und das SGID-Bit (Gruppe) gesetzt haben:

find / -perm +6000

* * * * *

Listings-Nr. 2

Dateien mit S-Bits sind nicht die einzigen Dateien, für die sich das Anlegen von Baselines lohnt. Das Herz Ihrer Linux-Konfiguration, die Dateien im Verzeichnis /etc liegen ebenso im Ziel eines Hackers. Hier können z.B. neue Ports geöffnet (inetd.conf) und Benutzer eingerichtet oder modifiziert werden (passwd).

Der Befehl ls bietet Ihnen eine Reihe von Optionen, mit denen Sie die Zeitinformationen aller Dateien von /etc und seiner Unterverzeichnisse auslesen können. Eine vollständige Zeitangabe erreichen Sie durch die Option **--full-time** in Kombination mit **-l**. Wenn Sie jetzt noch **-R** für rekursives Abarbeiten des Verzeichnisses und den Parameter **-a** zum Anzeigen aller Dateien verwenden, erhalten Sie eine umfassende Aufstellung der Eigenschaften der Konfigurationsdateien in /etc. Diese Aufstellung ist wieder eine gute Baseline für Ihr System:

ls -lRa --full-time /etc > ~/etc.base

Über ein diff-Kommando können Sie die Unterschiede sich anzeigen lassen und damit die Veränderungen identifizieren. Dies bedeutet jedoch auch wieder, dass Sie die Baseline neu anlegen müssen, wenn Sie Änderungen in /etc vornehmen, andernfalls erhalten sie »Fehlermeldungen«.

Das Ganze kann mit einem Cronjob automatisiert werden, dafür ist der crontab-Editor aufzurufen (siehe: crontab):

0 * * * * ls -lRa --full-time /etc | diff ~/etc.base -

Der Cronjob wird zu jeder vollen Stunde aufgerufen.

* * * * *

Listings-Nr. 3

Unter Unix-Systemen ist es im Gegensatz zu Windows-Betriebssystemen möglich, Dateien zu öffnen und sie anschließend im Dateibaum zu löschen. Obwohl diese Dateien aus der Sicht des Dateisystems nicht mehr existieren, können diese dennoch weiterhin gelesen und geschrieben werden, solange sie noch geöffnet sind.

So verblüffend dieser Effekt auch ist, so beliebt ist er auch bei Hackern. Eingeschleuste Software wie Trojanische Pferde und Sniffer müssen häufiger Daten zwischenspeichern.

Damit dies nicht auffällt, öffnen sie selbst neu angelegte Dateien und löschen diese anschließend im Dateisystem. Rein oberflächlich betrachtet, erweckt dies keinen Verdacht mehr.

Der Befehl **ls -l** (list open files) bietet ihnen die Möglichkeit offene Dateien anzuzeigen. Wenn Sie diesem die Option **+L** mit auf dem Weg geben, zeigt er Ihnen alle offene Dateien mit einem Link-Count kleiner eins an. Der Link-Count sagt aus, wie viele Einträge im Dateisystem auf diese offene Datei verweisen. Ist dieser Zähler null, so gibt es keinen Eintrag im Dateibaum auf diese Datei. Die Datei ist also gelöscht und genau diese Dateien wollen wir ja aufspüren.

Ein typisches Programm, das grundsätzlich solche Dateien ohne böse Absichten verwendet, ist der Apache Webserver. Diese »offenen, gelöschten« Dateien dürfen Sie nicht beunruhigen. Einen Filter, der Ihnen alle offenen Dateien des Apache Webserver heraus filtert, können Sie mit **awk** anlegen.

```
ls -l | awk '{if(NR != 1 && $1 != "httpd") {print}}'
```

Das Ganze kann mit einem Cronjob automatisiert werden, dafür ist der crontab-Editor aufzurufen (siehe auch: crontab):

```
0 * * * * ls -l | awk '{if(NR != 1 && $1 != "httpd") {print}}'
```

Der Cronjob wird zu jeder vollen Stunde aufgerufen.

```
* * * * * * * * * *
```

Skript: Crypt

siehe auch: useradd

* * * * *

Mit dieser Crypt-Funktion wird ein verschlüsseltes Passwort erzeugt (DES-Verschlüsselung; Standardlänge = 13 Zeichen, MD5-Verschlüsselung; Standardlänge = 34 Zeichen, BLOWFISH-Verschlüsselung; Standardlänge = 60 Zeichen). Ein DES-verschlüsseltes Passwort kann z.B. für htaccess-Dateien verwendet werden.

Voraussetzungen: ein Webserver (z.B. Apache) mit installierten PHP-Modul; die Datei mycrypt.php ist z.B. im Wurzelverzeichnis des Webserver (/srv/www/htdocs; /var/www) abzulegen;

Datei: mycrypt.php

```
<?php
if(!empty($_POST["crypt"])) {
    session_start();
    $salt_source=session_id() . dehex(time());
    if(CRYPT_STD_DES == 1) {
        $salt_DES_STANDARD=substr($salt_source,-2);
        $crypt_DES_STANDARD=crypt($_POST["crypt"],
        $salt_DES_STANDARD);
    }
    else
    {
        $crypt_DES_STANDARD="wird nicht unterstützt";
    }
    if(CRYPT_EXT_DES == 1) {
        $salt_DES_EXPANDED=substr($salt_source,-9);
        $crypt_DES_EXPANDED=crypt($_POST["crypt"],
        $salt_DES_EXPANDED);
    }
    else
    {
        $crypt_DES_EXPANDED="wird nicht unterstützt";
    }
    if(CRYPT_MD5 == 1) {
        $salt_MD5="$1$" . substr($salt_source,-9);
        $crypt_MD5=crypt($_POST["crypt"],$salt_MD5);
    }
}
```



```

else
{
$cript_MD5="wird nicht unterstützt";
}
if(CRYPT_BLOWFISH == 1) {
$salt_blowfish_a="$2$" . substr($salt_source,-13);
$cript_BLOWFISH_A=crypt($_POST["crypt"],$salt_blowfish_a);
}
else
{
$cript_BLOWFISH_A="wird nicht unterstützt";
}
if(CRYPT_BLOWFISH == 1) {
$salt_blowfish_b="$2a$" . substr($salt_source,-12);
$cript_BLOWFISH_B=crypt($_POST["crypt"],$salt_blowfish_b);
}
else
{
$cript_BLOWFISH_B="wird nicht unterstützt";
}
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>mycrypt</title>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
</head>
<body>
<br><br>
<form action="/script8a.php" method="POST">
<table align="center" border="0">
<tr>
<td colspan="2">Geben Sie hier eine beliebige Zeichenkette ein!</td>
</tr>
<tr>
<td colspan="2"><table><tr><td><input type="text" name="crypt"
value="" size="30" maxlength="128"></td><td><input type="submit"
name="go" value="« Go »"></td></tr></table></td>
</tr>
<tr>
<td><b>DES-STANDARD:</b></td>

```

<td><?php echo "\$crypt_DES_STANDARD"; ?></td>
</tr>
<td>DES-ERWEITERT:</td>
<td><?php echo "\$crypt_DES_EXPANDED"; ?></td>
</tr>
<td>MD5:</td>
<td><?php echo "\$crypt_MD5"; ?></td>
</tr>
<td>BLOWFISH (\$2\$):</td>
<td><?php echo "\$crypt_BLOWFISH_A"; ?></td>
</tr>
<td>BLOWFISH (\$2a\$):</td>
<td><?php echo "\$crypt_BLOWFISH_B"; ?></td>
</tr>
<td colspan="2">BEACHTEN: Es werden nicht immer alle Verschlüsselungsverfahren unterstützt. Mit dieser Crypt-Funktion wird ein verschlüsseltes Passwort erzeugt (DES-Verschlüsselung; Standardlänge = 13 Zeichen, MD5-Verschlüsselung; Standardlänge = 34 Zeichen, BLOWFISH-Verschlüsselung; Standardlänge = 60 Zeichen). Ein DES-verschlüsseltes Passwort kann z.B. für htaccess-Dateien verwendet werden. Die Passwörter werden i.d.R. in der Datei /etc/shadow gespeichert. Das unverschlüsselte Passwort, das hier eingegeben wird, sollte eine Länge von mindestens 8 Zeichen haben. Das verschlüsselte Passwort hat immer eine Länge von 13 Zeichen (DES), 34 (MD5) bzw. 60 Zeichen (BLOWFISH). Hinweis: Mögliche Algorithmen sind bei den meisten Linux-Distributionen die Verschlüsselungsverfahren: DES, MD5 und BLOWFISH. Diese Algorithmen erzeugen alle Einwegschlüssel, d.h. eine Entschlüsselung ist nicht möglich bzw. vorgesehen. Wird als Salt - crypt("geheim",\$salt) - das gespeicherte verschlüsselte Passwort benutzt, so erzeugt die crypt-Funktion bei Eingabe des richtigen Passwortes einen identischen Schlüssel. So dass ein Vergleich von erzeugtem Schlüssel (Passwort: geheim) und gespeichertem Schlüssel (Passwort: geheim) möglich wird. </td>
</tr>

</table>

</form>

</body>

</html>

Hinweis: Das verschlüsselte Passwort wird in der Datei **/etc/shadow** gespeichert. Ein Beispieleintrag könnte in etwa so aussehen:

username:Xldlkasoo2bn90lsal:12455:0:99999:-1::

Das verschlüsselte Passwort ist die Zeichenfolge nach username, zwischen den beiden Doppelpunkten (:). Bei einer DES-Verschlüsselung besteht diese Zeichenfolge aus genau 13 Zeichen. Besteht diese Zeichenfolge aus genau 60 Zeichen, so wird sehr wahrscheinlich die BLOWFISH-Verschlüsselung verwendet und bei 32 bzw. 34 Zeichen die MD5-Verschlüsselung.

Beispiel: das Passwort soll **nostromo** heißen

DES: **2CCu.2JCMimEY**

MD5: **\$1\$KzO6eCI2\$TGMTzaJpvbfG3TTtAGBbA0**

BLOWFISH:

\$2a\$10\$g4e99hx0AWogZLfNMR789uUd8.VQenw1ndXEYahNO03hDuabE7J.2

SHA-512:

\$6\$19uKKdas/ehHdq0E\$IRlxcO1uQcRV6PVvEi2m3IHRl5t4xJB7bgTvWQH0uRX/m.WRHxXdx2t3GFVqXcFagiW.odTgCBt5UtUGZ03wfl

* * * * *

Skript: Backup mit tar

Automatische Backups mit Linux

Sind Ihnen Backups zu umständlich, zu teuer, zu aufwendig? Mit Linux können Sie einzelne Rechner oder ganze Abteilungen zeitgesteuert und ohne teure Backup-Software sichern.

Datenverlust ist der Alptraum eines jeden Administrators. Das ist oft nicht nur ärgerlich, sondern kostet auch richtig Geld. Deswegen ist es nur allzu vernünftig, eine Backup-Strategie zu entwickeln.

Linux bringt alle benötigten Tools bereits mit. tar, ssh, find, cat und bash sind in nahezu jeder Distribution enthalten. Einzeln betrachtet sehen die Anwendungen auf den ersten Blick vielleicht etwas ungeeignet für ein professionelles Backup aus, doch gemeinsam und richtig eingesetzt, entwickelt sich aus ein paar Handgriffen eine gute und kostengünstige Lösung.

Dabei ist Verschlüsselung heutzutage wichtiger denn je - besonders dann, wenn Sie vertrauliche Daten sichern. Denn die Anzahl der einfach zu bedienenden Datenspionage-Tools nimmt stetig zu. Sicher darf man nicht in jedem Anwender einen potenziellen Hacker sehen, aber Vorsicht ist die Mutter der Porzellankiste.

Empfehlung: Die im Workshop behandelten Skripte und Dateien sollten im Home-Verzeichnis von **root** abgelegt werden, z.B. in dem Unterverzeichnis **/root/backup**. Da einige Verzeichnisse gesichert werden sollen, die Dateien enthalten die nur **root** einsehen kann.

Autorisierung über ssh

Kein Administrator hat Lust, nachts um drei Uhr ein Passwort einzugeben, damit die Sicherung anläuft. Deswegen verwenden wir eine Authentifizierungsmethode über einen so genannten **public_key**. Das im Installationsstandard enthaltene Paket **openssh** stellt diese Option zur Verfügung.

Um den Workshop anschaulicher zu gestalten, legen wir folgendes Beispielszenario fest:

linux_a: Das ist der Rechner, der gesichert werden soll. Da normalerweise nur ein User alle Zugriffsrechte besitzt, stößt root das Backup-Skript an. IP-Adresse 192.168.1.200.

linux_b: Aus Sicherheitsgründen ist ein remote login von root nicht erlaubt.

Alle Backups nimmt der Benutzer **backup** an. IP-Adresse 192.168.1.100.

root von Rechner **linux_a** soll sich an Rechner **linux_b** verschlüsselt, aber nur mit den Rechten von User **backup** anmelden. Der Vorgang muss ohne Passwort-Eingabe ablaufen, sonst wäre das Backup nicht automatisierbar.

Schlüsselpaar erstellen

Der erste Schritt ist die Erstellung eines ssh-Schlüsselpaares für **root** auf **linux_a**. Folgender Befehl erledigt dies:

ssh-keygen -t rsa

Die drei nächsten Zeilen jeweils mit »**Enter**« bestätigen. Wichtig ist, keine Passphrase einzugeben. Wenn Sie hier ein Kennwort vergeben, verlangt die Anmeldung als User **backup** an **linux_b** nach einem Passwort. Und genau das wollen wir vermeiden. Im Verzeichnis **/root/.ssh** befindet sich nun unter anderem die Datei **id_rsa.pub**. Diese enthält den öffentlichen Schlüssel von **root**.

Benutzer einrichten

Der nächste Schritt führt uns auf den Rechner **linux_b** (Rechner auf dem die Backup's landen). Zunächst legen Sie als **root** mit dem Befehl

useradd -m backup

den Benutzer **backup** an. Dieser erhält kein Passwort. Damit ist weder lokal noch remote jemand in der Lage, sich als **backup** anzumelden. Das soll nur via **public_key** funktionieren. Mit

su - backup

wechseln wir zu dem gerade erzeugten User. Ist das Verzeichnis **.ssh** nicht vorhanden, legen Sie es bitte an und wechseln dorthin.

mkdir .ssh

cd .ssh

Damit der ssh-Daemon weiß, wer überhaupt als Benutzer **backup** auf den Rechner darf, teilen wir ihm das in der Datei **authorized_keys** mit.

touch authorized_keys

legt eine zunächst leere Datei an. In diese Datei kopieren Sie nun den Inhalt

der Datei **/root/.ssh/id_rsa.pub**.

Vorsicht: Auch wenn es nicht so aussieht - der Inhalt ist eine Zeile ohne Zeilenumbruch. Um nicht über diesen Stein zu stolpern, kopieren Sie am besten die Datei **id_rsa.pub** auf **linux_b** und fügen den Schlüssel mit dem cat-Befehl

```
cat id_rsa.pub >> /home/backup/.ssh/authorized_keys
```

ein.

Firewall

Falls auf beiden Rechnern die Firewall aktiviert ist, so sollte zumindest der Port den **ssh** benutzt (i.d.R. **Port 22**) auf **linux_b** (Rechner auf dem die Backup's landen) geöffnet sein.

Test der Konfiguration

Nun sollte ein Einloggen von **linux_a** (Rechner der gesichert werden soll) nach **linux_b** (Rechner auf dem die Backup's landen) möglich sein. Als **root** auf **linux_a** probieren Sie das bitte aus.

```
ssh backup@192.168.1.100
```

Nur beim ersten Login-Versuch fragt Sie **linux_a**, ob Rechner **linux_b** in der Liste der bekannten Systeme aufgenommen werden soll. Sobald Sie das mit einem ausgeschriebenen »yes« bestätigt haben, ist die Konfiguration abgeschlossen. Probieren Sie das am besten mit den Befehlen

```
exit
```

und (der exit-Befehl führt zurück auf **linux_a**)

```
ssh backup@192.168.1.100
```

aus. Diese Funktionalität ist die Grundlage für den weiteren Workshop.

Komplettes Backup mit tar

tar ist so etwas wie der Dinosaurier unter den Backup-Programmen. Nur ausgestorben ist es noch nicht. Der Name tar steht für Tape Archiver. Die Bezeichnung ist etwas irreführend, da historisch bedingt. Das Utility kann seinen Output auf alle beschreibbaren Geräte, in Dateien oder via pipes übergeben. Der grundsätzliche Aufruf ist:

tar [Optionen] <archiv> <Dateien/Verzeichnisse>

<i>Die wichtigsten Schalter von tar</i>	
-c	create erzeugt eine neues Archiv
-f	Archive-Datei
-r	append hängt Dateien an vorhandenes Archiv an
-t	listet den Inhalt eines Archivs auf
-u	update hängt nur Dateien an, die neuer als die Kopie im Archiv sind
-x	extract packt Dateien aus dem Archiv aus
-j	Archiv durch bzip2 filtern
-Z	Archiv durch compress filtern
-z	Archiv durch gzip filtern
-p	Informationen über Dateizugriffsrechte mit extrahieren
--exclude=<Datei>	Datei <Datei> ausschließen
-X, --exclude-from=<Datei>	Inhalt aus <Datei> ausschließen

Wie bei jedem Linux-Tool ergibt auch hier ein Blick in die man-Page einen Gesamtüberblick:

man tar

Beispiel-Skript:

Für unser Beispielszenario sieht ein Beispiel-Skript so aus (backup_tar.sh):
User **root** auf **linux_a** triggert die Sicherung an.

```
00001 #!/bin/bash
00002 #
00003 ##### Anfang Konfiguration #####
00004 #
00005 # In der Regel ist nur die Variable REMOTE an Ihr System anzupassen.
00006 # Die anderen Eintraege koennen meistens unveraendert uebernommen
00007 # werden.
00008 # ACHTUNG: In den nachfolgenden Dateinamen bitte KEINE
```

Leerzeichen verwenden.

00009 # *Am besten nur das englische Alphabet, Zahlen, Punkt (.), Minuszeichen (-)*

00010 # *und den Unterstrich (_) verwenden.*

00011 #

00012 REMOTE="192.168.1.100" # *IP-Nr. des Rechners auf den gesichert wird; Backup-Server*

00013 REMOTEUSER="backup" # *Benutzer, auf den ueber ssh ohne Passwort zugegriffen wird*

00014 REMOTEDIR="/home/backup/" # *Backup-Verzeichnis auf dem Remote-Rechner (Backup-Server); ACHTUNG: Verzeichnisnamen mit einem Slash (/) abschliessen*

00015 MAILTO="./backup.mail" # *enthaelt die E-Mailadressen, die im Error-Fall eine E-Mail erhalten*

00016 SUBJECT="Backup_fehlgeschlagen!" # *Im Subject (Betreff) keine Leerzeichen verwenden!!*

00017 #

00018 INCLUDE="./backup.include" # *enthaelt die Verzeichnisse bzw. Dateien, von denen ein Backup gemacht wird*

00019 EXCLUDE="./backup.exclude" # *enthaelt die Verzeichnisse bzw. Dateien, von denen KEIN Backup gemacht wird*

00020 # *HINWEIS: Am Schluss von Verzeichnis-Namen in den beiden vorgenannten Dateien kann ein Stern (*)*

00021 # *vorkommen - Wildcard fuer alle Dateien in diesem Verzeichnis.*

00022 ERROR_FILE="./error.txt" # *enthaelt den Basis-Text, der im Error-Fall per E-Mail versandt wird*

00023 DATE=\$(/bin/date +%Y-%m-%d_%H-%M) # *Datum im Format Jahr-Monat-Tag_Stunde-Minute*

00024 HOST=\$(/bin/hostname)

00025 REMOTEFILE="backup_\${HOST}_\${DATE}.tgz" # *Dateiname der Backup-Datei*

00026 TR=/usr/bin/tr

00027 PING=/bin/ping

00028 GREP=/bin/grep

00029 AWK=/usr/bin/awk

00030 TAR=/bin/tar

00031 TAROPTIONS="-c"

00032 SSH=/usr/bin/ssh

00033 CAT=/bin/cat

00034 GZIP=/bin/gzip

00035 MAIL=/usr/bin/mail

00036 #

00037 ##### *Ende Konfiguration* #####

00038 #

00039 # *ANFANG: der kompletten TAR-Sicherung*

00040 BACKUPFILES="\$(\${CAT} \${INCLUDE} | \${TR} "\n" ' ' | \${TR} -s


```

[:blank:] ' ')
00041 CHECK_REMOTE=$((${PING} -c 1 $REMOTE 2> /dev/null | $
{GREP} packet | ${AWK} '{print $4}' | ${GREP} -c ^[1]$) # gibt 1 (online)
oder 0 (offline) zurueck
00042 if [ $CHECK_REMOTE -eq 1 ]
00043 then
00044 ${TAR} ${TAROPTIONS} --absolute-names ${BACKUPFILES} --
exclude-from=${EXCLUDE} | ${GZIP} | ${SSH} ${REMOTEUSER}@${
REMOTE} "${CAT} > ${REMOTEDIR}${REMOTEFILE}"
00045 else
00046 ${CAT} ${MAILTO} | while read LINE
00047 do
00048 ${MAIL} -s ${SUBJECT} ${LINE} < $ERROR_FILE
00049 done
00050 fi
00051 # ENDE: der kompletten Sicherung
00052 exit 0

```

Hinweis: Die Zeilennummern im Skript dienen nur der Orientierung und sind nicht Bestandteil des ausführbaren Skripts. Das Skript sichert auch die unsichtbaren Verzeichnisse und Dateien die mit einem Punkt (.) im Namen beginnen.

Um aus den Quellcode mit den Zeilennummern ein funktionstüchtiges Shellskript herzustellen, gehen sie wie folgt vor:

1. kopieren sie den Quellcode von **00001 #!/bin/bash** bis zum **exit 0** in der letzten Zeile in eine Textdatei z.B. quellcode.txt.
2. öffnen sie ein Terminal und bewegen sich mit **cd** zum Verzeichnis mit der Textdatei quellcode.txt und
3. geben folgenden Befehl ein

```
sed 's/^[0-9][:blank:]]\{6\}//' quellcode.txt > backup_tar.sh
```

4. mit **chmod 0755 backup_tar.sh** wird das Shellskript ausführbar

Konfigurationsdateien

Wir benötigen noch zwei weitere Dateien. Mit **backup.include** legen Sie fest, welche Verzeichnisse berücksichtigt werden sollen, beispielsweise (den Stern am Schluss nicht vergessen):

```

/var/*
/usr/*
/etc/*

```

/boot/*
/home/*
/root/*
/opt/*
/bin/*

Die Datei `backup.exclude` gibt an, welche Verzeichnisse nicht gesichert werden sollen, also beispielsweise (den Stern am Schluss nicht vergessen):

/home/<Benutzername>/.kde/*
/home/<Benutzername>/.mozilla/*

Funktion

Das Programm prüft zuerst die Erreichbarkeit von **linux_b** über einen **ping**. Ist dies der Fall, wird der Backup-Prozess angestoßen und alles gesichert was in »**backup.include**« angegeben ist. Ausgenommen sind die Dateien und Verzeichnisse, die in »**backup.exclude**« stehen.

Bei Nichterreichbarkeit sendet das Programm eine E-Mail mit dem Betreff »**Backup_fehlgeschlagen!**«. Die Empfänger dieser Warnmeldung sind in der Datei **backup.mail** hinterlegt. Den Text für den Hinweis enthält **error.txt**.

All diese Dateien müssen im selben Verzeichnis liegen, ansonsten sind die Pfade in den entsprechenden Parametern anzupassen.

Der Parameter **BACKUPFILES** gibt die zu sichernden Dateien und Verzeichnisse vor (liest sie aus der Datei `backup.include` ein).

Um Platz zu sparen, schickt das Programm den Output des **tar-Aufrufes** zunächst durch **gzip**. Danach wird die Möglichkeit genutzt, über **ssh** ein Programm auf einem entfernten Rechner anzustoßen. Der **cat-Befehl** schreibt auf **linux_b** den Inhalt in das Verzeichnis **/home/backup** und vergibt den Dateinamen im Format

backup_<Hostname von linux_a>_<Zeitstempel>.tgz

Der Vorteil hier ergibt sich aus den klar strukturierten Bezeichnungen der Backup-Archive. Allerdings möchten Sie aus Speicherplatzgründen vielleicht nicht jeden Tag ein Backup aller Dateien durchführen. Um das zu vermeiden, gibt es inkrementelle Sicherungen.

Inkrementelles Backup mit tar und find

In Standardinstallationen ist das Programm **find** enthalten. Der Schalter **-mtime n** ist für das Aufspüren von Dateien bestimmten Alters zuständig. Dabei wird im 24-Stunden-Takt gerechnet, also $n \cdot 24$.
-mtime -4 ... findet alles, was jünger als vier Tage ist
-mtime 4 ... findet alles, was zwischen vier und fünf Tagen alt ist
-mtime +4 ... findet alles, was älter als fünf Tage ist

Das Programm **find** kann noch viel mehr: Zum Beispiel findet der Schalter **-mmin n** alles, was **n Minuten** vorher verändert wurde. Ein Blick in die man-Page von **find** lohnt sich. Für das Beispiel-Skript **backup_incremental.sh** reichen uns aber die Schalter **-mtime** und **-type**. Mit **-type** geben wir lediglich den zu suchenden Dateityp an. Die Option **f** steht für reguläre Files.

Beispiel-Skript: backup_incremental.sh

```
00001 #!/bin/bash
00002 #
00003 ##### Anfang Konfiguration #####
00004 #
00005 # In der Regel ist nur die Variable REMOTE an Ihr System anzupassen.
00006 # Die anderen Eintraege koennen meistens unveraendert uebernommen
werden.
00007 #
00008 # ACHTUNG: In den nachfolgenden Dateinamen bitte KEINE
Leerzeichen verwenden.
00009 # Am besten nur das englische Alphabet, Zahlen, Punkt (.), Minuszeichen
(-)
00010 # und den Unterstrich (_) verwenden.
00011 #
00012 REMOTE="192.168.1.100" # IP-Nr. des Rechners auf den gesichert
wird; Backup-Server
00013 REMOTEUSER="backup" # Benutzer, auf den ueber ssh ohne Passwort
zugegriffen wird
00014 REMOTEDIR="/home/backup/" # Backup-Verzeichnis auf dem Remote-
Rechner (Backup-Server); ACHTUNG: Verzeichnisnamen mit einem Slash (/)
abschlieszen
00015 AGE="-1" # -1 = Dateien nicht aelter als 24 Stunden, -2 = 48 ...
00016 MAILTO="/backup.mail" # enthaelt die E-Mailadressen, die im Error-
Fall eine E-Mail erhalten
00017 SUBJECT="Backup_fehlgeschlagen!" # Im Subject (Betreff) keine
Leerzeichen verwenden!!
00018 #
00019 INCLUDE="/backup.include" # enthaelt die Verzeichnisse bzw.
Dateien, von denen ein Backup gemacht wird
```

00020 EXCLUDE="/.backup.exclude" # enthaelt die Verzeichnisse bzw. Dateien, von denen KEIN Backup gemacht wird

00021 # HINWEIS: Am Schluss von Verzeichnis-Namen in den beiden vorgenannten Dateien kann ein Stern (*)

00022 # vorkommen - Wildcard fuer alle Dateien in diesem Verzeichnis.

00023 ERROR_FILE="/.error.txt" # enthaelt den Basis-Text, der im Error-Fall per E-Mail versandt wird

00024 TMP_INCLUDE="/tmp_backup.include" # temporaere Datei; sie wird vom Skript automatisch erstellt und wieder geloescht

00025 DATE=\$(/bin/date +%Y-%m-%d_%H-%M) # Datum im Format Jahr-Monat-Tag_Stunde-Minute

00026 HOST=\$(/bin/hostname)

00027 REMOTEFILE="incremental_\${HOST}_\${DATE}.tgz" # Dateiname der Backup-Datei

00028 TR=/usr/bin/tr

00029 PING=/bin/ping

00030 GREP=/bin/grep

00031 AWK=/usr/bin/awk

00032 TAR=/bin/tar

00033 FIND=/usr/bin/find

00034 TAROPTIONS="-c"

00035 SSH=/usr/bin/ssh

00036 CAT=/bin/cat

00037 RM=/bin/rm

00038 GZIP=/bin/gzip

00039 MAIL=/usr/bin/mail

00040 #

00041 ##### Ende Konfiguration #####

00042 #

00043 # ANFANG: incrementelle Sicherung

00044 BACKUPFILES="\$({CAT} \${INCLUDE} | \${TR} "\n" | \${TR} -s[:blank:] ' ')"

00045 CHECK_REMOTE=\$((\${PING} -c 1 \$REMOTE 2> /dev/null | \${GREP} packet | \${AWK} '{print \$4}' | \${GREP} -c ^[1]\$) # gibt 1 (online) oder 0 (offline) zurueck

00046 if [\$CHECK_REMOTE -eq 1]

00047 then

00048 \${FIND} \${BACKUPFILES} -mtime \${AGE} -type f > \$TMP_INCLUDE

00049 # HINWEIS: Wer auf Nummer sicher gehen will, ersetzt -mtime (Modifikationszeit) durch -atime

00050 # (Access- bzw. Zugriffszeit).

00051 # TIPP: Wird das Skript z.B. stuenndlich durch einen Cronjob gestartet, so ist -mtime bzw.

00052 # -atime durch -mmin bzw. -amin zu ersetzen. Der Wert der Variable AGE (siehe weiter oben)

```

00053 # ist dann durch -60 (60 Minuten) zu ersetzen.
00054 ${TAR} ${TAROPTIONS} --absolute-names --files-from=${
${TMP_INCLUDE} --exclude-from=${EXCLUDE} | ${GZIP} | ${SSH} $
${REMOTEUSER}@${REMOTE} "${CAT} > ${REMOTEDIR}${
${REMOTEFILE}"
00055 else
00056 ${CAT} ${MAILTO} | while read LINE
00057 do
00058 ${MAIL} -s ${SUBJECT} ${LINE} < $ERROR_FILE
00059 done
00060 fi
00061 ${RM} -f ${TMP_INCLUDE}
00062 # ENDE: incrementelle Sicherung
00063 exit 0

```

Hinweis: Die Zeilennummern im Skript dienen nur der Orientierung und sind nicht Bestandteil des ausführbaren Skripts. Das Skript sichert auch die unsichtbaren Verzeichnisse und Dateien die mit einem Punkt (.) im Namen beginnen.

Um aus den Quellcode mit den Zeilennummern ein funktionstüchtiges Shellskript herzustellen, gehen sie wie folgt vor:

1. kopieren sie den Quellcode von **00001 #!/bin/bash** bis zum **exit 0** in der letzten Zeile in eine Textdatei z.B. `quellcode.txt`.
2. öffnen sie ein Terminal und bewegen sich mit `cd` zum Verzeichnis mit der Textdatei `quellcode.txt` und
3. geben folgenden Befehl ein

```

sed 's/^[0-9[:blank:]]\{6\}/' quellcode.txt >
backup_incremental.sh

```

4. mit **chmod 0755 backup_tar.sh** wird das Shellskript ausführbar

Funktion

Dieses Skript arbeitet abgesehen von zwei Unterschieden wie das vorherige: Zum einen werden nur Dateien gesichert, die maximal 24 Stunden alt sind. Zum anderen fügen wir aus Gründen der Übersichtlichkeit dem Dateinamen ein `incremental` ein.

backup_incremental_<Hostname von linux_a>_<Zeitstempel>.tgz

In einem **cronjob** sieht das Ganze wie folgt aus:

1. Sonntag-Nacht um 2.30 Uhr ein volles Backup

30 2 * * 7 /<Pfad-zum-Skript>/backup_tar.sh

2. von Montag bis Sonnabend um 2.30 Uhr eine inkrementelle Sicherung

30 2 * * 1-6 /<Pfad-zum-Skript>/backup_incremental.sh

Anmerkung: siehe auch Abschnitt »Cronjobs einrichten«

Hier scheiden sich allerdings die Geister. Die einen halten es für sinnvoll nur einmal im Monat eine Gesamtsicherung zu machen. Andere wiederum ziehen ihre »Full Backup« alle 14 Tage. Bedenken Sie beim Festlegen Ihrer Backup-Strategie jedoch Folgendes:

Im Falle eines Desasters müssen Sie zuerst das volle Backup einspielen und dann jedes einzelne inkrementelle hinterher schieben. Nur so ist sichergestellt, dass der Stand der letzten Sicherung hundertprozentig stimmt.

Je kürzer die Abstände zwischen den »Full Backups« sind, desto geringer ist der Aufwand im »Worst Case«, aber umso mehr Plattenplatz wird verbraucht.

Automatisches Löschen alter Backups

Jetzt stehen wir nur noch vor dem Problem, dass der Plattenplatz auf **linux_b** beschränkt ist. Wiederum ist hier das Programm **find** nützlich.

/usr/bin/find /home/backup/ -mtime +30 -type f -exec /bin/rm {} \;

löscht alles, was im Verzeichnis **/home/backup/** älter als **30*24** Stunden alt ist. Auch das können wir wieder mit einem **cronjob** automatisieren.

30 1 * * * /usr/bin/find /home/backup/ -mtime +30 -type f -exec /bin/rm {} \;

Dieser Eintrag kann nun sowohl in der **crontab** von **root** als auch von User **backup** stehen. Der Superuser darf ohnehin alles, und die Sicherungsarchive auf **linux_b** gehören dem Benutzer **backup**.

Achtung: Zwischen **}** und **** muss sich ein Leerzeichen befinden.

Vorsicht: Der Befehl löscht ohne Rückfrage.

Spielen Sie vorher einfach ein bisschen mit **find**, um ein Gefühl dafür zu bekommen, wie das Programm arbeitet. Zum Beispiel gibt

```
/usr/bin/find /etc/ -mtime +2 -type f -exec /bin/ls {} \;
```

alle Dateien aus dem Verzeichnis **/etc/** mit allen Unterverzeichnissen auf dem Bildschirm aus, die älter als **2*24** Stunden sind.

Achtung: Zwischen {} und \ muss sich ein Leerzeichen befinden.

Praxistipps

1. Das ganze Spiel funktioniert auch in die andere Richtung. Hierbei ist allerdings ein Sicherheitsaspekt zu bedenken. Wenn Rechner **linux_b** die Daten von **linux_a** holen soll, muss er sich als **root** anmelden. Sonst lassen sich wichtige Systemdaten nicht sichern, auf die nur der Superuser Zugriff hat. In der Regel soll aber ein **direktes remote-Einloggen von root unterbunden sein**.

2. Sie möchten einen Windows-Rechner sichern? Kein Problem. Mounten Sie via Samba am Anfang des Skripts (**Hinweis:** am besten nach der Variablen MAIL=/usr/bin/mail) einen Share des Windows-PCs ins Filesystem eines Linux-Rechners. Im Backup-Skript binden Sie dieses Verzeichnis in die Sicherung mit ein. Jede Linux-Distribution liefert Samba mit. Sie legen zum Beispiel ein Verzeichnis **/mnt/windowsrechner** an.

```
mount -t smbfs -o
```

```
username=<windowsuser>,password=<windowspasswort> //<ip-  
adresse-windows-rechner>/<share> /mnt/windowsrechner
```

bzw.

```
mount -t cifs -o
```

```
username=<windowsuser>,password=<windowspasswort>,iocharset=ut  
f8,sec=ntlm //<ip-adresse-windows-rechner>/Freigabename  
/mnt/myshare
```

lautet der Befehl, um eine Windows-Freigabe ins Linux-Filesystem einzubinden. Verwenden Sie als <share> C\$ dann haben Sie Zugriff auf das komplette Laufwerk C:\\. Ein

```
umount -l /mnt/windowsrechner
```

am Ende des Skripts (**Hinweis:** am Ende des Skripts, aber vor exit 0) löst die Verbindung wieder. Da bei dieser Lösung das Backup-Skript ein

Passwort im Klartext enthält, sollte es nur für root lesbar sein.

chmod 600 backup_tar.sh

und

chmod 600 backup_incremental.sh

verhindert einen Zugriff anderer Benutzer auf diese Dateien.

Vorsicht: Ist der Windows-Rechner offline, so hat dies unter Umständen fatale Folgen. Das Verzeichnis /mnt/windowsrechner ist leer, wenn der Sicherungsprozess beginnt. Das heißt dann, dass vom Windowsrechner kein Backup erstellt wird.

Sinnvoll ist es, die Windows-Rechner von **linux_b** aus zu sichern und vorher den **ping-Check** zu vollziehen. Ist der Rechner nicht erreichbar, passiert nichts außer einer gesendeten E-Mail. Oder Sie programmieren diese Eventualität in den Beispiel-Skripts nach, um das Ganze gefahrenfrei von **linux_a** aus laufen zu lassen.

Cronjobs einrichten

Mit Cronjobs können Programme zeitgesteuert aufgerufen und abgearbeitet werden. Es wird hier davon ausgegangen, dass die Crontab-Datei im Verzeichnis **/var/spool/cron/tabs/** abgelegt wird. Der Speicherort kann je nach verwendeter Linux-Distributionen davon abweichen.

Cronjobs auf linux_a einrichten – Erstellung der Backups

1. **su ...** als root auf **linux_a** anmelden
2. **cd /var/spool/cron/tabs/ ...** ins Crontab-Verzeichnis wechseln
3. **touch crontab ...** die Crontab-Datei crontab anlegen (Dateiname kann beliebig sein, crontab hat sich eingebürgert)
4. **mcedit ./crontab ...** die Datei zum bearbeiten mit mcedit öffnen
5. folgende Zeilen in der Datei crontab einfügen; die einzelnen Parameter sind durch ein Leerzeichen zu trennen; mit [F2] speichern und mit [F10] mcedit beenden

#Minute (1...59) Stunde (0...23) Tag (1...31) Monat (1..12) Tag der Woche (0...7) Kommando

#(Tag der Woche ... 1=Montag, 0 oder 7=Sonntag, 1-5=Werktag)

30 2 * * 7 /<Pfad-zum-Skript>/backup_tar.sh

30 2 * * 1-6 /<Pfad-zum-Skript>/backup_incremental.sh

6. **crontab -u root ./crontab** ... meldet die veränderte Crontab-Datei crontab bei cron an
7. **crontab -u root -l** ... Kontrolle; listet die angemeldeten Cronjobs auf

Anmerkung: Bei jeder Änderung der Crontab-Datei, muss die Datei erneut bei **cron** angemeldet werden.

Cronjobs auf linux_b einrichten – Löschung alter Backup-Archive

1. **su** ... als root auf **linux_b** anmelden
2. **cd /var/spool/cron/tabs/** ... ins Crontab-Verzeichnis wechseln
3. **touch crontab** ... die Crontab-Datei crontab anlegen (Dateiname kann beliebig sein, crontab hat sich eingebürgert)
4. **mcedit ./crontab** ... die Datei zum bearbeiten mit mcedit öffnen
5. folgende Zeilen in der Datei crontab einfügen; die einzelnen Parameter sind durch ein Leerzeichen zu trennen; mit [F2] speichern und mit [F10] mcedit beenden

```
#Minute (1...59) Stunde (0...23) Tag (1...31) Monat (1..12) Tag der
Woche (0...7) Kommando
#(Tag der Woche ... 1=Montag, 0 oder 7=Sonntag, 1-5=Werktag)
30 1 * * * /usr/bin/find /home/backup/ -mtime +30 -type f -exec
/bin/rm {} \;
```

6. **crontab -u backup ./crontab** ... meldet die veränderte Crontab-Datei crontab bei cron an
7. **crontab -u backup -l** ... Kontrolle; listet die angemeldeten Cronjobs auf

Anmerkung: Bei jeder Änderung der Crontab-Datei, muss die Datei erneut bei **cron** angemeldet werden. **Achtung:** Zwischen {} und \ muss sich ein Leerzeichen befinden.

siehe auch: man crontab

Wiederherstellung, Restore

Nachfolgend wird nur die manuelle Wiederherstellung, Restore verlorener oder beschädigter Dateien beschrieben.

Selbstverständlich halte ich niemanden davon ab, auch diesen Vorgang durch Shellskripts zu vereinfachen – grundlegende Kenntnisse über Shellskripte vorausgesetzt. Darüber hinaus ist es auch möglich alle Shellskripte zu einem einzigen Skript zu vereinen.

1. Wiederherstellung einer einzelnen Datei

a) Mit **su** melden Sie sich auf dem Rechner **linux_a** (der Rechner der wiederhergestellt werden soll) als **root** an.

b) Kopieren Sie dann den folgenden Befehl über das Kontextmenü der rechten Maustaste in ein geöffnetes Kommandozeilenfenster (Befehl ist eine einzige lange Zeile):

```
for i in $(ssh backup@192.168.1.100 "find ./ -iname '*.tgz' | sort -nr");do SUCHWORT="perry.txt";RESULT=$(ssh backup@192.168.1.100 "tar -tzf ${i} | grep -i \"^.*${SUCHWORT}.*$\"");if [ $? -eq 0 ];then echo -e "\n${i}";fi;done
```

Für **perry.txt** in der Variablen **SUCHWORT** geben Sie den gesuchten Dateinamen oder einen Teil des Dateinamens ein. Als Ergebnis wird der Archivname bzw. Archivnamen ausgegeben, in dem die Suche fündig geworden ist.

c) Mit **scp** kopieren Sie das Archiv vom Rechner **linux_b** zum Rechner **linux_a**:

```
scp backup@192.168.1.100:<backup_name.tgz>  
/tmp/<backup_name.tgz>
```

Hinweis: Als Zielverzeichnis sollten Sie das Verzeichnis **/tmp** bevorzugen.

d) Enpacken Sie mit **tar** das kopierte Backup-Archiv ins **tmp**-Verzeichnis:

```
tar -xvzf /tmp/<backup_name.tgz> -C /tmp/
```

Hinweis: Als Zielverzeichnis sollten Sie das Verzeichnis **/tmp** bevorzugen.

e) Nun können Sie die Datei wie gewohnt suchen und ins Zielverzeichnis kopieren. Dabei gibt es allerdings eine **Stolperfalle**. Haben Sie sich z.B. als **root** angemeldet und kopieren diese Datei auch als Benutzer **root**, so kann der **ursprüngliche Benutzer** mit dieser Datei nicht mehr viel anfangen - diese Datei gehört jetzt **root**. Daher sollten Sie beim benutzen des **copy**-Befehls zusätzlich noch den Parameter **-p**

(Zugriffsrechte und auch evtl. vorhandenen erweiterten Dateirechte bleiben erhalten) benutzen.

cp -p <Quelldatei> <Zielfile>

Anschließend können Sie das Archiv und das entpackte Archiv im tmp-Verzeichnis löschen.

2. komplette Wiederherstellung z.B. nach einem größeren Disaster

Szenario: Am Freitagnachmittag wurde durch ein Bedien- oder Softwarefehler eine größere Datenmenge gelöscht. Der Administrator entscheidet sich deshalb für eine komplette Wiederherstellung, so dass nur die Daten, die an diesem Freitag erstellt wurden, ganz oder teilweise verloren sind.

a) Mit **su** melden Sie sich auf dem Rechner **linux_a** (der Rechner der wiederhergestellt werden soll) als **root** an.

b) Erstellen Sie zuerst im Verzeichnis **/tmp** von **linux_a** (der Rechner der wiederhergestellt werden soll) mit folgenden Befehl ein Verzeichnis:

mkdir /tmp/backup

Anschließend übertragen Sie in das neu erstellte Verzeichnis die benötigten Tar-Archive – Fullbackup vom Sonntag und die inkrementellen Archive von Montag bis Donnerstag. Das inkrementelle Backup-Archiv von Freitag wurde bis zum Freitagnachmittag noch nicht erstellt (**siehe auch:** Cronjobs einrichten).

Hinweis: Die Befehle sind jeweils eine einzige lange Zeile.

scp backup@192.168.1.100:backup_<Host>_<Zeitstempel-Sonntag>.tgz /tmp/backup/backup_<Host>_<Zeitstempel-Sonntag>.tgz

scp backup@192.168.1.100:incremental_<Host>_<Zeitstempel-Montag>.tgz /tmp/backup/incremental_<Host>_<Zeitstempel-Montag>.tgz

scp backup@192.168.1.100:incremental_<Host>_<Zeitstempel-Dienstag>.tgz /tmp/backup/incremental_<Host>_<Zeitstempel-Dienstag>.tgz

[...]

```
scp backup@192.168.1.100:incremental_<Host>_<Zeitstempel-  
Donnerstag>.tgz /tmp/backup/incremental_<Host>_<Zeitstempel-  
Donnerstag>.tgz
```

Hinweis: Als Zielverzeichnis sollten Sie das Verzeichnis /tmp bevorzugen.

c) Nach der erfolgreichen Übertragung können Sie nun die Archive ins Zielverzeichnis (Wurzelverzeichnis /, da in der Datei backup.include der absolute Pfad eingetragen ist) entpacken:

```
tar -xvzf /tmp/backup/backup_<Host>_<Zeitstempel-Sonntag>.tgz -  
C /
```

```
tar -xvzf /tmp/backup/incremental_<Host>_<Zeitstempel-  
Montag>.tgz -C /
```

```
tar -xvzf /tmp/backup/incremental_<Host>_<Zeitstempel-  
Dienstag>.tgz -C /
```

```
[...]
```

```
tar -xvzf /tmp/backup/incremental_<Host>_<Zeitstempel-  
Donnerstag>.tgz -C /
```

d) Löschung der nicht mehr benötigten Archive im Verzeichnis /tmp/backup:

```
rm -rf /tmp/backup
```

Überprüfen Sie nach Abschluss der Wiederherstellung stichprobenartig den Erfolg der Wiederherstellung. Sie sollten das Augenmerk besonders auf die Benutzer- und Gruppenzugehörigkeit, den Zugriffsrechten und die Modifikationszeit richten (**ls -lA**).

Falls alles reibungslos verlaufen ist, haben Sie jetzt wieder ein funktionierendes System.

Nachteil: Der Nachteil der vorgenannten Vorgehensweise ist, dass Sie auf dem wiederherzustellenden Rechner ausreichend freien Speicherplatz zu Verfügung haben müssen. Falls dies nicht der Fall ist, bleibt Ihnen nicht anderes übrig als das Home-Verzeichnis des Benutzers **backup** auf **linux_b** per NFS (Network Filesystem) in den Verzeichnisbaum von **linux_a** einzubinden (Punkt b entfällt dann).

Fazit

Bandsicherung schön und gut, aber wenn man die Kosten für Bänder und Plattenplatz gegenüberstellt, dann spricht das Ergebnis eher für einen Backupserver.

Wenn Sie diesen mit ausreichend Festplatten bestücken und mit einem RAID-5 relativ wasserdicht machen, dann ist das eine gute Lösung. Außerdem geht die Rücksicherung schneller als von Band.

Die Sicherheitsbewussten können ja trotzdem den Backup-Server noch mit einem Bandlaufwerk bestücken. »Backup ist zwar etwas für Feiglinge«, aber wohl dem, der im Ernstfall eins hat.

Empfehlung: Testen Sie die Erstellung und besonders die Wiederherstellung zuerst immer auf ein Test-System. Nach den erfolgreichen Test-Durchläufen, sollte Sie die Wiederherstellung in regelmäßigen Abständen immer wieder üben. Da Sie sich sonst beim GAU (größten anzunehmenden Unfall) nur unnötige schlaflose Nächte zumuten.

Skript: Metadaten in PDF-Dateien löschen

Bevor man PDF-Dateien öffentlich zugänglich macht, sollten einige Daten (Meta-Daten) - die mitunter persönliche Angaben enthalten - aus den PDF-Dateien entfernt werden. Das Skript »pdf-metadaten-leeren.sh« benötigt das Programmpaket pdftk.

```
001 #!/bin/bash
002 # pdf-metadaten-leeren.sh © michael_squire (ubuntuusers.de) 2013-01-18
003 # Lizenz: Creative Commons Namensnennung 3.0 Unported
004 # Hilfetext: siehe letzte 12 Zeilen dieses Skripts
005 # Hilfetext ausgeben (letzte 12 Zeilen dieses Skripts) falls Skript
    irrtuemlicherweise mit Parametern gestartet wurde:
006 if [ $1 ]; then sed -e :a -e '$q;N;13,$D;ba' $0; exit 127; fi
007
008 PDF_FILE=$(ls *.pdf 2>/dev/null | grep -c "\.pdf")
009 if [ ${PDF_FILE} -gt 0 ]; then
010     mkdir ./temp # Temporaeres Verzeichnis erzeugen
011     if [ ! -d ./PDFs-ohne-Metadaten ]; then mkdir ./PDFs-ohne-Metadaten; fi
    # Zielverzeichnis erzeugen, falls nicht schon vorhanden
012     cp ./*.pdf ./PDFs-ohne-Metadaten # alle PDFs des aktuellen Verzeichnis
    ins Zielverzeichnis kopieren
013     echo ... Kopiere PDFs in den Ordner `pwd`/PDFs-ohne-Metadaten/ ...
014     else echo pdf-metadaten-leeren.sh: Keine PDFs im aktuellen Verzeichnis
    `pwd` gefunden
015     rm -r ./temp 2>/dev/null # Temporaere Daten loeschen
016     exit 1
017 fi
018
019 # Zur Verarbeitung der Dateinamen werden voruebergehend die
    Leerzeichen in den Dateinamen durch Unterstriche ersetzt:
020 for DATEI in ./PDFs-ohne-Metadaten/*.pdf; do rename "s/_/_g"
    "$DATEI" 2>/dev/null; done
021
022 for i in $( find ./PDFs-ohne-Metadaten -type f -name "*.pdf" ); #
    Dateiname wird in Variable i geschrieben
023 do
024     DATEINAME=`basename $i`
025     DATEINAME_MIT_LEERZEICHEN=`echo $DATEINAME | sed -e
    "s/_/_g"`
026
027     cp --remove-destination ./PDFs-ohne-Metadaten/$DATEINAME
    ./temp/temp.pdf # Datei zur verarbeitung ins temporaere Verzeichnis kopieren
028
029     # Temporaere Datei mit leeren Metadaten fuer das aktuell verarbeitete
```

PDF erzeugen (die echo-Konstruktion bewirkt mehrere Zeilen anlich einem "here document"):

```
030 ( echo InfoKey: Author; echo InfoValue;; echo InfoKey: Company; echo
InfoValue;; echo InfoKey: CreationDate; echo InfoValue;; echo InfoKey:
Creator; echo InfoValue;; echo InfoKey: ModDate; echo InfoValue;; echo
InfoKey: Producer; echo InfoValue;; echo InfoKey: SourceModified; echo
InfoValue;; echo InfoKey: Title; echo InfoValue:
$DATEINAME_MIT_LEERZEICHEN ) > ./temp/blank-metadata.txt
031
032 pdftk ./temp/temp.pdf update_info ./temp/blank-metadata.txt output $i 2>/
dev/null # Geaenderte Metadaten ins kopierte PDF und in Datei schreiben
033
034 echo ... Leere Metadaten fuer Kopie von
$DATEINAME_MIT_LEERZEICHEN ... # Feedback an die Kommandozeile
035
036 done
037
038 # Leerzeichen in Dateinamen wiederherstellen:
039 for DATEI in ./PDFs-ohne-Metadaten/*.pdf; do rename "s/_/ /g"
"$DATEI" 2>/dev/null; done
040
041 unset i DATEINAME DATEINAME_MIT_LEERZEICHEN DATEI #
Variablen loeschen
042 rm -r ./temp # Temporaere Daten loeschen
043
044 exit 0
045
046 # Hilfetext:
047 pdf-metadaten-leeren.sh: Bitte ohne Parameter starten
048
049 Anleitung:
050 Dieses Skript ins Verzeichnis der PDFs kopieren und dort ausfuehren.
051
052 Verwendung:
053 ./pdf-metadaten-leeren.sh
054
055 Dieses Skript kopiert alle PDFs des aktuellen Ordners in einen
Unterordner ./PDFs-ohne-Metadaten/
056 und entfernt mit pdftk die Metadaten dieser Kopien.
057 Der Titel ("Title") in den Metadaten wird jeweils durch den Dateinamen
ersetzt.
058 Die Originaldateien werden nicht veraendert.
```

Hinweis: Die Zeilennummern im Skript dienen nur der Orientierung und sind nicht Bestandteil des ausfuehrbaren Skripts.

Um aus den Quellcode mit den Zeilennummern ein funktionstüchtiges Shellskript herzustellen, gehen sie wie folgt vor:

1. kopieren sie den Quellcode von **001 #!/bin/bash** bis zur Zeilennummer **058** in eine Textdatei z.B. `quellcode.txt`.
2. öffnen sie ein Terminal und bewegen sich mit `cd` zum Verzeichnis mit der Textdatei `quellcode.txt` und
3. geben folgenden Befehl ein

```
sed 's/^[0-9][:blank:]]\{4\}//' quellcode.txt > pdf-metadaten-leeren.sh
```

4. mit **`chmod 0755 pdf-metadaten-leeren.sh`** wird das Shellskript ausführbar
5. Skript in das Verzeichnis mit den PDF-Dateien kopieren und wie folgt aufrufen: **`bash pdf-metadaten-leeren.sh`**

Mit dem Terminalprogramm `pdfinfo` kann das Ergebnis begutachtet werden. Die veränderten PDF-Dateien befinden sich im Unterverzeichnis `./PDFs-ohne-Metadaten`.

Hinweis: Das Programm **`pdfinfo`** ist Bestandteil des Programmpaketes **`poppler-utils`**.

`pdfinfo` Datei-Name.pdf

siehe auch: `pdftk`, `pdfinfo`

Skript: Start-/Stop-Skript für den Entwicklungsserver XAMPP

Datei: /etc/init.d/lampp

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: Start-/Stop-Skript für den Server XAMPP (PHP, MySQL)
# Required-Start:
# Required-Stop:
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start und Stop - XAMPP
# Description: Start und Stop - XAMPP
### END INIT INFO
# Author: Name <email@domain.tld>

# Aktionen
case "$1" in
    start)
        /opt/lampp/lampp start
        ;;
    stop)
        /opt/lampp/lampp stop
        ;;
    status)
        /opt/lampp/lampp status
        ;;
    restart)
        /opt/lampp/lampp restart
        ;;
    reload)
        /opt/lampp/lampp reload
        ;;
esac

exit 0
```

Der Kommentar-Text im Kopfteil der Datei ist dabei sehr wichtig und wird vom Befehl update-rc.d verwendet. Er sollte angepasst, aber nicht gelöscht werden! Die Datei lampp speichert man im Verzeichnis /etc/init.d/ und macht die Datei per

sudo chmod 755 /etc/init.d/lampp

ausführbar. Anschließend fügt man das Skript mit dem Befehl

sudo update-rc.d

in die entsprechenden Runlevel ein.

Von nun an, wird bei jedem Systemstart der XAMPP-Server mit dem Parameter start aufgerufen und damit gestartet. Beim Herunterfahren des Systems wird der XAMPP-Server automatisch gestoppt.

Manuell kann der XAMPP-Server über das Start-/Stop-Skript lampp oder über das XAMPP-eigenen Skript gesteuert werden.

Start-/Stop-Skript lampp:

```
sudo /etc/init.d/lampp start ... startet den XAMPP-Server  
sudo /etc/init.d/lampp stop ... stoppt den XAMPP-Server  
sudo /etc/init.d/lampp status ... gibt den aktuellen Status aus  
sudo /etc/init.d/lampp restart ... stoppt und startet den XAMPP-Server  
sudo /etc/init.d/lampp reload ... stoppt und startet den XAMPP-Server  
und liest gleichzeitig die Konfigurationsdateien erneut ein
```

XAMPP-eigenen Skript:

```
sudo /opt/lampp/lampp start ... startet den XAMPP-Server  
sudo /opt/lampp/lampp stop ... stoppt den XAMPP-Server  
sudo /opt/lampp/lampp status ... gibt den aktuellen Status aus  
sudo /opt/lampp/lampp restart ... stoppt und startet den XAMPP-Server  
sudo /opt/lampp/lampp reload ... stoppt und startet den XAMPP-Server  
und liest gleichzeitig die Konfigurationsdateien erneut ein  
sudo /opt/lampp/lampp help ... Kurzhilfe anzeigen
```

siehe auch: Autostart mit ROOT-Rechten, XAMPP

Nützliche und hilfreiche Bash-Befehle

Beachte: Die meisten Befehle sind eine einzige lange Zeile, auch wenn dies manchmal nicht so aussieht.

Bei einigen Linux-Distributionen kann man die mitunter sehr langen Zeilen auch mittels Drag and Drop ins Kommandozeilenfenster kopieren. Bei diesem Verfahren werden aber mitunter einige Sonderzeichen fehlerhaft übertragen, auch wenn dies mitunter nicht sichtbar ist. In diesem Fall, sind die Sonderzeichen zu löschen und manuell neu einzutragen.

siehe auch: Shell

* * * * *

Quota's überprüfen

dd if=/dev/zero of=/home/<benutzername>/test

/dev/zero ... Pseudo-Gerät das ständig Nullen (0) produziert

Überprüfung von Quotas => die Datei **test** wird ständig größer, bis die quota's anspringen bzw. bis die Partition komplett gefüllt ist.

* * * * *

EXT2-Partition auf einem USB-Stick erstellen

1. Als root oder als Benutzer mit root-Rechten auf der Kommandozeile anmelden und
2. mit cd ins Wurzelverzeichnis des USB-Stick's wechseln.
3. Danach folgende Befehle ausführen:

dd if=/dev/zero of=base bs=1024 count=20480

mkfs.ext2 -F base

mkdir /mnt/data

mount -o loop base /mnt/data

chown <Benutzer>:users /mnt/data

Anschließend kann die Datei base wie eine normale Partition angesprochen werden, d.h. es können dort Verzeichnisse und Dateien angelegt und

gespeichert werden.

Die Datei base besteht aus 20480 Blöcke mit einer Blockgröße von 1024 Byte, d.h. die Partition besitzt eine Größe von 20 MByte.

Mit **umount /mnt/data** wird die Partition aus dem Verzeichnisbaum ausgehängen. Der mount-Befehl muss nach einem Neustart des Rechners von root erneut eingegeben werden.

* * * * *

Textmeldung auf den Bildschirm ausgeben

xmessage

Das Programm xmessage gibt eine Textmeldung auf den Bildschirm aus. Der Text erscheint in der Nähe der Maus. xmessage verlangt noch einen String im ISO-Format. Das Euro-Zeichen ist nicht darstellbar.

xmessage -nearmouse \$(echo "Hallo Karl - Grüße von Ötzi" | iconv -f UTF-8 -t ISO-8859-1)

Die Textdatei **test1.txt** sollte im ISO-Format vorliegen.

xmessage -file test1.txt -nearmouse

zenity

Das Programm zenity gibt ebenfalls Textmeldungen auf dem Bildschirm aus. Die Textmeldungen müssen bei zenity nicht erst ins ISO-Format konvertiert werden (**siehe auch:** man zenity, xset).

zenity --info --text "Hallo Karl - Grüße von Ötzi"

Hinweis: Auf KDE-Systeme sollte statt zenity besser kdialog verwendet werden (**kdialog --msgbox "Hallo Karl - Grüße von Ötzi"**).

* * * * *

Dateien umbenennen und Zeichenersetzungen

Beispiel 1: Leerzeichen in Dateinamen ersetzen

Ersetzt im aktuellen Verzeichnis alle Leerzeichen im Dateinamen durch einen Unterstrich (_).

for i in *.mp3;do mv "\$i" \$(echo "\$i" | tr " " _);done

Beispiel 2: Tabulator ersetzen

In PHP-Skripte den Tabulator \t durch 3 Leerzeichen ersetzen. Damit wird erreicht, dass die Ansicht in allen Editoren gleich aussieht.

```
sed 's/\t/ /g' script.php > script_2.php
```

Beispiel 3: Dateien umbenennen

Mit einer for-Schleife können mehrere Dateien in einem Zuge umbenannt werden. Im Beispiel wird die Dateiendung **txt** aller Textdateien im aktuellen Verzeichnis durch die Dateiendung **csv** ersetzt. **Hinweis:** Funktioniert nicht mit Dateinamen die ein Leerzeichen enthalten.

```
for f in *.txt; do mv ${f} ${f%.txt}csv; done
```

Beispiel 4: in Dateinamen eine Zahl einfügen

Die ursprünglichen Dateinamen bleiben erhalten, es wird aber eine lfd. Nummer angehängen. Im Beispiel beginnt die lfd. Nummer mit der Zahl 45.

```
Q=45;for i in *.mp3;do mv "$i" $(basename $i .mp3)_${Q}.mp3;Q=$((Q+1));done
```

* * * * *

MP3-Dateien**Beispiel 1:** MP3-Dateien abspielen

Alle MP3-Dateien des aktuellen Verzeichnisses abspielen:
Spielt alle MP3-Dateien im aktuellen Verzeichnis und wartet nach jeder Datei 3 Sekunden, bis die nächste Datei abgespielt wird.

```
for i in *.mp3;do mpg123 $i; sleep 3;done
```

Das Programm **mpg123** muss i.d.R. erst nachinstalliert werden. Alternativ kann auch das Programm **plaympeg** verwendet werden.

Beispiel 2: Wave-Dateien in MP3-Dateien umwandeln

Alle WAVE-Dateien des aktuellen Verzeichnisses in MP3-Dateien konvertieren:

Da **lame** keine Wildcards (*) versteht, muss man für die Konvertierung von mehreren Dateien eine for-Schleife bemühen.

Wave-Dateien (z.B. track00.wav bis track25.wav) in MP3-Dateien

konvertieren:

```
for t in *{00..25}.wav;do lame -b 128 $t; done
```

Beispiel 3: Wave-Dateien in MP3-Dateien umwandeln

Alle WAVE-Dateien im aktuellen Verzeichnis zu MP3-Dateien umwandeln. Der Basisnamen bleibt dabei erhalten, nur die Datei-Endung wird in .mp3 umgewandelt.

```
for i in $(ls -1 *.wav);do lame -b 160 ${i} "${basename $  
{i} .wav}.mp3";done
```

* * * * *

Wave-Dateien

Die mit cdparanoia erstellten Wave-Dateien erhalten per Standard den Dateinamenszusatz .cdda (z.B. track01.cdda.wav). Diese Zeichenkette kann mit folgenden Befehl wieder entfernt werden.

Variante: 1

```
for i in $(ls -1 *.wav);do mv "$i" "${basename $  
{i} .cdda.wav}.wav";done
```

bzw.

```
cdparanoia -Bz;for i in $(ls -1 *.wav);do mv "$i" "${basename $  
{i} .cdda.wav}.wav";done;eject
```

Variante: 2

```
for i in *.wav;do k=$(echo $i | sed 's/^\.cdda//g');mv $i $k;done
```

bzw.

```
cdparanoia -Bz;for i in *.wav;do k=$(echo $i | sed 's/^\.cdda//g');mv $i  
$k;done;eject
```

* * * * *

Integrität eines Verzeichnisses, samt seiner Unterverzeichnisse, überprüfen

Es werden dabei nur Veränderungen bei der Dateianzahl - hinzugefügt oder entfernt - bzw. Änderungen bei den Dateinamen erfasst. Das Löschen bzw. das Einfügen von leeren Verzeichnissen wird nicht erfasst (-type f). Änderungen an den Zugriffsrechten oder Änderungen innerhalb der einzelnen Dateien werden bei diesem Beispiel ebenfalls NICHT erfasst.

MD5-Datenbank erstellen:

```
find /home/ben/tmp -type f | md5sum > ./md5
```

MD5-Prüfsummen der Dateien mit der MD5-Datenbank vergleichen:

```
find /home/ben/tmp -type f | md5sum --warn --check ./md5
```

* * * * *

Integrität von Dateien, einschließlich der Dateien in den Unterverzeichnissen, überprüfen

Achtung: Beim Aufruf des Befehls sollte das aktuelle Verzeichnis NICHT identisch mit dem Zielverzeichnis (im Beispiel: /home/ben/tmp) sein.

Folgenden Änderungen werden nicht erfasst:

- Dateien (**siehe:** vorherigen Tipp) bzw. Verzeichnisse werden hinzugefügt
- Änderungen an den Zugriffsrechten
- Löschen von leeren Verzeichnissen (-type f)

Es werden nur Dateien überprüft die in der MD5-Datenbank aufgenommen wurden.

MD5-Datenbank erstellen:

```
for i in $(find /home/ben/tmp -type f -print);do md5sum "$i">>  
./md5;done
```

MD5-Prüfsummen der Dateien mit der MD5-Datenbank vergleichen:

```
md5sum --warn --check ./md5
```

* * * * *

Suchen nach Dateien bzw. Dateiinhalte

Beispiel 1: Textdateien durchsuchen

Sucht nur in Textdateien (a) nach "SUCHWORT" und gibt nur den Dateinamen (l) aus. Das "SUCHWORT" muss als ganzes Wort (w) vorkommen.

for i in \$(find /etc -type f -print);do grep -alw "SUCHWORT" \$i;done

Beispiel 2: Shellskripte durchsuchen

Dieses Beispiel sucht nur in Shellskripte (Endung .sh).

for i in \$(ls -l *.sh);do grep -l "SUCHWORT" \$i;done

Beispiel 3: HTML-Dateien durchsuchen

HTML-Dateien nach einem Suchwort durchsuchen;

Ausgabe: Dateiname:Zeilennummer:Zeile mit dem gefundenen Suchwort

find ./ -type f -regex "^.*\.html\$" -exec grep -Hin "^.*SUCHWORT.*" {} \;

Achtung: Zwischen {} und \ muss sich ein Leerzeichen befinden.

Beispiel 4: Sucht nur innerhalb von vordefinierten Textdatei-Typen (.txt, .php, .html etc.) nach dem "SUCHWORT".

find ./<Verzeichnis> -type f -iregex "^.*
(\..txt\|\..php\|\..html\|\..htm\|\..js\|\..css\)\$" -exec grep -Hin
"^.*SUCHWORT.*\$" {} \;

Achtung: Zwischen {} und \ muss sich ein Leerzeichen befinden.

Beispiel 5: PDF-Dateien durchsuchen

Durchsucht alle PDF-Dateien im aktuellen Verzeichnis und gibt die Zeilen mit dem gefundenen SUCHWORT im Terminal aus; es wird zwischen Groß- und Kleinschreibung unterschieden.

for i in *.pdf;do pdftotext \$i \$ausgabe.txt; grep "SUCHWORT" \$ausgabe.txt; done

Beispiel 6: PDF-Dateien durchsuchen

Durchsucht alle PDF-Dateien im aktuellen Verzeichnis und gibt die Dateinamen aus in dem SUCHWORT enthalten ist; es wird zwischen Groß- und Kleinschreibung unterschieden.

for i in *.pdf; do pdftotext \$i tmp_ausgabe.txt; if [\$(grep -c
"SUCHWORT" tmp_ausgabe.txt) -gt 0]; then echo "\$i"; fi; done; rm -
f tmp_ausgabe.txt

Beispiel 7: Verzeichnisse und Dateien finden

Findet alle Verzeichnisse und Dateien die innerhalb der letzten 14 Tage geändert wurden. **Ausnahmen:** Verzeichnisse und Dateien in den Verzeichnissen /windows, /tmp, /proc und /home

Hinweis: Befehl als root aufrufen

```
find / -mtime -14 -not -regex "^/home.*" -and -not -regex "^/proc.*" -  
and -not -regex "^/tmp.*" -and -not -regex "^/windows.*"
```

Beispiel 8: Zeilen mit einem Tabulator finden

Zeigt alle Zeilen nummeriert an, in denen sich ein Tabulator befindet.

-s ... Zeilen die keinen Trenner enthalten nicht anzeigen

-f 1-20 ... nur die Felder 1-20 ausgeben; Standardfeldtrenner ist der Tabulator

```
grep -n " file.txt | cut -s -f 1-20
```

Beispiel 9: Innerhalb von Textdateien (.txt, .php, .html etc.) nach SUCHWORT suchen.

```
find ./<Verzeichnis> -type f -iregex "^.*\  
(\\.txt\\|\\.php\\|\\.html\\|\\.htm\\|\\.js\\|\\.css\\)$" -exec grep -Hin  
"^.*SUCHWORT.*$" {} \;
```

Achtung: Zwischen {} und \ muss sich ein Leerzeichen befinden.

* * * * *

Zugriffsrechte ändern

Die Zugriffsrechte aller **Dateien** (-type f) des aktuellen Verzeichnisses und seiner Unterverzeichnisse auf 0644 ändern.

```
find ./<Verzeichnis> -type f -exec chmod 0644 {} \;
```

Die Zugriffsrechte des aktuellen **Verzeichnisses** und seiner **Unterverzeichnisse** (-type d) auf 0755 ändern.

```
find ./<Verzeichnis> -type d -exec chmod 0755 {} \;
```

Achtung: Zwischen {} und \ muss sich ein Leerzeichen befinden.

* * * * *

Bilder erstellen und bearbeiten

Beispiel 1: Bilder verkleinern

Alle Dateien im aktuellen Verzeichnis, die auf .jpg enden, auf 20% verkleinern und dem Ergebnis die Dateiergung _s.jpg zuweisen.

```
for i in *.jpg; do convert -resize 20% $i $(basename $i .jpg)_s.jpg; done
```

Hinweis: Das Programm convert ist Bestandteil des ImageMagick-Pakets. Es muss i.d.R. erst nachinstalliert werden.

Beispiel 2: Thumbnail-Dateien erstellen

Um von allen Dateien im aktuellen Verzeichnis, die auf .jpg enden, eine Thumbnail-Datei zu erstellen, deren Breite 250 Pixel (Bilder im Breit-Format) bzw. deren Höhe 250 Pixel (Bilder im Hochformat) beträgt, tippen Sie beispielsweise folgendes:

```
for i in *.jpg; do convert $i -geometry 250x250 -quality 90 ${i%.*}_thumb.jpg; done
```

Der Original-Bildname wird um die Endung _thumb.jpg erweitert.

Beispiel 3: Bilder in ein anderen Bildtyp umwandeln

Alle Dateien im aktuellen Verzeichnis, die auf .bmp enden, in jpg-Dateien umzuwandeln.

```
for i in *.bmp; do convert $i $(basename "$i" .bmp).jpg; done
```

Beispiel 4: Bildinformationen erhalten

Informationen über die Bildbreite x Bildhoehe aller GIF-Bilder im aktuellen Verzeichnis ermitteln .

```
for i in *.gif; do identify -format %f-%wx%h $i; done
```

identify -verbose bild01b.jpg ... alle Bild-Informationen anzeigen

Beispiel 5: Bild-Ausschnitt erstellen

Von dem Bild bild.jpg wird ein Bild-Ausschnitt (150x120 Pixel) herausgeschnitten. Dabei wird 100 Pixel von links und 50 Pixel von oben im Bild bild.jpg begonnen. Der Bild-Ausschnitt wird im neuen Bild ausschnitt.jpg gespeichert. Die Kompressionsrate beträgt 90%.

```
convert -crop 150x120+100+50 -quality 90 bild.jpg ausschnitt.jpg
```

Beispiel 6: Bild-Ausschnitt erstellen

Von PNG-Bilder wird ein Bild-Ausschnitt (800x600Pixel) herausgeschnitten und der Bild-Ausschnitt ins JPEG-Format umgewandelt. Anschließend wird die Größe der Bilder (Bild-Ausschnitt) auf 640x480 Pixel verkleinert.

```
for i in *.png; do IMG=$(basename "$i" .png)\.jpg; convert -crop  
800x600+300+150 -quality 100 "$i" "$IMG"; convert "$IMG" -  
geometry 640x480 -quality 95 "$IMG"; done
```

Beispiel 7: Bild-Ausschnitt erstellen

Wie Beispiel 6; aber die Ausgangsbilder für die Bild-Ausschnitte werden nach der Erstellung der verkleinerten Bild-Ausschnitte (640x480 Pixel) gleich gelöscht.

```
for i in *.png; do IMG=$(basename "$i" .png)\.jpg; convert -crop  
800x600+300+150 -quality 100 "$i" "$IMG"; convert "$IMG" -  
geometry 640x480 -quality 95 "$IMG"; rm -f "$i"; done
```

Anmerkung: Statt `$(basename "$i" .png)\.jpg` kann man auch `${i%\.JPEG}\.jpg` schreiben.

Hinweis: Die Programme `convert` und `identify` sind Bestandteil des ImageMagick-Pakets.

* * * * *

Tabellen

Tabelle: Darstellung der ASCII-Zeichen in Hexadezimalschreibweise

Sehen wir uns diese zwei Links an:

[http://127.0.0.1/inject/html_ex.php?music=<script>alert\('hakin9'\)</script>](http://127.0.0.1/inject/html_ex.php?music=<script>alert('hakin9')</script>)

[http://127.0.0.1/inject/html_ex.php?music=%3Cscript%3Ealert](http://127.0.0.1/inject/html_ex.php?music=%3Cscript%3Ealert%28%27hakin9%27%29%3C%2Fscript%3E)

[%28%27hakin9%27%29%3C%2Fscript%3E](http://127.0.0.1/inject/html_ex.php?music=%3Cscript%3Ealert%28%27hakin9%27%29%3C%2Fscript%3E)

Es ist gut zu wissen, dass sie auf dieselbe Webseite verweisen. Es ist einfach

- das ASCII-Zeichen < trägt die (hexadezimale) Codierung 3C, statt

<script> können wir also %3Cscript%3E schreiben. Wozu? Manchmal

wollen wir keine untypischen Zeichen in der URL platzieren - einige

Webanwendungen oder Clients versuchen sie möglicherweise zu entfernen.

Einige ausgewählte Zeichen und ihre Hexadezimalcodes finden Sie in der

Tabelle.

| <i>Zeichen</i> | <i>Hexadezimal-
code</i> |
|-----------------------|-------------------------------------|
| ! | %21 |
| " | %22 |
| # | %23 |
| \$ | %24 |
| % | %25 |
| & | %26 |
| ' | %27 |
| (| %28 |
|) | %29 |
| * | %2A |
| + | %2B |
| , | %2C |
| - | %2D |
| . | %2E |
| / | %2F |
| : | %3A |
| ; | %3B |
| < | %3C |
| = | %3D |

| <i>Zeichen</i> | <i>Hexadezimal-
code</i> |
|----------------|------------------------------|
| > | %3E |
| ? | %3F |
| @ | %40 |
| [| %5B |
| \ | %5C |
|] | %5D |
| ^ | %5E |
| _ | %5F |
| ~ | %7E |

Hinweis: Die hexadezimale Schreibweise muss immer in der Adresszeile eines Browsers eingegeben werden, in den Texteingabefeldern von Formularen hat diese Schreibweise nicht die gewünschte Wirkung. Werden die Webseiten mit einem PHP-Interpreter dynamisch erzeugt und ist in der Konfigurationsdatei von PHP - der php.ini - die Option **magic_quotes_gpc** eingeschaltet (magic_quotes_gpc = on), so werden einfache und doppelte Hochkommas mit einem Backslash versehen ('\''). Mit dieser Option schützen sich die Webmaster vor einigen HTML-Injection-Angriffe - dies gilt auch für die entsprechende hexadezimale Kodierung. Das heißt, es können dann nur Skripte die ohne einfache oder doppelte Hochkommas auskommen, ihre Wirkung entfalten.

Beispiel: HTML-Injection

Die nachfolgende Skripte können bei interaktiven Webseiten (z.B. Gästebücher, Foren) die unzureichend geschützt sind, einige mehr oder weniger lästige Ereignisse hervorrufen.

```
<script>var a=test(); function test()
{window.location.href="_URL_";}</script>
<script>var a=test(); function test() {window.location.reload(true);
setTimeout(test(),1000);}</script>
<script>var a=test(); function test() {windows.history.go(-3);}</script>

<script>var a=test(); function test() {windows.location.back();}</script>

<script>var a=test(); function test()
{alert(window.history.length);}</script>
```

Manchmal ist es notwendig, den Tag `<script language=“` zu verwenden. Der Code `<script language=“` am Ende würde den Rest der Seite als ein unvollendetes JavaScript-Programm erscheinen lassen (**Beispiel:**
`<script>var a=test(); function test()
{windows.location.back();}</script><script language=“`).

Achtung: Dieser Programmcode kann nicht über Cut & Paste in ein **Formularfeld** eingefügt werden, er muss in der Regel immer über die Tastatur einzeln eingegeben werden, wenn er die gewünschte Wirkung haben soll. Dies funktioniert nach ein- oder mehrmaligen Versuchen auch, wenn in der `php.ini` die Option **`magic_quotes_gpc = on`** eingeschaltet ist. Die Skripte die ohne einfache oder doppelte Hochkommas auskommen, können auch über die Adresszeile des Browsers übergeben werden (siehe weiter oben).

Schutz vor HTML-Injection-Angriffe

Werden die Webseiten mit einem PHP-Interpreter dynamisch erzeugt, so sollten Sie sich mit folgenden PHP-Funktionen näher beschäftigen.

`strip_tags(String);`

Entfernt HTML- und PHP-Tags im übergebenen `String`.

`htmlspecialchars(String, ENT_QUOTES);`

Mit dieser Funktion, werden alle Sonderzeichen und HTML-Tags durch den dafür vorgesehenen HTML-Code ersetzt.

`str_replace(Suchstring, Ersatzstring, String);`

Mit dieser Funktion können einzelne Zeichen z.B. durch den entsprechenden HTML-Code ersetzt werden, z.B.

`<` durch `<`; oder `>` durch `>`;

`preg_replace(/Suchstring/i, Ersatzstring, String);`

Mit dieser Funktion können einzelne Zeichen z.B. durch den entsprechenden HTML-Code ersetzt werden, z.B.

`<` durch `<`; oder `>` durch `>`; Sinnvoll ist diese Funktion aber nur, wenn mit **REGULÄREN AUDRÜCKEN** nach entsprechenden »gefährlichen« Zeichen gesucht wird.

Anmerkung: Die vorgenannten JavaScripte funktionieren nur auf schlecht oder nachlässig programmierten Webseiten.

Tabelle: Variablen

Über **echo <Variablenname>** kann an der Kommandozeile der Inhalt dieser Variablen angezeigt werden.

| <i>Shell Variablen</i> | <i>Erläuterung</i> |
|------------------------|--|
| \$HOME | Zeigt das Home-Verzeichnis des aktuellen Benutzers an. |
| \$HISTSIZE | Zeigt die max. Anzahl der gespeicherten Kommandos des Kommandozeilenspeichers an. |
| \$HISTFILE | Zeigt den Dateinamen des Kommandozeilenspeichers an, meist ~/.bash_history . |
| \$PATH | Jeweils durch einen Doppelpunkt getrennt, enthält die Variable alle Verzeichnisse, in denen die Shell nach einem Kommando sucht. |
| \$SHELL | Zeigt den Namen und den Pfad der Standardshell an. |
| \$RANDOM | erzeugt Zufallszahlen von 0 bis 32767 |
| \$UID | UID des aktuellen Benutzers |
| \$MAIL | Mailpfad des aktuellen Benutzers |
| \$MAILCHECK | Zeitintervall in Sekunden für die Überprüfung des Postfachs nach neuen Email's - Vorgabe 60 |

Mit **export** oder **printenv** werden alle bekannten Shell Variablen ausgegeben.

Die Pfadvariable \$PATH

Immer wieder wundern sich Linux-Neulinge, warum die Shell beim Start eines Kommando verärgert reagiert:

```
user@sonne> ls -l doit
-rwxr-xr-x    1 user  users   177 Sep 23 10:52 doit
```

```
user@sonne> doit
bash: doit: command not found
```

Der Grund wird sofort klar, wenn man sich den Inhalt der Shellvariablen \$PATH betrachtet:

```
user@sonne> echo $PATH
/usr/local/bin:/usr/bin:/usr/X11R6/bin:/bin:/usr/lib/java/bin:
/usr/games/bin: /usr/games: /opt/gnome/bin:/opt/kde/bin:
/usr/openwin/bin
```

Jeweils durch den Doppelpunkt getrennt, enthält die Variable alle Verzeichnisse, in denen die Shell nach einem Kommando sucht. Das aktuelle Verzeichnis ist nicht enthalten!

Die Reihenfolge der Pfadangaben in PATH entspricht der Suchreihenfolge der Bash. Existiert z.B. ein Kommando »**rm**« unterhalb von »/usr/local/bin« und in »/bin«, so wird immer ersteres ausgeführt (es sei denn, /bin/rm wird mit seinem vollständigen Pfad angegeben).

Um einen Pfad der Shellvariablen \$PATH zu übergeben, ist folgendes einzugeben:

```
user@sonne> export PATH=$PATH:/sbin
```

Die Übergabe wird nicht dauerhaft in der Shellvariablen \$PATH gespeichert.

Achtung: Die Pfadvariable \$PATH des Administrators sollte aus Sicherheitsgründen niemals das aktuelle Verzeichnis (. ... Punkt) mit einschließen (Beispiel: **export PATH=\$PATH:.**).

siehe auch: /etc/profile, man /etc/profile

Einführung in die Shellprogrammierung

Grundlagen: Shellskripte

Zum Schreiben eines Shellskripts verwendet man einen Texteditor wie KWrite oder Kate. Die erste Zeile beginnt immer mit der Zeichenfolge **#!/**, gefolgt vom vollen Pfad der Shell, die das Skript ausführen soll:

#!/bin/bash

In diesem Fall ist das die Standardshell **/bin/bash**, auf der Sie sonst auch Ihre Kommandos eingeben. Normalerweise gilt eine mit einer Raute (**#**) beginnende Zeile als Kommentar, den die Shell später beim Ausführen des Skripts ignoriert - die erste Zeile ist die Ausnahme von dieser Regel. Anschließend notieren Sie die Befehle, die das Skript bearbeiten soll, jeweils eine pro Zeile. Dabei können Sie jedes Kommando verwenden, das Sie sonst direkt in die Shell (Kommandozeilen-Interpreter, Terminal, Konsole) eingeben.

Speichern Sie das Skript unter einem aussagekräftigen Namen. Damit daraus ein kleines Programm wird, versehen Sie es mit Ausführungsrechten:

chmod +x meinskript.sh

bzw.

chmod 0755 meinskript.sh

Aufgerufen wird es danach entweder mit einer absoluten Pfadangabe wie

/voller/pfad/zu/meinskript.sh

oder, wenn Sie schon im Verzeichnis sind, in dem das Skript liegt, mit

./meinskript.sh

Die Zeichenfolge **./** ist die relative Pfadangabe für das aktuelle Verzeichnis, das bei Linux, anders als unter Windows, nicht standardmäßig im Suchpfad für Programme liegt. Um das Skript allen Benutzern des Rechners zugänglich zu machen, loggen Sie sich mit **su** als **root** ein oder führen als Hauptbenutzer mit einem vorangestellten **sudo** die folgenden Befehle aus. Kopieren Sie das Skript mit dem Befehl

cp meinskript.sh /usr/local/bin/

nach /usr/local/bin/ und setzen mit

chmod u=rwx,go=rx /usr/local/bin/meinskript.sh

bzw.

chmod 0755 /usr/local/bin/meinskript.sh

die Rechte so, dass es zwar jeder Benutzer ausführen, aber nur der Administrator ändern darf. Nun genügt zum Starten des Skripts der Name **ohne** Pfadangabe (dies gilt für alle normalen Benutzer, aber aus Sicherheitsgründen nicht für den Benutzer root).

Zeichen, Wörter und Spezialzeichen

In Shellskripts werden Wörter, also Kommandos und Parameter, immer durch Leerzeichen oder Tabulatoren getrennt bzw. unterschieden. Zeichen mit Sonderbedeutung sind

**;\$|&()<>{}[]?*`'~#% und **

also fast alles was sich über den Zahlentasten befindet. Diese Sonderzeichen dienen zur Prozesskommunikation und -steuerung, als Ersatz für Dateinamen und zum Zugriff auf Variablen. Diese Sonderzeichen werden in den nachfolgenden Artikeln einer ausführlichen Betrachtung unterzogen.

Variablen - Grundlage leistungsfähiger Programme

Komfortable Programmierung benötigt schon bei einfachen Aufgaben eine Möglichkeit, sich etwas vorübergehend zu merken. In fast allen Programmiersprachen gibt es deshalb Variablen, denen man Werte zuweisen und diese auch wieder abfragen kann. In der bash können Variablen ohne vorherige Vereinbarung einfach durch Wertzuweisung der Form

Name=Wert

erzeugt werden.

Name kann dabei aus alphanumerischen Zeichen aus a-z und A-Z, Ziffern und dem Unterstrich bestehen, darf aber nicht mit einer Ziffer anfangen.

Wert kann dabei leer sein, beliebige Zeichen und Ziffern, auch Leer- und Enterzeichen enthalten und wird vor seiner Zuweisung interpretiert, wobei aber keine Zerlegung in Wörter vorgenommen wird. Es darf auch kein Leer- oder Tabulatorzeichen zwischen dem Namen und dem Gleichheitszeichen (=) stehen, da sonst versucht würde, ein Programm namens Name zu finden, was sicher nicht den gewünschten Effekt hat.

Zugreifen kann man auf den Wert einer Variablen, indem man dem Namen ein Dollarzeichen (\$) voranstellt. Hier gilt das oben angeschnittene Argument, dass eine Variable vor der Expansion geschützt werden muss noch mehr, denn Variablen sind oft Ausgaben anderer Programme und dabei kann man fast nie sicher sein, ob nicht doch ungünstige Zeichen wie * oder & darin vorkommen.

Ein weiterer Fallstrick lauert, wenn Zeichenketten zusammen gehängt werden sollen. Die nahe liegende Lösung

b=\$VARIABLEtext

geht schief, da dann versucht würde eine Variable mit dem Namen

VARIABLEtext

zu finden, die aber nicht das gewünschte Ergebnis liefert. Umgehen kann man dies, indem man den Variablennamen für die Shell eindeutig macht und ihn mit geschweiften Klammern ({ ... }) einklammert.

b=\${VARIABLE}text

wäre also das sichere Vorgehen.

Die häufigste Art, einer Variablen einen Wert zuzuweisen, ist ihr die Ausgabe eines Programms zu übergeben. Dies ist durch die Konstruktion

Name=\$(Programmaufruf)

möglich. Eine andere Form existiert auch,

Name=`Programmaufruf`

und hätte dasselbe Ergebnis. Solche Programmaufrufe dürfen auch verschachtelt werden, das Ergebnis ist dann die Ausgabe des Programms ohne führende Returnzeichen.

Variablen mit spezieller Bedeutung

Für die vollständige Kontrolle eines Programmablaufes ist der Zugriff auf Größen wie die übergebenen Argumente, Rückgabewerte oder Prozessnummern notwendig. Dazu werden spezielle Variable geschaffen, die dann diese Werte enthalten. Diese Variablen kann man **nicht** selbst setzen, sondern nur auslesen.

Die wichtigsten Namen sind

***, @, #, ?, \$, 0;**

und auf sie zugreifen kann man in der üblichen Weise mit

\${Name}

In Shellskripts gilt es als guter Stil, die Rückgabewerte von Programmen abzufragen, ob sie auch Erfolg hatten, bzw. bei Fehlermeldungen einen sinnvollen Kommentar auszugeben. Dazu genügt es meist nicht, den Rückgabewert mit einer if-Anweisung zu testen. Dazu stellt die Shell nun die spezielle Variable **?** zur Verfügung, die immer den Rückgabewert des zuletzt ausgeführten Programms enthält.

\$? ... Rückgabewert (Status) des zuletzt ausgeführten Kommandos

Shellskripts - Einfache Batchabläufe erstellen - Artikel Nr. 1

Ein Shellskript ist vergleichbar mit den unter MS-DOS bekannten »Batchdateien«, mit den eine Reihe von Befehlen zusammengefasst werden kann. Unter UNIX und Linux wird eine solche Reihe von Befehlen in einer einfachen Textdatei gespeichert. Diese muss das Ausführungsrecht besitzen, damit sie gestartet werden kann.

Die ausführbare Datei wird über die Reihenfolge der Verzeichnisse gesucht, die in der Umgebungsvariable PATH gespeichert sind (Nicht zuerst im aktuellen Verzeichnis !).

Um die Abläufe der Shellskripts im Fehlerfall zu testen, kann mit der Anweisung »set -x« bewirkt werden, dass alle Befehle als Echo auf dem Bildschirm noch einmal angezeigt werden, bevor sie ausgeführt werden.

Um Bildschirmausgaben besser lesen zu können, kann der Befehl sleep <Wert> zur Verzögerung verwendet werden.

Zuweisung von Werten zu Variablen

Die einfache Syntax »VARIABLE=WERT« führt zur lokalen Zuweisung, d.h. die Variablen haben nur innerhalb des Shellskripts Gültigkeit. Mit dem Befehl »export VARIABLE« kann die Variable auch von Befehlen und Programmen genutzt werden, die in diesem Shellskript aufgerufen werden. »unset VARIABLE« löscht den Inhalt der Variable wieder. Der Wert einer Variablen kann mit echo \$VARIABLE ausgegeben werden. Das Kommando »set« gibt eine Liste aller gesetzten Variablen mit ihren Werten aus.

Um eine Benutzereingabe einer Variablen zuzuordnen, kann der Befehl »read VARIABLE« eingesetzt werden. Die Benutzereingabe muss mit RETURN abgeschlossen werden.

Parameterübergabe

Die an ein Shellskript übergebenen Parameter sind in folgenden Systemvariablen enthalten.

| | |
|------|-----------------------------------|
| \$0 | Der Name des Shellskripts selbst. |
| \$\$ | Prozessnummer der Shell |
| \$1 | erster Parameter |
| \$2 | zweiter Parameter usw. |
| \$# | Anzahl der Parameter |
| \$* | alle Parameter zusammen |

Mit dem Befehl »shift« werden alle Parameter »aufgerückt«, d.h. \$1 entfällt, \$2 wird zu \$1 kopiert, \$3 zu \$2 usw.

Returnwerte abfragen / setzen

Der Returncode der letzten Anweisung ist in der Variablen »\$?« gespeichert. Hierbei wird der Wert 0 in der Regel mit einem positiven Ergebnis verknüpft.

Der Returncode des aktuellen Shellskripts wird als Parameter zum Befehl exit übergeben.

Beispiel:

```
exit -1  #Beendet das aktuelle Shellskript mit dem Wert -1
```

Einfache Bedingungsprüfung

Die einfache Entscheidung zwischen zwei Bearbeitungsfällen wird durch den if-Anweisungsblock realisiert.

Die allgemeine Form:

```
if <Bedingung>
then
    Anweisung1    #für den Fall, dass die Bedingung zutrifft
else
    Anweisung2    #für den Fall, dass die Bedingung nicht zutrifft
fi
```

Hinweise:

- Sind mehrere Fälle zu unterscheiden kann statt »else« auch »elif« gefolgt von »then« für einen neuen Anweisungsteil verwendet werden:

```
if <Bedingung1>
then
    Befehl1
elif <Bedingung2>
then
    Befehl2
else
    Befehl3
fi
```

- Als <Bedingung> sind Ergebnisse des Befehls »test« zu verwenden, der

in mehreren verschiedenen Syntaxvarianten einzusetzen ist. Im Manual des Befehls »test« ist eine vollständige Dokumentation aller Optionen vorhanden. Hier eine beschränkte Auswahl

| Option | Wirkung |
|-----------------|---|
| Wert1 -eq Wert2 | Ist wahr, wenn Wert1 = Wert2, nur für numerische Werte. Für nicht numerische Werte kann auch Wert1 == Wert2 verwendet werden. |
| Wert1 -ge Wert2 | Wert1 ist größer oder gleich Wert2 |
| Wert1 -le Wert2 | Wert1 ist kleiner oder gleich Wert2 |
| ! | Der NOT-Operator kehrt das Testergebnis um. |
| -f <Datei> | Ist wahr, wenn die angegebene Datei eine reguläre Datei ist. |
| -z "Wert1" | Ist wahr, wenn die Variable leer ist (für Zeichenketten). |
| -e <Datei> | Ist wahr, wenn die angegebene Datei eine Datei ist. |
| -d <Datei> | Überprüfung, ob die Datei ein Verzeichnis ist. |
| -r <Datei> | Überprüfung, ob ich Leserecht für die Datei habe. |
| -w <Datei> | Überprüfung, ob ich Schreibrecht für die Datei habe. |
| -h <Datei> | Überprüfung, ob die Datei ein symbolischer Link ist. |
| -s <Datei> | Überprüfung, ob die Größe der Datei mehr als Null ist. |
| -u <Datei> | Überprüfung, ob die Datei das bit SUID eingestellt hat. |
| -x <Datei> | Überprüfung, ob ich die Datei vorhanden und ausführbar ist. |

Möglich ist auch diese Syntax:

```
test -f <Datei>
```

oder

```
[ -f <Datei> ]
```

Die logische Grundverknüpfung OR entspricht "||" und AND "&&"

Beispiel:

```

if [ $# -eq 0 ] || [ $# -eq 1 ]
then
    #wenn die Anzahl der Parameter 0 oder 1 ist.
else
    #wenn die Bedingung nicht zutrifft.
fi

```

Auch möglich ist diese Syntax:

```

if [ $# -eq 0 -o $# -eq 1 ]
then
    #wenn die Anzahl der Parameter 0 oder 1 ist.
else
    #wenn die Bedingung nicht zutrifft
fi

```

Die Option -o bewirkt die OR-Verknüpfung, -a steht für die AND-Verknüpfung.

Mehrfache Bedingungen

Sind für eine Bedingung mehrere mögliche Fälle zu unterscheiden ist die Abfrage mit »if-Anweisungsblöcken« unübersichtlich. In diesem Fall wird die case-Anweisung verwendet. Allgemeine Syntax:

```

case <Variable> in
Fall1)
    Anweisung1
    ;;
Fall2)
    Anweisung2
    ;;
*)
    # die Fälle 1 und 2 treffen nicht zu
    ;;
esac

```

Hinweise:

Durch die »;;« wird das Ende eines Anweisungsteils gesetzt und durch esac wird das Ende der Case-Anweisung angezeigt.

Beispiel:

```

echo "Wollen Sie die Bearbeitung wirklich durchführen ? (J/N)"

```



```

read EINGABE
case $EINGABE in
J)
    echo "Die Verarbeitung wird durchgeführt"
    ;;
N)
    echo "Die Verarbeitung wird nicht durchgeführt"
    ;;
*)
    echo "Falsche Eingabe $EINGABE . Programmabbruch"
    exit -1
    ;;
esac

```

Anmerkung:

read -n 7 VAR ... Begrenzung der maximalen Zeichenanzahl die von read ausgewertet wird

read -s VAR ... die eingegebenen Zeichen werden am Bildschirm unterdrückt , d.h. sie werden nicht angezeigt (hilfreich z.B. bei Passworteingaben); s steht für Silent

Schleifen

Durch eine while-Schleife kann eine Folge von Anweisungen solange ausgeführt werden, bis eine bestimmte Bedingung erreicht wird.

Allgemeine Syntax:

```

while <Bedingung>
do
    Anweisung
done

```

Beispiel:

```

echo "Wollen Sie die Bearbeitung wirklich durchführen ? (J/N)"
read EINGABE
while [ ! "$EINGABE" == "J" -a ! "$EINGABE" == "N" ]
do
    echo "Falsche Eingabe, Geben Sie J oder N ein"
    read EINGABE
done

```

Durch eine for-Schleife kann eine Folge von Anweisungen mehrfach mit

verschiedenen Daten ausgeführt werden.

Allgemeine Syntax:

```
for <Variable > in <Werteliste>
do
  Anweisung
done
```

Beispiel:

```
for i in "datei1" "datei2" "datei3"
do
  if [ ! -f $i ]
  then
    echo "ACHTUNG, die Datei $i existiert nicht"
  fi
done
```

Mit den folgenden Befehlen wandeln Sie Leerzeichen im Dateinamen in allen MP3-Dateien des aktuellen Verzeichnisses in Unterstriche um:

```
for i in *.mp3;
do
  mv "$i" $(echo "$i" | tr " " _);
done
```

So wird z.B. aus Sendung über Linux.mp3 der neue Name Sendung_über_Linux.mp3.

siehe auch: man bash, tr --help, man tr

* * * * *

Einführung in die Shell-Programmierung - Artikel Nr. 2

Was ist ein Shell-Skript?

Ein Shell-Skript ist eine Textdatei bestehend aus

- Programm- und Funktionsaufrufen
- Shell internen Kontrollstrukturen
- Variablenzuweisungen

Das schreiben von Shell-Skripts verlangt Wissen über

- die Shell internen Kontrollstrukturen
- möglichst viele Unix/Linux-Programme und
- deren Optionen.

Wozu Shell-Skripts?

Shell-Skripts bieten die Möglichkeit schnell und einfach Aufgaben zu Automatisieren. Man kann eine Shell auch als eine Programmiersprache verwenden. Jedoch unterscheiden sich Shell-Skripts von Programmen in Sprachen wie C beträchtlich. Im Shell-Skript sind alle Systemprogramme direkt verwendbar. Es gibt nur einen Typ von Variablen. Diese müssen nicht initialisiert werden.

Hier ließe sich nun eine lange Liste von Unterschieden angeben, stattdessen sei nur Folgendes gesagt: Shell-Skripte eignen sich besonders gut um Aufgabe »quick and dirty« zu lösen. Ein Systemadministrator, der vor einer größeren Aufgabe steht - etwa der Erzeugung von 100 durchnummerierten Accounts, wird in der Regel versuchen dies mit einem Skript zu lösen. Wird die Aufgabe größer so gehen die Überlegungen in Richtung Perl und letztlich zu einer Compilersprache. Shell-Skripts sollte man nicht verwenden wenn

- die Geschwindigkeit des Programms eine große Rolle spielt
- komplizierte Aufgaben gelöst werden sollen, die ein stark strukturiertes Programmieren erfordern
- Sicherheitsmechanismen verwendet werden sollen.
- viele Dateien gelesen und geschrieben werden sollen
- man ein GUI (grafische Benutzeroberfläche) haben will - GUI's are for wimps anyway

Verschiedene Shells

sh ... Bourne shell, die erste Unix Shell von S. R. Bourne

bash ... Bourne Again Shell, Standard auf Linux, von der FSF

csh ... Berkeley Unix C Shell, C-ähnliche Syntax, Standard auf BSD

tcsh ... TENEX C Shell, erweiterte csh
ksh ... Korn Shell, proprietäre Shell von David Korn
pdksh ... Public Domain Korn Shell
rc ... Shell für Plan 9-OS, erweiterbare Shell auf Basis von rc
zsh ... Z Shell von Paul Falstad
ash ... kleine, POSIX konforme Shell, /bin/sh auf NetBSD
esh ... Easy Shell, kleine, leicht bedienbare Shell
kiss ... Karels Interactive Simple Shell
lsh ... Shell mit DOS Kommandos für Umsteiger
osh ... Operators Shell, mit erweiterten Sicherheitsmechanismen
sash ... Stand-alone Shell, braucht keine Libraries
psh ... Perl Shell, Perl Syntax

Die Shell unserer Wahl

Shells gibt es sehr viele. Die obige Aufstellung erhebt keinerlei Anspruch auf Vollständigkeit. Die Bourne Shell (sh), benannt nach ihrem Erfinder, ist die Mutter aller Shells auf Unix/Linux-Systemen. Die Bourne Shell bot schon zu Anfang fast alle Möglichkeiten der Programmierung, die die bash heute bietet. Die interaktive Bedienung war allerdings nicht sehr bequem. Dieser Umstand führte zur Geburt der csh. csh-Skripts folgen einer anderen Syntax, die der Programmiersprache C ähnlich ist, als sh-Skripts. Ein weiterer Meilenstein in der Geschichte der Shells war die Korn Shell ksh, ebenfalls nach ihrem Erfinder benannt. Diese war ein sehr erfolgreiches proprietäres Produkt. Die bash ist der Versuch einer Mischung der Vorzüge verschiedener Shells, wobei die Programmierung in einer erweiterten sh-Syntax erfolgt. Im weiteren werden wir unser Hauptaugenmerk auf die bash legen, die heute wohl die meist verwendete Shell ist. Die meisten unserer Konstruktionen werden allerdings ohne weiteres auch mit einer sh funktionieren.

Die Login Shell

Nach dem Login landet man auf Unix/Linux-Systemen in einer Shell. Welche Shells auf einem System als mögliche Login Shell installiert sind steht in der Datei /etc/shells.

```
cat /etc/shells
```

Die Login Shell eines jeden Benutzers steht in der Datei /etc/passwd.

```
grep $USER /etc/passwd | cut -d : -f 7
```

Ändern kann man seine Login Shell mit dem Befehl **chsh** (aka change

shell). Die Standard Login Shell auf Linux Systemen ist die bash.

Hello World

Das obligate Hello World Programm ist als Shell-Skript denkbar einfach. Folgender Text soll in der Datei helloworld gespeichert sein.

```
#!/bin/bash  
echo "hello world"
```

Ist es ein Shell-Skript?

file helloworld

Ausführbar machen mit

```
chmod 755 helloworld
```

und ausführen mit

```
./helloworld
```

Ein Shell-Skript schreiben

Ein Shell-Skript ist eine Textdatei, die Kommandos enthält. Welchen Texteditor man zur Erzeugung dieser Datei verwendet ist letztendlich egal, aber ...

Beginnt diese Textdatei mit der Zeichenfolge

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

und wird die Datei ausführbar gemacht

```
chmod 755 Dateiname
```

so kann das Skript direkt mit seinem Namen aufgerufen werden.

```
./scriptname
```

Das Skript wird dann Zeile für Zeile, vom Interpretor, in unserem Fall der bash, abgearbeitet. Im Unterschied dazu müssen Programme, die in Compilersprachen geschrieben sind, erst kompiliert werden, bevor sie ausgeführt werden können.

Wie in allen Programmiersprachen ist das Kommentieren eines Skripts ein oft ignoriertes Merkmal guten Programmierstils. Zeilen die mit # beginnen gelten als Kommentare und werden ignoriert.

Kommentare

Zeilen im Skript, die mit # beginnen werden als Kommentar gewertet und beim Ausführen ignoriert.

```
# Kommentare können helfen die  
# Lesbarkeit von Shell-Skripts  
# zu erhöhen.
```

Die einzige Ausnahme ist, wenn die ersten zwei Zeichen des Skripts #! lauten und unmittelbar nachher, der Pfad des Interpreters steht. Dies macht das Skript direkt ausführbar und gilt für beliebige Interpretersprachen.

- #!/bin/bash
- #!/bin/csh
- #!/bin/sh
- #!/usr/bin/perl

Variablen

Variablenamen bestehen aus Buchstaben, und Zahlen, wobei oft auf Kleinbuchstaben verzichtet wird. Die Zuweisung eines Wertes funktioniert folgendermaßen:

```
VARIABLENNAME=wert
```

Auf den Wert einer Variablen greift man mit \$VARIABLENNAME zu.

```
echo $VARIABLENNAME
```

unset nimmt einer Variable ihren Wert.

```
unset VARIABLENNAME
```

In Shell-Skripts gibt es nur einen Variablentyp. Man unterscheidet also nicht wie in anderen Programmiersprachen Integer-, Gleitkomma- und Stringvariablen. Wenn ein Skript davon abhängt, dass in einer Variable ein Integerwert steht, so ist der Autor dafür verantwortlich, dass dem auch so ist. Variablen müssen nicht initialisiert werden. Dies bereitet oft Probleme bei Tippfehlern. Folgendes Beispiel führt zu keiner Fehlermeldung:

```
TEST=Legasthenie
echo $TSET
```

Der Befehl **set** gibt eine Liste, der von der bash gesetzten Variablen. Unter Anderen befindet sich darunter die Variable **PS1** die das Aussehen des Prompts bestimmt. MS-DOS Nostalgiker könnten sich über folgendes freuen:

```
PS1="C:\> "
C:\>
```

In Skripten sind vor allem die Variablen **\$1-\$9** von großer Bedeutung. Diese enthalten die an das Skript übergebenen Parameter. Dazu folgendes Beispiel, das in der ausführbaren Datei parameters gespeichert sein soll:

```
#!/bin/bash
# Ausgabe des ersten Parameters
echo "Erster Parameter: $1"

# Ausgabe des zweiten Parameters
echo "Zweiter Parameter: $2"
```

Und das kommt dabei heraus:

```
./parameters foo bar
Erster Parameter: foo
Zweiter Parameter: bar
```

Variablen - kurze Zusammenfassung

- Variablennamen bestehen aus Buchstaben, und Ziffern.
- Zuweisung erfolgt über VARIABLENNAME=wert
- auf den Wert zugreifen mit echo \$VARIABLENNAME
- unterschiedliche Variablenarten mit Beispielen
 - selbst definierte Variablen: \$FOO, \$BAR, ...
 - systemweite Variablen: \$HOSTNAME, \$HOSTTYPE, ...
 - built-in Variablen: \$1-\$9, \$PS1, \$PATH, ...
- Variablen exportieren
export VARIABLENNAME

Subshells, Variablen exportieren

Ein grundsätzliches Konzept des Unix-Prozessmanagements ist, dass jeder Prozess von einem Parentprozess (Elternprozess) abstammt. Dieser Text entsteht in einer Instanz des Editors **vim**. Ein Auszug aus der Ausgabe des

Kommandos **ptree** soll dieses Konzept verdeutlichen:

```
ptree
init+-arptatch
[...]|
|-xdm+-XF86_SVGA
|    |-xdm---fvwm2+-FvwmCommandS
|    |               |-xterm---bash---vim
[...]|               [...]
```

Der Prozess init ist »Ahne« aller Prozesse. Der Login Manager xdm wartet auf Benutzerauthentifizierung und startet einen Windowmanager fvwm2, aus dem ein xterm gestartet wurde in dem eine bash läuft, aus der ein vim gestartet wurde. Eine Subshell ist eine Shell, die aus einer anderen gestartet wurde. Ruft man ein Shell-Skript via

```
./script oder
bash script
```

auf, so wird es in einer Subshell gestartet. Ruft man es via

```
. script oder
source script
```

auf, so werden die darin enthaltenen Befehle in der aktuellen Shell ausgeführt. Dies entspricht einem **include-Befehl** wie unter der Programmiersprache C oder der Interpretersprache PHP. Besonders beachten sollte man, dass die Verwendung von Pipelines auch innerhalb eines Shell-Skripts zu Subshells führt.

Dies ist wichtig weil, Variablenzuweisungen auf eine bash Instanz beschränkt sind. Die Zuweisung kann in alle Subshells mit dem Befehl **export** exportiert werden. Folgende Kommandoabfolgen sollen das verdeutlichen, wobei man sich durch **ptree -h** einen Überblick über die aktuelle Situation machen kann.

```
# Test wird nicht exportiert
TEST=rose
echo $TEST
rose
bash
echo $TEST

TEST=eros
```



```

echo $TEST
eros
exit
exit
echo $TEST
rose
unset TEST
exit

# Test wird exportiert
TEST=rose
echo $TEST
rose
export TEST
bash
echo $TEST
rose
TEST=eros
echo $TEST
eros
exit
exit
echo $TEST
rose
unset TEST
exit

```

Setzt man also in der Shell eine Variable und startet dann ein Shell-Skript, so ist diese Variable - so sie nicht exportiert wurde - im Skript nicht gesetzt.

Subshells - kurze Zusammenfassung

Aus einer Shell kann man eine weitere Shell starten, welche dann als Subshell der ursprünglichen Shell bezeichnet wird.

```
bash
```

Die im Skript enthaltenen Kommandos werden in

- einer Subshell ausgeführt, wenn das Skript mit

```

./script [&] oder
bash script [&]

```

- in der aktuellen Shell ausgeführt, wenn das Skript mit

```
. script oder
source script
```

gestartet wird, wobei das optionale **&** das Skript im Hintergrund startet. Dies entspricht einem include-Befehl (. script, source script) wie unter der Programmiersprache C oder der Interpretersprache PHP.

Spezielle Zeichen und Quoting

Leerzeichen <space> und Tabulatoren <tab> werden als Trennzeichen verwendet. So werden im folgenden Beispiel das Kommando, die Optionen und die Parameter eines Befehls durch Leerzeichen getrennt:

```
ls -l foo bar
```

Manchmal ist es erwünscht, einem Zeichen mit spezieller Bedeutung diese zu nehmen. So kann ein Dateiname auch ein Leerzeichen enthalten. Seit die Windows-Betriebssysteme über die 8:3 Namenskonvention hinausgekommen sind, tritt dieses Phänomen leider gehäuft auf. Als Beispiel nehmen wir eine Datei mit dem Namen Mein Lied.mp3 an. Versucht man diese in der Shell zu löschen und schreibt:

```
rm Mein Lied.mp3
```

so versucht die Shell zwei Dateien mit Namen **Mein** beziehungsweise **Lied.mp3** zu löschen. Um der Shell mitzuteilen, dass das Leerzeichen in diesem Fall keine spezielle Bedeutung hat verwendet man eine der drei Quoting Methoden. Diese seien im Folgenden anhand unseres Beispiels angeführt:

```
rm Mein\ Lied.mp3
rm 'Mein Lied.mp3'
rm "Mein Lied.mp3"
```

Der Backslash \ (escape character) hebt dabei die spezielle Bedeutung des unmittelbar folgenden Zeichens auf. Auch wenn dieses Zeichen ein »newline«, also das Resultat des Betätigens der Return Taste, ist. Sowohl einfache " (single quotes) als auch doppelte Anführungszeichen "" (double quotes) heben die spezielle Bedeutung der Zeichen dazwischen auf, wobei dies bei den einfachen Anführungszeichen für alle Zeichen gilt. Bei doppelten Anführungszeichen bleibt die spezielle Bedeutung der Zeichen \$,

' und \ erhalten. Vergleiche die Ausgaben folgender Kommandos:

```
TEST=rose
echo "In $TEST steht \"$TEST\""
In $TEST steht "rose"
echo 'In $TEST steht \"$TEST\"'
In $TEST steht \"$TEST\"
```

Im weiteren werden wir noch des öfteren auf spezielle Zeichen stoßen.

Quoting - kurze Zusammenfassung

Quoting wird dazu benutzt, um bestimmten Zeichen ihre spezielle Bedeutung zu nehmen. Es gibt drei Möglichkeiten um zu quoten.

```
\ ... escape character
" ... single quotes
"\" ... double quotes
```

Hat ein geistreicher Benutzer eine Datei unter »Mein Lied * Napster.mp3« abgespeichert, so kann man diese mit einer der folgenden Zeilen löschen.

```
rm Mein\ Lied\ *\ Napster.mp3
rm "Mein Lied * Napster.mp3"
```

Der escape character nimmt dem folgenden Zeichen seine spezielle Bedeutung.

Alle Zeichen zwischen single quotes sind von ihrer speziellen Bedeutung befreit.

Double quotes lassen den Zeichen \$, ' und \ ihre spezielle Bedeutung.

Double quotes sind also schwächer als single quotes.

Befehle zu Gruppen zusammenfassen

Befehle werden durch einen Zeilenumbruch <newline> oder ein Semikolon (;) getrennt. Im folgenden zwei Schreibweisen, die das gleiche liefern:

```
cd /usr/bin
ls
```

bzw.

```
cd /usr/bin; ls
```

oder

```
FOO=bar; echo "\$FOO ist $FOO"; echo "\$FOO ist $FOO"
```

Ausgabe:

```
$FOO ist bar
```

```
$FOO ist bar
```

Es gibt zwei Möglichkeiten Kommandos zu Gruppieren. Runde Klammern () (parantheses) beziehungsweise geschwungene oder geschweifte Klammern {} (curly braces). Befehle die in runden Klammern gruppiert sind, werden in einer Subshell ausgeführt, während in geschwungenen Klammern gruppierte Befehle in der aktuellen Shell ausgeführt werden. Dazu folgende Beispiele:

```
{ FOO=bar; echo "\$FOO ist $FOO"; }; echo "\$FOO ist $FOO"
```

Ausgabe:

```
$FOO ist bar
```

```
$FOO ist bar
```

```
( FOO=bar; echo "\$FOO ist $FOO" ); echo "\$FOO ist $FOO"
```

Ausgabe:

```
$FOO ist bar
```

```
$FOO ist bar
```

Die runden und die geschwungenen oder geschweiften Klammern zählen zu den speziellen Zeichen. Diese müssen also, so sie in Dateinamen bzw. Strings auftreten, mit Quotes versehen werden.

Kommandos Gruppieren - kurze Zusammenfassung

; ... Trennt verschiedene Kommandos, die in der gleichen Zeile geschrieben werden. Die Kommandos werden hintereinander ausgeführt

() ... Ein oder mehrere Kommandos innerhalb runder Klammern werden in einer Subshell ausgeführt.

{ } ... Ein oder mehrere Kommandos innerhalb geschwungener Klammern werden als Block in der aktuellen Shell ausgeführt.

Wild Cards und Pattern Matching

Gehen wir im Folgenden davon aus, dass im aktuellen Verzeichnis folgende Dateien existieren:

```
ls
```

```
glut gut hut mut mutter
```

* steht für beliebige Zeichenketten; auch leere.

ls mut*
mut mutter

? steht für genau ein beliebiges Zeichen.
ls ?ut
gut hut mut

Alle in eckigen Klammern angegebenen Zeichen dürfen anstelle der eckigen Klammern vorkommen.
ls [gh]ut
gut hut

Ein ! oder ^ verneint diese Auswahl.
ls [!gh]ut
mut

Alle in geschwungenen Klammern angegebenen Zeichenketten, die durch Kommata getrennt sind, dürfen vorkommen.
ls {gl,m}ut
glut mut

Diese Konstrukte sind auch kombinierbar.
ls {gl,m}ut*
glut mut mutter

Wildcards und Pattern Matching

Folgende Zeichen haben in der Bash spezielle Bedeutung

* ... beliebige Zeichenkette
? ... genau ein beliebiges Zeichen
[] ... genau eines der genannten Zeichen
[^] ... genau ein nicht genanntes Zeichen oder [!]
{ , } ... genau eine der Zeichenketten

Exit Status - kurze Zusammenfassung

Jedes Kommando liefert einen Exit Status zurück. Der Exit Status ist ein Integerwert zwischen 0 und 255. 0 bezeichnet den Erfolg des Kommandos, alle anderen Werte bezeichnen verschiedene Ausmaße des Scheiterns. Der Exit Status des letzten Kommandos steht in der Variable \$?.

```
grep root /etc/passwd  
root:x:0:0:root:/root:/bin/bash  
echo $?
```

0

```
grep Administrator /etc/passwd  
echo $?  
1
```

```
true  
echo $?  
0
```

```
false  
echo $?  
1
```

Die Programme **true** beziehungsweise **false** dienen einzig und allein dazu Exit Status 0 beziehungsweise 1 zu liefern. Das scheint auf den ersten Blick sinnlos, aber das tut auch /dev/null.

Um den Exit Status eines Shellskripts zu beeinflussen gibt es den built-in Befehl **exit**. Dieser beendet ein Skript sofort und setzt einen anzugebenden Wert als Exit Status des Skripts. Das Skript namens parameterexists liefert Exit Status 0, wenn mindestens ein Parameter übergeben wurde. Andernfalls liefert es Exit Status 1.

```
#!/bin/bash  
if [ $1 ]; then  
    exit 0  
else  
    exit 1  
fi
```

Man sollte dabei unbedingt die Konvention beachten, dass 0 für einen Erfolg steht, während andere Zahlen als Misserfolg interpretiert werden.

Exit Status

Jedes Programm liefert einen Exit Status zurück.

0 steht für eine erfolgreiche Beendigung des Programms
1-255 bezeichnet verschiedene Ausmaße des Scheiterns.

Der Exit Status eines Shellskripts kann über den bash built-in Befehl **exit** gesteuert werden. Der Exit Status des letzten ausgeführten Kommandos steht in der Variable \$?.

Flow Control

Unter Flow Control versteht man Kontrollstrukturen wie bedingtes Ausführen von Kommandos und Schleifen. Im Wesentlichen sind dies **if**, **case**, **for** und **while**. Wir werden diese im Weiteren nacheinander Abhandeln, wobei wir nebenbei allerlei Nützliches mitnehmen werden. Zuerst wenden wir uns dem **if** zu, von dem wir schon ein Beispiel sahen. Im folgenden Syntax Überblick wird bei Zutreffen von Bedingung1 die Anweisung1 ausgeführt. Trifft Bedingung1 nicht, Bedingung2 aber schon zu, so wird Anweisung2 ausgeführt. Trifft keine der beiden Bedingungen zu, so wird Anweisung3 ausgeführt.

Flow Control, if

Die **if**-Anweisung erlaubt eine bedingte Ausführung von Kommandos und ist dem **if** in Programmiersprachen sehr ähnlich. Es verlangt nach folgender Syntax:

```
if Bedingung1
then
    Anweisung1
elif Bedingung2
then
    Anweisung2
else
    Anweisung3
fi
```

wobei sowohl der **elif**, als auch der **else**-Block optional sind. Die Bedingung ist ein Kommando, das 0 (true) oder != 0 (false) als Exit Status zurückgibt.

Bedingungen

Um ein **if** formulieren zu können, müssen wir wissen, was eine Bedingung ist. Eine Bedingung ist nichts anderes als ein Kommando. Ist dessen Exit Status 0, so gilt die Bedingung als erfüllt. Andernfalls gilt sie als nicht erfüllt. Das erklärt auch, warum man den Konventionen des Exit Status in eigenen Skripts folgen sollte.

Ein in Bezug auf Bedingungen besonders wertvoller Befehl ist der built-in Befehl **test**. Will man zum Beispiel feststellen, ob der Inhalt der Variable **TEST** gleich »rose« ist, so kann man das so überprüfen:

```
TEST=rose
test $TEST = rose
echo $?
0
```

```

TEST=eros
test $TEST = rose
echo $?
1

```

Als Synonym für **test** können auch eckige Klammern `[]` verwendet werden, wobei darauf zu achten ist, dass nach der öffnenden und vor der schließenden Klammer Leerzeichen stehen.

```

TEST=rose
[ $TEST = rose ]
echo $?
0

```

Bedingungen (Strings, Dateien)

Das ist eine unvollständige Liste von Tests mit Strings (Zeichenketten, Text, Zahlen etc.) und Dateien. Im folgender Liste bezeichnen S1 beziehungsweise S2 Strings und D1 beziehungsweise D2 Dateinamen.

| Test | Wahr wenn |
|---------------|------------------------|
| [S1 = S2] | Strings ident |
| [S1 != S2] | Strings nicht ident |
| ... | ... |
| [-e D1] | Datei D1 existiert |
| [-d D1] | D1 ist ein Verzeichnis |
| [-x D1] | D1 ist ausführbar |
| [D1 -nt D2] | D1 neuer als D2 |
| ... | ... |

Bedingungen (Integers)

Die folgenden Tests gehen davon aus, dass in den Variablen A und B ganze Zahlen stehen. Ist dies nicht so, so gibt es eine Fehlermeldung.

| Test | Wahr wenn |
|-----------------|---------------------|
| [\$A -lt \$B] | \$A kleiner als \$B |
| [\$A -gt \$B] | \$A größer als \$B |

| Test | Wahr wenn |
|-----------------|------------------------|
| [\$A -le \$B] | \$A kleiner gleich \$B |
| [\$A -ge \$B] | \$A größer gleich \$B |
| [\$A -eq \$B] | \$A gleich \$B |
| [\$A -ne \$B] | \$A ungleich \$B |

Arithmetik

Neben der Möglichkeit ganze Zahlen gegeneinander zu testen, kann man in der Shell mit ganzen Zahlen auch Rechnen. Ausdrücke die mit `$[]` oder `$(())` umschlossen sind werden als ganzzahlige Rechenoperationen ausgelegt und auszuwerten versucht.

```
echo "1+1=$[1+1]"
1+1=2
echo "3*3=$((3*3))"
3*3=9
echo "11/4=$[11/4] mit Rest $[11%4]"
11/4=2 mit Rest 3
```

In der Programmierung ist es häufig gefragt, den Inhalt einer Variable zu inkrementieren, sprich um 1 zu erhöhen. Das sieht in C definitiv schöner aus, als in der bash:

```
N=1
echo $[N+1]
2
```

Um in der Shell oder innerhalb von Skripts Gleitkommarechnungen oder allgemein kompliziertere Rechnungen durchzuführen sei dem Leser die Terminalprogramme `bc` und `dc` ans Herz gelegt.

Arithmetik - kurze Zusammenfassung

Integerarithmetik wird innerhalb eckiger oder doppelter runder Klammern ausgewertet.

```
echo $[1+1]
echo $((1-1))
```

Eine unvollständige Liste der existierenden Operatoren:

| Operator | Bedeutung |
|----------|----------------|
| + | Plus |
| - | Minus |
| * | Multiplikation |
| / | Division |
| % | Modulo |
| ** | Exponent |

Das Inkrementieren einer Variable funktioniert folglich so:

```
N=1
N=$((N+1))
echo $N
```

Bedingungen verknüpfen

Neben dem if gibt es noch eine einfachere, eingeschränkte Möglichkeit des bedingten Ausführens von Kommandos. So wird im Folgenden der Befehl **ls** nur ausgeführt, wenn die Programmdatei /bin/ls existiert und ausführbar ist:

```
[ -x /bin/ls ] && ls
helloworld parameterexists parameters
```

Der Exit Status solch einer Kette von Kommandos ist der Exit Status des letzten ausgeführten Kommandos. Ein weiteres Beispiel legt eine Datei foo nur an, wenn diese noch nicht existiert:

```
touch foo
ls
foo

[ -e foo ] || touch foo
echo $?
0
[ ! -e foo ] && touch foo
echo $?
1
```

Bedingtes Ausführen

In folgender Zeile wird Kommando2 ausgeführt, wenn Kommando1 Exit

Status 0 hat.

Kommando1 && Kommando2

Während in folgender Zeile Kommando2 ausgeführt wird, wenn Kommando1 Exit Status != 0 hat.

Kommando1 || Kommando2

Der Exit Status dieser Zeilen ist der Exit Status des letzten ausgeführten Kommandos. Ein ! vor einem Kommando verneint den Exit Status.

| Operator | Bedeutung |
|----------|-----------|
| && | Und |
| | Oder |
| ! | Nicht |

Dies kann auch dazu benutzt werden verknüpfte Bedingungen in einer if-Anweisung zu erstellen.

Flow Control, case

Eine spezielle und für manche Anwendungen sehr angenehme Variante der if-Anweisung ist die case-Anweisung, die nach folgender Syntax verlangt.

```
case Ausdruck in
Pattern1 )
    Anweisungen ;;
Pattern2 )
    Anweisungen ;;
...
esac
```

Das bietet sich zum Beispiel dazu an, um Switches zu realisieren. Wie bei init-Skripts üblich erlaubt folgendes Skript eine »start« und »stop« Option:

```
case $1 in
start) ...;;
stop) ...;;
*) Usage: ...;;
esac
```

Die case-Anweisung wird oft in init-Skripts verwendet. Diese Skripts dienen zum starten und stoppen von Systemdiensten. Üblicherweise liegt für jeden Systemdienst im Verzeichnis /etc/init.d/ (manchmal auch /etc/rc.d/init.d oder /etc/rc.d/) ein Skript, das die Parametern start, stop oder restart versteht.

cat /etc/init.d/inetd

```
#!/bin/sh bzw. #!/bin/bash
#
# start/stop inetd super server.
[...]
case "$1" in
start)
    echo -n "Starting internet superserver:"
    [...]
;;
stop)
    echo -n "Stopping internet superserver:"
    [...]
;;
restart)
    echo -n "Restarting internet superserver:"
    [...]
;;
*)
    echo "Usage: /etc/init.d/inetd {start|stop|restart}"
    exit 1
;;
esac
```

Gibt man einen Parameter an, der nicht angeführt ist, so wird der Punkt unter *) ausgeführt. Dieser klärt über die richtige Benutzung des Skripts auf und beendet das Skript mit Exit Status 1. Die for-Schleife unterscheidet sich stark von for-Schleifen anderer Programmiersprachen. In ihr durchläuft ein Variable alle Werte einer Liste. Die Einträge dieser Liste sind durch Leerzeichen, Tabulatoren oder Zeilenumbruch getrennt. Alle Frauen sind herzlichst dazu eingeladen das Skript polygam nach ihren Bedürfnissen zu modifizieren.

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

```
LISTE="Liese Ingrid Stefanie Frauke"
```

```
for FRAU in $LISTE; do
echo "$FRAU ist meine Frau."
done
```

Flow Control, for

Die for-Schleife der bash ist in ihrer Syntax substantiell verschieden zu vielen Programmiersprachen.

```
for Name in Liste
do
Anweisungen
done
```

Der Inhalt der Variable Name durchläuft dabei alle Elemente der Liste Liste. So spielt der folgende Code alle mp3-Dateien im aktuellen Verzeichnis ab.

```
for LIED in *.mp3
do
mpg123 $LIED
done
```

mpg123 ist dabei der mp3-Player. *.mp3 wird von der Bash zu einer Liste aller Dateien mit der Endung .mp3 expandiert.

Command Substitution

Oft will man an einer bestimmten Stelle in einem Skript die Ausgabe eines Kommandos platzieren. Das nennt man Command Substitution und lässt sich auf zwei Arten realisieren. Das Kommando wird entweder mit Backticks `` oder mit \$() umschlossen. Das Skript summerton ist der Telefonzeitansage nachempfunden:

```
#!/bin/bash

# Zeitausgabe ala Zeitansage per Telefon

# Warten bis zu den naechsten vollen 10 Sekunden
while [ `date +%S%10` -ne 0 ]; do
sleep 1
done

# Alle 10 Sekunden Zeitansage
while true; do
```

```
echo "Es wird mit dem Summertonten `date +%k` Uhr, `date +%M`
Minuten und `date +%S` Sekunden."
sleep 10
done
```

Die Frage ob man `` oder \$() verwendet ist nicht ganz irrelevant. Die Verwendung von \$() hat den Vorteil, dass man diese Konstruktion ineinander Verschachteln kann. Der Nachteil ist, dass diese Konstruktion in der bash, nicht jedoch in der sh existiert. Versucht man ein Skript, das \$() verwendet auf einem System ohne einer bash zu verwenden, so wird dies scheitern.

Das übliche For I

Um das in anderen Programmiersprachen übliche Verhalten einer for-Schleife zu erhalten kann folgender Code in usualfor1 dienen:

```
#!/bin/bash

# Initiiert ein for wie in C, wobei dies einem
# for(int i=$START, i<=$STOP, i=i+$STEP) entspricht

START=1; STOP=10; STEP=3
for N in `seq $START $STEP $STOP`
do
echo $N
done
```

\$N durchläuft alle Werte von \$START bis \$STOP in Abständen von \$STEP.

Zur Command Substitution werden hier Backticks (``) verwendet, was die Portabilität des Codes erhöht. In den meisten Fällen kann auf eine einfachere Syntax von **seq** zurückgegriffen werden. Siehe dazu

```
man seq
```

Flow Control, while, until

Die while- bzw. until-Schleife entspricht ihren Entsprechungen in anderen Programmiersprachen. Ein Anweisungsblock wird, solange eine Bedingung wahr bzw. nicht wahr ist, ausgeführt.

```
while Bedingung
do
Anweisungen
```

done

Als Beispiel ein Codesegment, das alle 5 Sekunden schaut ob eine Modemverbindung existiert und falls diese zusammengebrochen ist ein Funktion namens reconnect() ausführt.

```
while ifconfig | grep ppp0
do
sleep 5
done
reconnect()
```

Das übliche For II

Eine weitere Methode das in anderen Programmiersprachen übliche for zu erhalten ist die folgende Umschreibung in usualfor2, die natürlich nicht Shell spezifisch ist:

```
#!/bin/bash

# umschreibt ein for wie in C
# durch ein while

START=1; STOP=10; STEP=3

N=$START
while [ $N -le $STOP ]; do
echo $N
N=$((N+STEP))
done
```

Auch hier durchläuft \$N alle Werte von \$START bis \$STOP in Abständen von \$STEP.

siehe auch: man bash, /etc/init.d/*, ~/.bashrc

* * * * *

Einführung in die Shell-Programmierung - Artikel Nr. 3

Dieses Kapitel widmet sich der Einführung in die Shell-Programmierung. Hierbei handelt es sich keinesfalls um eine vollständige Dokumentation. Für weiterführende Details empfehle ich dringend einen Blick in die Manpages der jeweiligen Programme sowie der Bash (man bash) zu werfen.

Allgemein

Was ist die Shell? Die Shell ist ein Interpreter, der Kommandos entsprechend seiner eigenen Syntax interaktiv oder selbständig ausführt. Was im Folgenden als Shell bezeichnet wird, ist ein Metaprogramm dessen Hauptaufgabe es ist, weitere Programme zu laden. Die Shell stellt die Schnittstelle zwischen Benutzer und Betriebssystem dar. In der UNIX-Welt hat sie den Status eines Benutzerprogramms und kann deshalb nach Belieben ausgetauscht werden.

Im folgenden die wichtigsten Vertreter:

| | | |
|-------|-----------------|--|
| sh: | Bourne Shell | die Mutter aller Shells |
| csh: | C-Shell | Shell mit C-ähnlicher Syntax |
| ksh: | Korn Shell | mächtige, C-orientierte Shell (Solaris) |
| zsh: | Z-Shell | erweiterte, komfortable Shell, Bash kompatibel |
| bash: | Bourne Again SH | erweiterte, komfortable Bourne Shell |

In diesem Kapitel wollen wir uns mit der Bash beschäftigen. Sie ist der Standard Kommandointerpreter unter LINUX. Die Bash ist kompatibel zur Standard-Bourne-Shell welche von Steven R. Bourne für AT&T Unix entwickelt wurde.

Erste Schritte

Im Grunde ist ein Shellskript nichts anderes als eine Textdatei, in der Befehlsfolgen enthalten sind. Diese Befehlsfolgen können mit Hilfe von Schleifen und Variablen gesteuert werden. Man kann solche Befehlsfolgen auch direkt in der Shell eingeben. Denkbar wären zum Beispiel Folgendes:

```
echo "Hallo Europa";echo "Hallo Osterhase..."  
Hallo Europa  
Hallo Osterhase...
```

Oder aber man schreibt ein kleines Skript, dass man dann jederzeit wieder aufrufen kann. Zu diesem Zwecke öffne man einen Editor seiner Wahl (vi,

emacs, mcedit, Kwrite, kate ...) und gebe folgendes ein:

```
#!/bin/sh
# Das ist ein Kommentar
echo "Hallo Europa"
echo "Hallo Osterhase..."
```

Danach die Datei unter einem selbst gewählten Namen abspeichern und ausführbar machen. Anschließend kann das Skript gestartet werden.

```
chmod 744 meine_datei.sh
./meine_datei.sh
Hallo
Hallo Osterhase...
```

Eine Shell-Skript beginnt mit der Angabe des Kommandointerpreters. Zeile 1 ist also der Pfad zu dem Programm, das die folgenden Zeilen interpretieren kann. Hier wird auf die Bourne Shell (sh) verwiesen. Die Datei /bin/sh ist in unserem Fall ein Symlink auf die Bash. Kommentare werden zeilenweise mit # gekennzeichnet. Das Shellkommando echo existiert in zwei Formen. Zum einen ist es ein in der Shell enthaltendes Kommando, zum anderen ist es aber auch eine Datei, welche sich im Verzeichnis /bin befindet. Linux bietet eine Vielzahl von Tools und Kommandos, die das Arbeiten auf der Textkonsole ermöglichen und erleichtern. Doch dazu später mehr.

Variablen und Quoting

Wie in jeder Programmiersprache gibt es auch in der Shell Variablen. Die Bash unterscheidet bei Variablen nicht nach Typen. Grundsätzlich wird jede Variable erst einmal als String aufgefasst. Je nach Kontext kann sie dann auch als Integer interpretiert werden.

Hier nun einige Beispiele, die den Umgang erläutern sollen:

```
#!/bin/sh
# Wertzuweisung
Variable_1=10
Variable_2="Der Mond ist ein grüner Käse."
Variable_3="A B C D"
Variable_4=$(hostname)
# alte Schreibweise: Variable_4=`hostname`
# Wertabfrage
echo \${Variable_1} = ${Variable_1}
```

```

echo "$Variable_2 = $Variable_2"
echo '$Variable_1 + $Variable_2' = ${Variable_1}${Variable_2}
echo $Variable_3
echo "$Variable_3"
echo $Variable_4
Ergibt:
$Variable_1 = 10
$Variable_2 = Der Mond ist ein grüner Käse.
$Variable_1 + $Variable_2 = 10Der Mond ist ein grüner Käse.
A B C D
A B C D

```

Um \$ \ " auf dem Bildschirm darzustellen zu können, müssen sie wie folgt maskiert werden: \\$ \ \ ". In Zeile 6 wird die Shell angewiesen, erst den rechten Teil der Variablenzuweisung auszuführen, nämlich den Befehl hostname. Dies wird durch \$(...) oder "Backquotes" (Zeile 7) erreicht. Die Variable wird also mit dem Hostnamen des jeweiligen Rechners belegt. Zu lange Befehlszeilen können mit einem \ getrennt werden.

```

#!/bin/sh
echo "Dies ist ein wahnsinnig langer, sinnfreier und \
unglaublich überflüssiger Satz."

```

Spezielle Typen von Variablen

local variables

Diese sind nur innerhalb eines Code Blocks ({...}) oder einer Funktion gültig und werden mit local Variable definiert.

environmental variables

Sie bestimmen das Erscheinungsbild der Shell und dienen der Anpassung ihres Verhaltens in der Systemumgebung. Die Prozessumgebung wird vom Elternprozess an seine Kinder vererbt. D.h. alle Umgebungsvariablen der Bash werden an die von ihr gestarteten Prozesse weitergegeben. Die Kommandos **printenv** und **export** geben alle Umgebungsvariablen aus. Mit dem **set** Kommando können Umgebungsvariablen gesetzt werden.

Quotierung

Quotierung wird benutzt, um die spezielle Bedeutung von Kontrollzeichen, reservierten Wörtern und Namen auszuschalten. Es gibt 3 Formen:

Fluchtsymbol \

Es entwertet das unmittelbar folgende Sonderzeichen. Ein durch \

entwertetes Zeilenende wird ignoriert.

Hochkomma ' (Quote)

Von Hochkomma eingeschlossene Worte werden von der Shell nicht weiter bearbeitet. Allerdings darf ein Hochkomma nicht in Hochkommas eingeschlossen sein. Auch nicht, wenn es durch \ maskiert ist.

Anführungszeichen " (Doublequotes)

Von in Anführungszeichen eingeschlossenen Wörtern erkennt die Shell nur die Sonderzeichen \$ '\ als solche. Das Fluchtsymbol behält seine Bedeutung für die Zeichen \$ ' " \ oder dem Zeilenende.

Arrays

Die Bash unterstützt auch eindimensionale Arrays. Diese können mit `declare -a` Variable initialisiert werden, müssen aber nicht. Einzelne Elemente des Arrays werden mit `${variable[xx]}` angesprochen.

```
#!/bin/bash
array=( zero one two three four five )
array[6]="Dieser Text ist ein Element des Arrays"
echo ${array[0]}      # zero
echo ${array[1]}      # one
echo ${array:0}       # zero
                        # Parametererweiterung, erstes Element.
                        # Start an Position #0 (1. Buchstabe).
echo ${array:1}       # ero
                        # Parametererweiterung, erstes Element,
                        # Start an Position #1 (2. Buchstabe).
echo ${array[1]:1}    # ne
                        # Parametererweiterung, zweites Element,
                        # Start an Position #1 (2. Buchstabe).
echo ${#array[2]}     # 3
                        # Länge des dritten Elements.
element_count=${#array[@]} # oder
element_count=${#array[*]} # Anzahl der Elemente: 7
```

Kommandosubstitution

Die Kommandosubstitution erlaubt es, die Ausgabe eines Kommandos direkt an eine Variable zu übergeben. Zwei Formen sind möglich:

`$(Kommando)` oder ``Kommando``

Bei der Substitution mit Backquotes (nicht Hochkomma!) müssen Sonderzeichen maskiert werden. Bei der Klammervariante bleiben alle Zeichen unverändert.

Damit lässt sich z.B. der Inhalt eines Textfiles in ein Array laden.

```
#!/bin/bash
filename=sample_file
# cat sample_file
#
# 1 a b c
# 2 d e fg
declare -a array1
array1=$(cat "$filename" | tr '\n' ' ')
# Loads contents
# of $filename into array1.
# list file to stdout.
# change linefeeds in file to spaces.
echo ${array1[@]}
# List the array:
# 1 a b c 2 d e fg
#
# Each whitespace-separated "word" in the file
#+ has been assigned to an element of the array.
element_count=${#array1[*]}
echo $element_count      # 8
```

Klammererweiterung

Die Klammererweiterung erzeugt aus einer in geschweiften Klammern eingeschlossenen Liste von Bausteinen mehrere Zeichenketten. Zum Beispiel erzeugt der Befehl:

```
mkdir ~/{dir1,dir2}
die Verzeichnisse dir1 und dir2 im Home-Verzeichnis.
```

und der Befehl

```
mkdir ~/{dir1,dir2}{1,2,3}
die Verzeichnisse dir11, dir12, dir13, dir21, dir22 und dir23.
```

Parameter

Positionsparameter

Einem Shellskript können beim Aufruf auch Parameter mitgegeben werden.

COMMAND Parameter1 Parameter2 ...

Diese Parameter lassen sich im Skript mit \$1, \$2, ... abfragen. Ab dem 10. übergebenen Parameter müssen geschweifte Klammern gesetzt werden ({10}).

Mit dem shift-Kommando ist es möglich, die Positionsparameter nach links zu verschieben. Das bedeutet, dass der zweite Parameter der Erste wird, der Dritte der Zweite usw.. \$0 (der Skriptname) bleibt unberührt. Dieses Kommando macht zum Beispiel bei Funktionen Sinn:

```
#!/bin/bash
multiply ()    # multipliziert die übergebenen Parameter
{
    # Anzahl der Parameter ist variabel
    local product=1
    until [ -z "$1" ] # Until nutzt den ersten übergebenen
Parameter ...
    do
        let "product *= $1"
    shift
    done
    echo $product # wird nicht auf STDOUT ausgegeben,
}                # wenn es an eine Variable übergeben wird
mult1=15383; mult2=25211
val1=`multiply $mult1 $mult2`
echo "$mult1 X $mult2 = $val1"    # 387820813
```

Spezialparameter

| | |
|------|---|
| \$* | Bezeichnet alle Positionsparameter von 1 an. In Anführungszeichen gesetzt, steht "\$*" für ein einziges Wort, bestehend aus dem Inhalt aller Positionsparameter, mit dem ersten »internen Feldseparator« (meistens Leerzeichen, Tab und Zeilenende) als Trennzeichen. |
| \$@ | bezeichnet alle Positionsparameter von 1 an. In Anführungszeichen gesetzt, wird es durch die Werte der einzelnen Positionsparameter (jeweils ein einzelnes Wort) ersetzt. |
| \$# | Anzahl der Positionsparameter |
| \$? | Rückgabewert (Status) des zuletzt ausgeführten Kommandos |
| \$- | Steht für die Optionsflags (von set oder aus der Kommandozeile). |
| \$\$ | Prozessnummer der Shell |
| #! | Prozessnummer des zuletzt im Hintergrund aufgerufenen Kommandos |

| | |
|-----|---|
| \$0 | Name des Shell-Skripts |
| \$_ | letztes Argument des zuletzt ausgeführten Kommandos |

Parametererweiterung

`${Parameter}`

Lässt sich ein Variablenname nicht eindeutig von den darauf folgenden Zeichen trennen, oder besteht ein Positionsparameter aus mehr als einer Ziffer, muss dieser Parameter in geschweifte Klammern gesetzt werden. Die folgenden Konstruktionen stellen verschiedene Arten bedingter Parametererweiterungen dar. Enthält die Konstruktion einen Doppelpunkt, so wird der Parameter daraufhin getestet, ob er leer oder ungesetzt ist. Wird der Doppelpunkt in diesen Konstruktionen weggelassen, wird nur darauf getestet, ob er gesetzt (auch leer!) oder ungesetzt ist.

Parametererweiterungen eignen sich z.B. für die Defaultwertzuweisung bei Variablen. Sie dürfen nur als Bestandteil eines Kommandos oder einer Zuweisung durchgeführt werden. Soll eine Parametererweiterung als einzelnes Kommando stehen, beispielsweise bei einer Fehlermeldung, dann muss die Zeile mit einem Doppelpunkt begonnen werden.

`${Parameter:-default}`

Wenn der *Parameter* ungesetzt oder leer ist, wird *default* anstelle des gesamten Ausdrucks eingesetzt.

`${Parameter:=default}`

Wenn der *Parameter* ungesetzt oder leer ist, wird der Inhalt von *default* dem Parameter zugewiesen und der neue *Parameter* eingesetzt.

Positionsparametern und Spezialparametern kann allerdings auch auf diese Weise kein Wert zugewiesen werden.

`${Parameter:?err_msg}`

Gibt eine Fehlermeldung wenn der *Parameter* leer oder ungesetzt ist. *err_msg* wird als Fehlermeldung auf STDERR ausgegeben. Ist der *Parameter* gültig gesetzt, wird sein Inhalt eingesetzt.

`${Parameter:+alt_value}`

Erzwingt die Benutzung eines anderen Wertes. Wenn der *Parameter* weder leer, noch ungesetzt ist, wird der Inhalt von *alt_value* eingesetzt. Sonst wird nichts eingesetzt.

`${Parameter:Offset:Länge}`

Hier wird *Parameter*, von *Offset* an, mit der Länge *Länge* neu gesetzt.

`${#Parameter}`

Gibt die Anzahl der Zeichen im Parameter wieder.

`${var#Pattern}` und `${var##Pattern}`

Entfernt den übereinstimmenden Teil von *Pattern* in *var* beginnend von links. Bei *#* wird das kürzeste treffende Stück entfernt, bei *##* das Längste.

`${var%Pattern}` und `${var%%Pattern}`

Entfernt den übereinstimmenden Teil von *Pattern* in *var* beginnend von rechts. Bei *%* wird das kürzeste treffende Stück entfernt, bei *%%* das Längste.

`${var/Pattern/Replacement}` und `${var//Pattern/Replacement}`

Das größte auf *Pattern* passende Stück in *var* wird durch *Replacement* ersetzt. Bei */* wird einmal ersetzt, bei *//* wird jede auftretende Übereinstimmung ersetzt. Werden Positionsparameter oder Arrays übergeben, wird das Kommando auf jeden einzelnen Parameter bzw. jedes Feld angewandt.

```
#!/bin/bash
leer=
default="voll"
string="1234567890"
array=( zero one two three four five )
echo ${leer-$default} # gibt nichts aus, denn $leer ist definiert
echo ${undef-$default} # gibt "voll" aus, denn
                        # $undef ist nicht definiert
echo ${leer:-$default} # gibt "voll" aus (:)
default_filename=generic.data
: ${1:?}"Dateiname wird auf \
generic.data gesetzt."} # Fehlermeldung, wenn $1 fehlt
filename=${1:=$default_filename} # setzen des Parameters
leer=${leer:+$default} # sollte leer nicht NULL sein,
                        # wird er mit "voll" belegt
echo ${string:0:1} # von links beginnend mit 0 und einem
Zeichen: 1
echo ${string:(-3):2} # von rechts und 2 Zeichen: 89
laeng_string=${#string} # ergibt 10
echo ${#array} # Laenge des ersten Elements: 4
element_count=${#array[@]} # oder
element_count=${#array[*]} # Anzahl der Elemente: 6
var1=abcd12345abc6789
pattern1=a*c # * (wildcard) trifft alles zwischen 'a' und 'c'
```

```

pattern2=b*9      # alles zwischen 'b' und '9'
echo ${var1#$pattern1}    # d12345abc6789
echo ${var1##$pattern1}   # 6789
echo ${var1%$pattern2}    # abcd12345a
echo ${var1%%$pattern2}   # a
echo ${pattern1/abc/ABC}  # "abcd12345abc6789" ->
"ABCD12345abc6789"
echo ${pattern1//abc/ABC} # "abcd12345abc6789" ->
"ABCD12345ABC6789"

```

Bedingte Ausführung

Wie in jeder Programmiersprache, können Kommandos auch miteinander verknüpft werden.

COMMAND1 && COMMAND2

stellen eine logische UND-Verknüpfung dar. Wurde Kommando1 fehlerfrei ausgeführt (exit status 0 heißt Abarbeitung ohne Fehler), wird auch Kommando2 ausgeführt.

COMMAND1 || COMMAND2

Stellen eine logische ODER-Verknüpfung dar. Kommando2 wird nur ausgeführt, wenn bei Kommando1 ein Fehler aufgetreten ist.

Tests, Verzweigungen und Schleifen

if ... then

Syntax: *if Bedingung then Liste [elif Liste then Liste...][else Liste] fi*

If ... then Konstruktionen überprüfen, ob der Exit-Status einer Liste von Kommandos 0 ist. Ist dies der Fall, werden weitere, entsprechend definierte, Kommandos ausgeführt.

Im folgenden Beispiel wird mit dem Kommando `grep` in einer Textdatei nach Zeilen, die das Wort "Bash" enthalten, gesucht. Existieren solche Zeilen, gibt `grep` als Exit-Status 0 (= true) aus. Das bedeutet, die folgenden Kommandos werden ausgeführt.

```

#!/bin/sh
if grep Bash file.txt
then echo "File contains at least one occurrence of Bash."

```


fi

Es existiert auch ein verwandtes Kommando: [...]. Dieser Ausdruck ist ein Synonym für das Bash-Kommando `test`. Es existiert außerdem ein externes Kommando `/usr/bin/test`.

```
if [ Bedingung1 ]
then
    Befehl1
    Befehl2
    Befehl3
elif [ Bedingung2 ]
# dasselbe wie else if
then
    Befehl4
    Befehl5
else
    default-command
fi
```

Weitere Informationen zum Kommando `test` bzw. [...] entnehmen Sie bitte der jeweiligen man-Page.

for...do

Syntax: `for Name [in Wort] do Liste done`

Mit *Name* wird eine Shellvariable definiert, die in jedem Schleifendurchlauf einen neuen Wert erhält. Die Werte werden normalerweise mit dem Schlüsselwort *in* übergeben. Wird der Teil *in Wort* weggelassen, wird die Liste für jeden gesetzten Parameter einmal ausgeführt.

```
#!/bin/sh
for planet in Mercury Venus Earth Mars Jupiter Saturn Uranus
do
    echo $planet
done
# oder aber auch
NUMBERS="9 7 3 8 37.53"
for number in `echo $NUMBERS` # for number in 9 7 3 8 37.53
do
    echo "$number "
done
```

while und until

Syntax: while *Liste* do *Liste* done

Syntax: until *Liste* do *Liste* done

Der Schleifenkörper *do Liste done* wird so lange wiederholt, bis die in *while Liste* formulierte Bedingung falsch ist.

Die *until*-Schleife entspricht der *while*-Schleife mit dem Unterschied, dass der *do*-Teil so lange ausgeführt wird, wie das letzte Kommando der *until Liste* einen Status ungleich 0 liefert.

```
#!/bin/sh
var0=0
LIMIT=10
while [ "$var0" -lt "$LIMIT" ]
do
    echo -n "$var0 "      # -n suppresses newline.
    var0=`expr $var0 + 1` # var0=$((var0+1)) also works.
done
```

In *for*-, *while*- und *until*-Schleifen kann ein Schleifendurchlauf vorzeitig mit **continue** beendet werden. Bei verschachtelten Schleifen kann ein Parameter übergeben werden z.B. **continue 2** (continue ohne Parameter entspricht continue 1). Bei zwei ineinander verschachtelten Schleifen wird mit continue 2 in der inneren Schleife, die Schleife abgebrochen und mit einem neuen Schleifendurchlauf in der äußeren Schleife begonnen.

Beispiel:

```
#!/bin/bash
LISTE_A="Guten Fröhlichen Gesunden"
LISTE_B="Nacht Abend Nachmittag Mittag Vormittag Morgen"
for i in $LISTE_A
do
    for j in $LISTE_B
    do
        if [ "$j" = "Nacht" ]
        then
            echo "Ich sage nicht ${i} ${j}"
            continue 2
        fi
        echo "${i} ${j}"
    done
done
echo
```

done

Durch continue 2 bedingt, weigert sich das Skript »Guten Nacht«, »Fröhlichen Nacht« und »Gesunden Nacht« zu sagen.

Anmerkung: Um eine Schleife komplett zu beenden und nicht nur einen Schleifendurchlauf, können Sie **break** benutzen.

case

Syntax: case *Wort* in [*Muster* [| *Muster*]] *Liste* ;; ...] esac

Mit der case-Anweisung können Verzweigungen programmiert werden. *Wort* wird mit den angegebenen Mustern verglichen. Bei Übereinstimmung wird die Liste von Kommandos ausgeführt. In den Suchmustern können auch Wildcards und reguläre Ausdrücke verwendet werden.

```
#!/bin/sh
arch=$1
case $arch in
  i386 ) echo "80386-based machine";;
  i486 ) echo "80486-based machine";;
  i586 ) echo "Pentium-based machine";;
  i686 ) echo "Pentium2+-based machine";;
  *    ) echo "Other type of machine";;
esac
```

Arithmetik

Arithmetische Operationen werden über die Shellkommandos **expr** und **let** realisiert. Dabei ist **let** ein internes Kommando der Bash, und **expr** ein Externes. Es ist sowohl möglich Berechnungen mit Hilfe des Kommandoaufrufes zu machen, als auch durch eine verkürzte Schreibweise:

```
#!/bin/sh
z=`expr $z + 3` # Aufruf des externen Kommandos expr
let z=z+3      # Aufruf des internen Kommandos
let "z += 3"   # Mit Quotes sind Leerzeichen und special operators
               # erlaubt.
z=$((z+3))    # neue verkürzte Schreibweise (ab Version 2.0)
z=${z+3}      # alte Schreibweise
```

Berechnungen finden, wie in C, mit »langen Ganzzahlwerten« statt. Eine Überlaufkontrolle gibt es nicht. Division durch Null führt zu einem Fehler,

der aber mit Hilfe der trap-Shellfunktion abgefangen werden kann. Folgende Operatoren sind erlaubt (Prioritätshierarchie):

| | |
|----------|---|
| + - | Vorzeichen |
| ! ~ | logische und bitweise Negation |
| * / % | Multiplikation, Division, Modulo |
| + - | Addition und Subtraktion |
| << >> | bitweise links und rechts-Shift-Operation |
| <= >= <> | Vergleiche |
| == != | gleich und ungleich |
| & | bitweise Addition |
| ~ | bitweise XOR |
| | bitweise ODER |
| && | logisches UND |
| | logisches ODER |

Funktionen

Wie auch in C, kann man in der Bash einzelne Programmteile zu Funktionen zusammenfassen. Mit dem local-Shellkommando ist es möglich, lokale Variablen für Skriptfunktionen zu erzeugen. Mit return können Werte aus einer Funktion zurückgegeben werden.

```
#!/bin/bash
myadd() {
    # $1 erstes Argument
    tmp=0
    args=$@
    for i in $args
    do
        tmp=`expr $tmp + $i`
    done
    return $tmp
}
# main
myadd 1 2 3 $VAR
RES=$?
echo $RES
myadd $RES 5 6 $VAR2
RES=$?
```

echo \$RES

Hinweis: Die mittels **return** übergebenen Werte sollten genau genommen nur ganzzahlige Zahlen von **0** bis **255** sein, d.h. die Variable die die Werte mittels **return** übergibt sollte vor Übergabe überprüft werden.

Die Variablen innerhalb einer Funktion können auch außerhalb der Funktion ausgelesen werden, es sei denn sie werden ausdrücklich als lokale Variable definiert – z.B. **local VAR**. Die lokale Variable sollte am besten gleich am Anfang der Funktion deklariert werden, danach kann mit der lokalen Variablen innerhalb der Funktion wie gewohnt gearbeitet werden.

Außerhalb der Funktion können die lokalen Variablen **nicht** ausgelesen werden.

Einige Beispiele für die Rückgabe von Werten aus Funktionen.

```
#!/bin/bash
math_1() {
  local ADD_1_RESULT_A
  ADD_1_RESULT_A=$((1+$2))
  return $ADD_1_RESULT_A
}
```

```
math_2() {
  eval echo "\$((1+$2))"
}
math_3() {
  eval echo "\$((1*))"
}
```

```
ZAHL_A=3
ZAHL_B=12
```

```
# Berechnung 1. Variante
math_1 $ZAHL_A $ZAHL_B
ADD_1_RESULT_B=$?
echo "1. Ergebnis: $ADD_1_RESULT_B"
```

```
# Berechnung 2. Variante
ADD_2_RESULT=$(math_2 $ZAHL_A $ZAHL_B)
echo "2. Ergebnis: $ADD_2_RESULT"
# Durch eval wird ein String als Befehl gewertet.
```

```
# Berechnung 3. Variante
ADD_3_RESULT=$(math_3 $ZAHL_A + $ZAHL_B)
echo "3. Ergebnis: $ADD_3_RESULT"
# Durch eval wird ein String als Befehl gewertet.

exit 0
```

Ein-/Ausgabe-Umleitungen

Jedes Programm erhält beim Start drei offene »Datenkanäle«:

| | |
|------------------------|------------|
| Standard Input: | STDIN (0) |
| Standard Output: | STDOUT (1) |
| Standard Error Output: | STDERR (2) |

Durch das Umlenken der Ein-/Ausgabekanäle können Dateien oder Dateisysteme zum Lesen bzw. Schreiben für Kommandos geöffnet werden. Es gibt viele Möglichkeiten Daten umzuleiten.

Hier ist eine kleine Auswahl:

| | |
|-------------------------------------|---|
| COMMAND<infile | Eingabeumlenkung |
| COMMAND>outfile | Ausgabeumlenkung |
| COMMAND>>outfile | Ausgabeumlenkung, anhängen |
| COMMAND 2>&1 | STDERR mit auf STDOUT legen |
| COMMAND>>EofListe
Liste EofListe | Zeilen in <i>Liste</i> werden in COMMAND umgeleitet |

Pipes

Bei einer Pipe wird der Standardausgabekanal eines Kommandos mit dem Standardeingabekanal eines anderen Kommandos zusammengelegt. Dabei werden beide Kommandos als separate Prozesse gleichzeitig gestartet. Beim folgenden Beispiel wird der Inhalt der Datei `.zshrc` durch `cat` auf den Standardausgabekanal geschrieben und an `grep` über den Standardeingabekanal übergeben. `grep` sucht nach allen Zeilen die das Wort "HISTSIZE" enthalten, und gibt diese aus.

```
cat .zshrc|grep -i HISTSIZE
export HISTSIZE=1000
```

Textmanipulationen

In diesem Kapitel soll es um das komplexe Thema »Suchen und Ersetzen« gehen. Im Weiteren wird auf die Grundlagen eingegangen und zwei der mächtigsten Tools auf diesem Gebiet kurz beleuchtet. Wer es wirklich genau wissen will, sollte sich im Internet nach weiterführenden Dokumentationen umsehen.

reguläre Ausdrücke

Es gibt in der UNIX-Welt einige sehr mächtige Tools (sed, awk, grep), die das Durchsuchen von Texten nach bestimmten Mustern ermöglichen. Um diese Tools effektiv nutzen zu können, ist es unbedingt notwendig, sich mit regulären Ausdrücken zu beschäftigen.

Reguläre Ausdrücke beschreiben eine nicht leere Menge von Zeichenfolgen, die aus Textzeichen (Buchstaben, Ziffern, Sonderzeichen) und/oder Metazeichen mit erweiterter Bedeutung bestehen. Textzeichen stehen für sich selbst, Metazeichen (Spezialzeichen) stellen Operatoren dar, mit deren Hilfe komplexe Textmuster beschrieben werden können. Als Begrenzung der Mustersuche gilt in den meisten Fällen das Zeilenende. D.h. es ist nicht möglich reguläre Ausdrücke zu definieren, die über das Zeilenende hinaus prüfen.

siehe auch: man 7 regex

Zeichenklassen

| | |
|---------|---|
| . | Ist ein Platzhalter und bezeichnet jedes einzelne Zeichen außer das Zeilenende. |
| [abc\$] | Trifft alle aufgeführten Zeichen. |
| a-c | Bezeichnet alle Zeichen im angegebenen Limit. |
| [^exp] | Trifft alle Zeichen außer den angegebenen. |
| ^abc | Trifft das angegebene Muster, wenn es am Zeilenanfang steht. |
| abc\$ | Trifft das angegebene Muster, wenn es am Zeilenende steht. |
| \ | Maskierung des folgenden Zeichens |

Wiederholungsoperatoren

| | |
|---|--|
| * | Trifft den vorangegangenen Ausdruck 0 oder mehrmals. |
| + | Trifft den vorangegangenen Ausdruck ein oder mehrmals. |

| | |
|-------|--|
| ? | Trifft den vorangegangenen Ausdruck 0 oder einmal. |
| | Ist ein Trennzeichen. Trifft entweder den folgenden oder vorangegangenen Ausdruck. |
| (...) | Bildet eine Gruppe von regulären Ausdrücken. |

Die Syntax von grep und egrep variiert in manchen Punkten. Für +, ?, |, (...) ergibt sich für grep eine andere Schreibweise: \+, \?, \|, \(...). Reguläre Ausdrücke werden von links nach rechts aufgelöst.

Operatoren werden in der folgenden Reihenfolge abgearbeitet:

[...] ? + * Verkettung Verknüpfungen |

Abweichungen davon können mit Klammerung einzelner Ausdrücke erreicht werden.

Die Operatoren ?, +, *, ^, \$ und | können wiederum auch auf gruppierte (geklammerte) Ausdrücke angewendet werden. Beispielsweise trifft (AB|CD+)?(EF)+

die Zeichenketten ABEF, CDEF, CDDEF, EFEF, EFEFEF usw..

Das hinter dem ersten Klammernpaar stehende Fragezeichen bedeutet, dass das Vorkommen, der darin enthaltenden Zeichenketten AB und CD, optional ist. Das Pluszeichen hinter D sagt aus, dass nach einem oder mehreren D's gesucht wird (CD, CDD, CDDDD, ...). Die so gefundenen Zeichenketten der ersten Klammer, müssen unmittelbar gefolgt sein von mindestens einem Vorkommen der Zeichenkette EF. Das Plus bezieht sich hier auf den ganzen Klammerausdruck.

Wie aus obigen Regeln zu ersehen ist, handelt es sich bei regulären Ausdrücken um eine Wissenschaft. Die hier aufgezeigten Muster stellen nur einen Auszug dar. Noch ein paar Beispiele hinterher:

| | |
|------------|---|
| .aus | trifft Haus, raus, Maus, Laus,... |
| xy*z | trifft auf xy...was auch immer...z |
| ^abc | jede Zeile, die mit abc beginnt |
| abc\$ | jede Zeile, die mit abc endet |
| * | trifft jeden Stern |
| [Mr]aus | trifft Maus und raus |
| [[abc] | trifft [(muss am Anfang stehen), a, b, c |
| [KT]?ELLER | trifft ELLER, TELLER, KELLER |
| [^a-zA-Z] | schließt alle Buchstaben aus |

| | |
|------------|--|
| [0-9]\$ | Trifft jede Zeile, die mit einer Zahl endet. |
| [0-9][0-9] | trifft jede zweistellige Nummer |
| H(e a)llo | trifft Hallo und Hello |
| (ab)? | trifft entweder "ab" oder nichts ("ab" ist optional) |
| ^\$ | trifft alle Leerzeilen |

Aber es ist noch mehr möglich.

| | |
|---------|--|
| \{n,m\} | Trifft ein Muster mindestens n-mal und höchstens m-mal. |
| \<abc> | Trifft das eingeschlossene Muster nur, wenn es sich um ein separates Wort handelt.
\< ... Wortanfang; \> ... Wortende |
| \(abc\) | Die Klammern fassen Ausdrücke zusammen. Jede Zeile wird nach dem angegebenen Muster durchsucht und jeder Treffer wird in einem Puffer gespeichert (max. 9 dieser Muster sind in einem Befehl möglich). |
| \n | referenziert obige Muster |

Es lassen sich des weiteren syntaktische Gruppen bilden. Hierbei handelt es sich nur um eine andere Schreibweise bereits besprochener Ausdrücke. Diese Schreibweise kann die Lesbarkeit regulärer Ausdrücke deutlich verbessern.

| | |
|------------|---|
| [:alnum:] | alle alphanumerischen Zeichen [A-Za-z0-9] |
| [:alpha:] | alle Buchstaben [A-Za-z] |
| [:blank:] | ein oder mehrere Leerzeichen und Tab |
| [:cntrl:] | alle Kontrollzeichen wie z.B. <newline> |
| [:digit:] | alle dezimalen Zahlen [0-9] |
| [:graph:] | alle druckbaren Zeichen (ASCII 33 - 126) ohne das Leerzeichen |
| [:print:] | alle druckbaren Zeichen |
| [:lower:] | alle Kleinbuchstaben [a-z] |
| [:upper:] | alle Großbuchstaben [A-Z] |
| [:space:] | Leerzeichen und horizontales Tab |

| | |
|--------------|---------------------------------------|
| [[:xdigit:]] | alle hexadezimalen Zahlen [0-9A-Fa-f] |
|--------------|---------------------------------------|

Reguläre Ausdrücke sind zwar allgemein gültig, jedoch ist der Funktionsumfang der einzelnen Tools nicht einheitlich.

| | [...] | . | * | ? | + | ^ | \$ | | () |
|-------|-------|---|---|---|---|---|----|---|----|
| grep | x | x | x | | | x | x | | |
| egrep | x | x | x | x | x | x | x | x | x |
| sed | x | x | x | | | x | x | x | x |
| awk | x | x | x | x | x | x | x | x | x |

grep

Syntax: grep [-CVbchilnsvwX] [-Anzahl] [-AB Anzahl] [[-e] Ausdruck | -f Datei] [Datei...]

grep durchsucht die angegebenen Dateien (oder die Daten aus der Standardeingabe) nach einem Ausdruck und gibt die entsprechenden Zeilen aus. Der Status von grep ist 0, wenn der Ausdruck gefunden wurde und sonst 1.

Wieder ein paar einfache Beispiele:

| Befehl | cat file | grep b.*g file | grep b.*g. file | grep ggg* file |
|----------|----------|----------------|-----------------|----------------|
| Resultat | big | big | bigger | bigger |
| | bad bug | bad bug | boogy | |
| | bag | bag | | |
| | bigger | bigger | | |
| | boogy | boogy | | |

Stern und Punkt sind Sonderzeichen. Will man nach Mustern suchen, die den Punkt als literarisches Zeichen auffassen, so muss dieser maskiert werden.

```
ls | grep Name.ext
trifft auch Name0ext, NameBext, usw...
ls | grep Name\ext
```

Trifft nur die Datei mit dem Namen Name.ext.

Wir wollen in einem Textfile alle Zeilen, die den Namen Fred Feuerstein und Fredericke Feuerstein enthalten. Das bedeutet der Teil "ericke" ist optional.

```
grep "Fred\(\ericke\)? Feuerstein" textfile
```

Die Klammern bilden eine Gruppe. Das Fragezeichen bedeutet ein oder kein Vorkommen des vorherigen Musters.

Hier werden Klammern innerhalb anderer Klammern ausgeschlossen:

```
grep "([^\()*)"
```

Trifft (hello) und (aksjdhaksj d ka) aber nicht $x=(y+2(x+1))$.

Jetzt wollen wir nach sich wiederholenden Mustern suchen. Eine gutes Beispiel sind Telefonnummern. Wir suchen nach einer Vorwahl (3 Ziffern) und der Nummer (7 Ziffern), getrennt durch einen - , einem Leerzeichen oder gar nicht.

```
grep "[0-9]\{3\}[-]?[0-9]\{7\}" file
```

[0-9] steht für alle Zahlen, \{3\} besagt, dass sich das vorherige Muster 3 mal wiederholen soll. [-]? repräsentiert die Auswahl des Trennzeichens (Leerzeichen, - oder gar nichts).

Angenommen, wir suchen eine Zeile in der nur das Wort "Hallo" steht. Es ist zudem noch möglich, dass sich vor und/oder hinter "Hallo" Leerzeichen befinden. Eine Möglichkeit wäre folgendes:

```
grep "^[:,space:]*Hallo[:,space:]*$" file
```

^ steht für den Zeilenanfang, \$ für das Zeilenende.

Manchmal ist es nötig, Zeilen zu suchen, in denen entweder das Eine oder das Andere steht.

```
grep "Ich habe \(\Schröder\|Stoiber\) gewählt" file
```

\| entspricht einem logischen ODER.

Hat man einmal ein Muster in \(...\) definiert, kann man es mit \Zahl erneut einsetzen.

```
echo bla blub bla | grep "\(bla\).*\1"
```

sed

Syntax: sed [-n] [-e Editorkommando] [-f Skriptdatei] [inputfile]

sed ist ein Editor zur automatischen Textbearbeitung. Die Bearbeitung erfolgt mit Editorkommandos, die sed in einer separaten Skriptdatei oder direkt in der Kommandozeile übergeben werden. Um in der Kommandozeile mehrere Editorkommandos zu übergeben, kann die -e Option mehrfach verwendet werden. Die Editorkommandos können auch durch ein Semikolon getrennt werden. Wird nur ein einziges Editorkommando in der Kommandozeile übergeben, kann die Kennzeichnung mit der -e Option auch weggelassen werden. Damit die Shell keine Veränderungen an der Zeichenkette mit dem Editorkommando vornimmt, muss sie in Hochkommas eingeschlossen werden.

Eine Skriptdatei kann beliebig viele Editorkommandos enthalten, die durch Zeilenende oder Semikolon voneinander getrennt werden müssen.

Jedes Kommando besteht aus einem Adressteil und einem Funktionsteil. Der Adressteil gibt an, welche Zeilen einer Textdatei mit diesem Kommando bearbeitet werden sollen, und der Funktionsteil beschreibt die Veränderung, die an den im Adressteil bestimmten Zeilen vorgenommen werden soll. Wenn kein Adressteil angegeben ist, wird die Funktion mit jeder Zeile ausgeführt.

Die Bearbeitung eines Textes erfolgt, indem die Eingabe zeilenweise in einen Arbeitsspeicher gelesen wird, und dann der Adressteil aller Editorkommandos der Reihe nach mit dem Text im Arbeitsspeicher verglichen werden. Die Funktionen der passenden Kommandos werden in der Reihenfolge ihres Auftretens ausgeführt. Normalerweise wird nach der Bearbeitung aller Kommandos der Inhalt des Arbeitsspeichers auf die Standardausgabe ausgegeben und danach durch die nächste Eingabezeile ersetzt. Die automatische Ausgabe des Arbeitsspeichers nach jeder Zeile kann mit der Option -n unterdrückt werden.

Zusätzlich zu dem Arbeitsspeicher gibt es noch einen Zwischenspeicher (Puffer), der von verschiedenen Funktionen benutzt werden kann.

Der Arbeitsspeicher kann auch mehrere Zeilen auf einmal enthalten.

Im Adressteil können die Zeilen entweder durch ihre Zeilennummern, oder durch reguläre Ausdrücke ausgewählt werden. Alle Funktionen außer den a, i, q und = akzeptieren einen Adressbereich, bei dem eine Start- und eine Endadresse durch ein Komma getrennt angegeben werden. Ein Dollarzeichen steht für die letzte Zeile. Wenn eine Endadresse mit einem regulären Ausdruck bezeichnet ist, wird die erste passende Zeile als Bereichsende eingesetzt.

Reguläre Ausdrücke müssen in einfachen Schrägstrichen (Slashes /) eingeschlossen werden. sed benutzt die gleichen Routinen zur Auswertung

regulärer Ausdrücke wie emacs oder grep. Darüber hinaus kann auch die an die sed Syntax angelehnte Konstruktion `/Muster/#` (mit jedem beliebigen Zeichen für `#`) benutzt werden, die wie `/Muster/` interpretiert wird. Im Muster kann auch ein `\n` vorkommen, das auf das Zeilenende passt.

Optionen von sed

Mit sed gibt es schier unendliche Möglichkeiten der Textmanipulation. sed bietet unter anderen folgende Funktionalität:

| Operator | Name | Effekt |
|--|------------|--|
| <code>[Muster/Adressraum]/p</code> | print | Gibt den mit <code>[Muster/Adressraum]</code> gekennzeichneten Bereich aus. |
| <code>[Adressraum]/d</code> | delete | Löschen des mit <code>[angegebener Adressraum]</code> gekennzeichneten Bereichs. |
| <code>s/Muster1/Muster2/</code> | substitute | Ersetze das erste in einer Zeile auftretende Muster1 durch Muster2. |
| <code>[Adressraum]/s/Muster1/Muster2/</code> | substitute | Ersetze über einen angegebenen Adressraum das erste in einer Zeile auftretende Muster1 durch Muster2. |
| <code>[Adressraum]/y/Muster1/Muster2/</code> | transform | Transformiere über einen angegebenen Adressraum, jedes Zeichen in Muster1 durch das korrespondierende Zeichen in Muster2 (äquivalent zum Befehl tr.) |
| <code>g</code> | global | Wendet das vorherstehende Kommando auf jedes vorkommende Ersetzungsmuster einer Zeile an. |
| <code>[Anzahl]q</code> | quit | Beendet sed nach "Anzahl" Zeilen. |
| <code>[Muster/Adressraum]/w file</code> | write | Schreibt gefundene Zeilen in file. |
| <code>i\ Text</code> | insert | fügt Text vorher ein |
| <code>a\ Text</code> | append | fügt Text danach ein |

| c\ Text | change | ersetzt durch Text |
|---------|--------|--|
| = | | Gibt die aktuelle Eingabezeilennummer aus. |
| {...} | | Die von den Klammern eingeschlossenen und durch Zeilenende oder Semikolon getrennten Funktionen, werden als Einheit behandelt. |

Beispiele

Im ersten Beispiel soll das Wort »UNIX« durch das Wort »Linux« ersetzt werden. Das abschließende g sorgt dafür, dass jedes in einer Zeile befindliche »UNIX« durch »Linux« ersetzt wird. Ohne g würde nur das erste auftretende »UNIX« ersetzt. Der Befehl ist in Hochkommata eingefasst, damit Sonderzeichen von der Shell nicht interpretiert werden (in diesem Fall die Klammern). Ohne die Hochkommata sieht der Befehl etwas anders aus. Klammern und Leerzeichen müssen maskiert werden.

```
sed 's/UNIX (TM)/Linux/g' file
sed -e s/UNIX\ (TM)/Linux/g file
```

Es ist möglich die Befehlsfolgen, die sed abarbeiten soll, in einer Skriptdatei zu speichern. Die Ausgabe lässt sich auf dem üblichen Wege in eine Datei umleiten. Der Befehl sieht dann so aus:

sed -n -f muster_file inputfile > outputfile

Normalerweise gelten die vorgegebenen Muster immer für den gesamten Text. Aber es ist auch möglich bestimmte Zeilen und Bereiche zu adressieren. Dies kann mit Zahlen, Mustern und mit dem \$-Zeichen, als Kennung für die letzte Zeile, geschehen. Mit dem ! können Zeilen ausgeschlossen werden. Die Syntax sieht dann folgendermaßen aus:

[Adresse1],[Adresse2] Kommando [Parameter]

Beispiele sagen mehr als tausend Worte:

```
sed '10,$ s/WWJD/TWJD/g' file
sed '/Josef/ s/WWJD/TWJD/g' file
sed '!s/WWJD/TWJD/g' file
sed 'y/OPY/AHU/' file
```

Im ersten Beispiel wird von Zeile 10 bis zum Dateiende der String »WWJD« in »TWJD« geändert. Im zweiten Beispiel wird nur in Zeilen ersetzt, in denen das Wort »Josef« vorkommt. Das dritte Beispiel verändert »WWJD« in »TWJD« überall, außer in der ersten Zeile. Im letzten Beispiel wird O durch A, P durch H und Y durch U ersetzt, dies gilt hier für die gesamte Datei - die Ausgabe erfolgt auf die Standardausgabe (Bildschirm).

Noch ein paar nützliche Muster:

| | |
|--------------------|--|
| 8d | löscht die achte Zeile |
| /^\$/d | löscht alle leeren Zeilen |
| 1,/^\$/d | löscht alles von Zeile 1 bis einschließlich der ersten leeren Zeile |
| /GUI/d | löscht alle Zeilen in denen »GUI« vorkommt |
| /Jones/p | gibt nur Zeilen aus in denen der Name »Jones« vorkommt (mit -n) |
| 1,10 p | gibt Zeilen 1 bis 10 aus (mit -n) |
| /^begin/,/^end/p | gibt jede Zeile aus, die sich zwischen Zeilen die mit »begin« und »end« am Zeilenanfang befindet |
| s/Windows/Linux/ | ersetzt das erste in einer Zeile vorkommende, »Windows« mit »Linux« |
| s/BSOD/stability/g | setzt »stability« für jedes »BSOD« ein |
| s/00*/0/g | ersetzt "00", "000", ... mit "0" |
| s/GUI/g | löscht »GUI« in jeder Zeile |
| /^[0-9]/s/^/ / | alle Zeilen, die mit einer Zahl beginnen, um 3 Leerzeichen einrücken (...s/^»3 Leerzeichen«/) |
| /^\$/s/^/XXX/ | alle leeren Zeilen mit »XXX« auffüllen |
| 10q | zeigt die ersten 10 Zeilen an |
| /^X/w file | schreibt alle Zeilen, die mit »X« beginnen, in file |

Kniffliger wird die Angelegenheit, wenn es sich um die Optionen i, a und c handelt:

```
#!/usr/bin/sed -nf
/ganz bestimmter Text/{
    i\
    Text gehört davor
    a\
    Text der danach stehen soll
```

```
}
```

Kommando und Ergebnis sehen dann so aus:

```
echo 'ganz bestimmter Text' | sed -f skriptdatei
Text gehört davor
ganz bestimmter Text
Text der danach stehen soll
```

Aber es geht natürlich auch ohne Skriptdatei:

```
echo 'ganz bestimmter Text' | \
sed -e '/ganz bestimmter Text/{;i\' \
-e 'Text gehört davor' -e 'a\' -e 'Text der danach stehen soll' -e '}'
```

An diesem Beispiel kann man des Weiteren erkennen, dass es möglich ist, Befehlsgruppen zu bilden. Eine Gruppe wird durch die geschweiften Klammern zusammengefasst. Das angegebene Suchmuster »/ganz bestimmter Text/« wird dadurch von a und i gemeinsam genutzt.

Nichts verstanden? Macht nichts, es ist auch nicht einfach! Im Internet gibt es eine Vielzahl von Tutorien zu regulären Ausdrücken, sowie zu sed und grep. Und wie immer kann auch ein Blick ins Manual nicht schaden.

* * * * *

Hinweise zu einigen Programmen und speziellen Vorgehensweisen

Multimedia-Live-CD: GeeXboX

GeeXboX ist eine Linux-Live-CD nach dem Knoppix-Prinzip, d.h. Sie laden sich das ISO-Image aus dem Internet und brennen dieses anschließend auf eine CD-ROM. GeeXboX basiert auf den Mplayer und spielt verschiedene Video-Formate, MP3-, OGG-, WAV-, Real-Player-Dateien und einige andere Multimedia-Formate.

6. Das ISO-Image **geebox-0.98.6-en.i386.iso** oder höher von der Webseite **<http://www.geebox.org/download/>** herunterladen und als CD-Abbilddatei auf CD-ROM brennen.
7. Lassen Sie die bootfähige CD im Laufwerk und starten den Rechner neu (evt. muss die Bootreihenfolge im BIOS geändert werden). Das GeeXboX-Logo erscheint und die Distribution bootet.
8. GeeXboX-Hauptmenü erscheint und der CD/DVD-Schacht öffnet sich automatisch und Sie können die GeeXboX-CD z.B. gegen eine beliebige Film-DVD austauschen.
9. Gewöhnlich spielt GeeXboX die DVD nach ein paar Sekunden selbstständig ab - häufig jedoch in Originalsprache. Das ändern Sie mit der Tastatur.

Tastaturbefehle:

[m] ... On-Screen-Display (OSD) erscheint, Auswahl mittels der Pfeiltasten
Die Einstellung für den Audio-Kanal versteckt sich hinter dem Menüpunkt »Controls« bzw. »Options« und durch erneutes Drücken auf die Taste **[m]** verschwindet das On-Screen-Menü wieder.

[j] ... wechseln oder anzeigen der Untertitel während der Film läuft

[v] ... ausschalten der Untertitel

[Pfeil-nach-rechts] und **[Pfeil-nach-links]** ... kleine Zeitsprünge von 10 Sekunden

[Pfeil-nach-oben] und **[Pfeil-nach-unten]** ... Zeitsprünge von 1 Minute

[Bild-nach-oben] und **[Bild-nach-unten]** ... Zeitsprünge von 10 Minuten
[*]/[] (auf dem rechten Nummernblock) bzw. **[0]** und **[9]** (Zahlen über den Buchstaben) ... Ton lauter oder leiser stellen

[c] ... Ton abschalten

[Leertaste] ... Pause, durch nochmaliges Drücken wird der Film wieder gestartet

[h] oder **[g]** ... spielt jeweils den nächsten bzw. den vorherigen Track auf der DVD ab

[+] und **[-]** auf dem rechten Nummernfeld ... beeinflusst die Audio-Video-

Verzögerung, falls Ton und Bild nicht synchron laufen

[+] (nicht auf dem Nummernblock) und [ü] ... Film schneller bzw. langsamer abspielen

[Strg] + [Alt] + [Entf] ... Neustart des Systems, z.B. falls das System nicht mehr reagiert

[q] ... quit, Film beenden

[Pfeil-nach-rechts] und [Pfeil-nach-links] ... Navigation innerhalb des Menüs (siehe: [m])

GeeXboX beenden ... Taste [m] drücken und anschließend den Menüpunkt »Quit« auswählen und mit [Enter] bestätigen => evtl. vorher mit der Pfeiltaste [Pfeil-nach-links] zum Hauptmenü navigieren

Anmerkung: Falls GeeXboX das Bildformat nicht richtig erkannt hat, dann rufen Sie das Menü über die Taste [m] auf. Im Hauptmenü wählen Sie »Options« »Aspect...« und wählen dort das richtige Bildformat aus – evt. ein wenig mit den Einstellungen experimentieren. Falls ein Untermenü nach dem Druck auf die Taste [m] erscheint, so bewegen Sie sich mit der Pfeiltaste [Pfeil-nach-links] zum Hauptmenü.

Einige Tastatur-Befehle funktionieren **nicht** in allen Versionen von GeeXboX. Weitere Details finden Sie im Hilfemenü, auf der Webseite von GeeXboX oder im Internet.

Administrator-Passwort vergessen

Info-Stand: 2005 bzw. 2014

Achtung: Die Anwendung der nachfolgenden Hinweise erfolgt auf **EIGENE GEFAHR**.

Anmerkung: Haben Sie unter Linux das root-Passwort vergessen, so hilft Ihnen nur noch die Rettungs-CD oder ein Linux-Livesystem wie KNOPPIX aus dem Dilemma.

siehe auch: passwd

A: Betriebssystem Windows 2000

Die nachfolgenden Zeilen gelten weitestgehend auch für das Betriebssystem Windows NT.

1. Einführung und Voraussetzungen

Windows 2000 ist so konzipiert, dass es keinerlei Fremdzugriffe zulässt. Eine Ausnahme gibt es allerdings, wenn die Datei **sam** beschädigt ist oder gar völlig fehlt. In diesem Fall, sind keine Benutzer mehr auf Ihrem System eingerichtet und Sie können sich die Passworteingabe sparen. Es hat also jeder Benutzer freien Zugriff auf alle Ordner und Dateien.

allg. Voraussetzungen:

- Sie haben physischen Zugriff zum Rechner, von dem Sie das Passwort vergessen haben - d.h. Sie bekommen mit den nachfolgenden Hinweisen nur Zugriff auf einen lokalen Rechner, nicht aber Zugriff auf Netzwerk-Rechner (z.B. Anmelde-Server, Fileserver etc.)
- Ist der Rechner Mitglied eines lokalen Netzwerkes, so ist Ihnen mit einem vergessenen Administrator-Passwort für das Netzwerk der Zugang zu Netzwerk-Ressourcen auch weiterhin versperrt. Es sei denn, Sie haben auch physischen Zugang zum Anmeldeserver (PDC ... Primary Domain Controller) des lokalen Netzwerkes.
- Die Benutzer und die Passwörter werden in der SAM-Datei (SAM ... Security Access Management; %Systemroot%\winnt\system32\config\sam bzw. %Systemroot%\WINDOWS\system32\config\sam) gespeichert. Der direkte Zugriff auf die SAM-Datei ist nicht möglich, da sie ständig vom Betriebssystem benutzt wird und somit immer geöffnet ist. Standardmäßig existiert zudem noch eine Kopie der SAM-Datei (\Winnt\repair\SAM._), die jeder lesen kann, der angemeldet ist. Um einen Brute-Force-Angriff zu erschweren, sollte die Kopie der SAM-

Datei daher lediglich auf einem externen Datenträger hinterlegt werden.

2. Administrator-Passwort vergessen, aber es existiert noch ein Benutzer mit Administrator-Rechten

Bei der Windows-2000-Installation haben Sie ein Administrator-Passwort vergeben, dass Sie später vergessen haben?

Dann können Sie es ändern, wenn Sie sich als »zusätzlicher« Benutzer mit Administrationsrechten anmelden. D.h. es muss neben dem Administrator mindestens noch ein Benutzer mit Zugang zum Rechner existieren, von dem man das Passwort **nicht** vergessen hat und dieser Benutzer hat Administrationsrechte.

- Start => Ausführen und CMD eingeben.
- In der folgenden Eingabeaufforderung gibt man dann **net User Administrator <Kennwort>** ein. Statt Kennwort ein neues Passwort eintragen.
- Das alte Passwort wird dabei überschrieben, ohne dass nach dem alten Passwort gefragt wird.

»Normalen« Benutzern bleibt dieser Weg versperrt - Systemmeldung:
»Zugriff verweigert«.

Anmerkung: Bei den NICHT-Professional-Versionen (z.B. XP-Home) kann das Passwort auch im abgesicherten Modus geändert werden - während des Hochfahrens an »passender« Stelle die Funktionstaste [F8] drücken.

3. Windows-Rechner mit dem Dateisystem vfat

Die Passwörter sind in der SAM-Datei (SAM ... Security Access Management; %Systemroot%\winnt\system32\config\sam bzw. %Systemroot%\WINDOWS\system32\config\sam) gespeichert, wird diese Datei gelöscht, kann man sich beim nächsten Neustart als Administrator ohne Passwort anmelden. Dabei gehen allerdings auch alle Benutzerkonten auf dem Rechner verloren! Da der lokale Administrator standardmäßig auch Wiederherstellungsagent ist, kann man auf diese Weise unter Umständen auch wieder auf alle verschlüsselten Daten im System zugreifen.

Die Löschung wird von einem »Fremdsystem« aus durchgeführt: Ist Windows 2000 auf einer FAT-16/32 Partition installiert, wird von Win98/Me CD-ROM gebootet und die Datei unter Windows mit dem Befehl

del c:\windows\system32\config\sam

gelöscht (%windir% und Laufwerk sind natürlich entsprechend anzupassen, z.B. d:\winxp; vor dem Löschen sollte die SAM-Datei evtl. kopiert werden, falls man sich später doch wieder an das Passwort erinnert: copy sam sam.old). Ist NT4 oder W2k auf einer NTFS-Partition installiert, kann von einer normalen DOS-/Windows-Umgebung nicht auf das Dateisystem zugegriffen werden.

Sollte ein zweites Betriebssystem installiert sein, kann die SAM auch direkt von diesem aus gelöscht werden (bei 98/Me wieder nur mit Zusatztool / nicht kostenlos).

SAM-Datei unter Knoppix löschen oder umbenennen:

Das CD-ROM oder das DVD-Laufwerk sollte bootfähig sein (evtl. BIOS-Einstellungen ändern). Nachfolgend wird davon ausgegangen, dass das Windowslaufwerk von Linux mit den Namen **hda1** angesprochen wird. Falls Linux bei Ihnen einen anderen Namen vergeben hat, so gilt nachfolgendes analog.

- den Rechner mit KNOPPIX booten
- das Windows-Laufwerk mit einem Dateimanager öffnen - Desktop-Icon »**Hard Disk Partition [hda1]**« anklicken
- anschließend, das Kontextmenü dieses Desktop-Icon aufrufen und den Menü-Punkt »**Aktionen**« auswählen und den Schreibzugriff auf das Laufwerk aktivieren
- im Dateimanager ins Verzeichnis ...\\Windows\\System32\\config wechseln und die Datei **sam** löschen oder in **old_sam** umbenennen
- den Dateimanager schließen und die Laufwerksverbindung über das Kontextmenü des Desktop-Icon lösen
- Knoppix beenden und den Rechner mit Windows hochfahren
- in der Anmeldemaske des Windowsrechners als Benutzernamen **Administrator** eingeben, das **Passwortfeld** bleibt **leer**
- danach können die »normalen« Benutzerkonten wieder neu eingerichtet werden

Hinweis: In aktuellen KNOPPIX-Versionen kann die Vorgehensweise von der hier beschrieben abweichen.

4. Windows-Rechner mit dem Dateisystem ntfs

Die Passwörter sind in der SAM-Datei (SAM ... Security Access

Management; %Systemroot%\winnt\system32\config\sam bzw. %Systemroot%\WINDOWS\system32\config\sam) gespeichert, wird diese Datei gelöscht, kann man sich beim nächsten Neustart als Administrator ohne Passwort anmelden. Dabei gehen allerdings auch alle Benutzerkonten auf dem Rechner verloren! Da der lokale Administrator standardmäßig auch Wiederherstellungsagent ist, kann man auf diese Weise unter Umständen auch wieder auf alle verschlüsselten Daten im System zugreifen.

Mit Zusatztools wie NTFSDOS-Pro kann man mit DOS auf das NTFS-Filesystem zugreifen und die Datei wie oben beschrieben (siehe Pkt. 3.) löschen. Allerdings ist nur die (in diesem Fall) nutzlose Read-Only Version Freeware, die Vollversion mit Lese-/Schreibzugriff kostenpflichtig ...

* * *

Der LINUX-Treiber - z.B. von der Live-Distribution KNOPPIX mit den Kernelversionen 2.4 (erweiterte Version) bzw. 2.6.x - kann alle Versionen von NTFS lesen - trotzdem, dass jede Version leicht verschieden ist und der Treiber nur die Dateien und Verzeichnisse zu interpretieren hat.

Der Schreibzugriff wird standardmäßig von den Linux-Systemen nicht aktiviert, d.h. er muss erst nachträglich eingerichtet werden.

NTFS unterstützt eine breite Palette an Features - aber leider nur wenige können von den derzeitigen Linux NTFS-Treiber imitiert werden.

Wie schon erwähnt, sollte der Treiber im Nur-Lese-Modus sicher sein, da er keinerlei Änderungen an der Partition vornimmt.

Hinweis: In aktuellen KNOPPIX-Versionen kann die Vorgehensweise von der hier beschriebenen abweichen.

Achtung: Fehlverhalten durch irgendwelche Inkompatibilitäten können in diesem noch **experimentellen Stadion nicht** ausgeschlossen werden. Bei meinen Experimenten - Testumgebung: Knoppix 3.2 bzw. 3.8 und Windows 2000, Dateisystem NTFS, Service Pack 4 - wurde der Schreibzugriff nur ungenügend unterstützt. Aus diesem Grund kann ich Knoppix hier nicht empfehlen, deshalb wird nachfolgend eine etwas umständlichere Variante beschrieben (Info-Stand: 2005).

* * *

Grundsätzlich wollen wir hier mit dem Dateisystem NTFS dasselbe

erreichen, wie unter dem Punkt 3. »Windows-Rechner mit dem Dateisystem vfat« (siehe weiter oben) schon beschrieben wurde - das Löschen oder die Umbenennung der SAM-Datei.

Voraussetzungen: Für diese Variante ist ein zweiter Rechner mit der **gleichen Betriebssystem-Version** (am besten mit dem NTFS-Dateisystem) notwendig. Nachfolgend wird davon ausgegangen, dass im zweiten Rechner eine Festplatte und CD-ROM oder DVD-Laufwerk eingebaut sind. Festplatte (Master) und CD-ROM oder DVD-Laufwerk (Slave) sind am 1. IDE-Kabel angeschlossen. Am 2. IDE-Kabel sind keine Geräte angeschlossen. Falls dies bei Ihnen nicht so ist, so sollten Sie sich mit den Einstellungen von IDE-Laufwerken - Master, Slave - oder SCSI-Laufwerken beschäftigen (siehe: Internet).

- Festplatte aus dem Rechner mit dem vergessenen Passwort ausbauen
- diese Festplatte in den zweiten Rechner am 2. IDE-Kabel anschließen - der Jumper sollte auf Master stehen, i.d.R. sind hier keine Änderungen notwendig; die Stromversorgung für die Festplatte nicht vergessen
- von aktuellen BIOS-Versionen wird die neu eingebaute Festplatte automatisch erkannt, d.h. es brauchen hier keine Änderungen eingetragen werden; falls Sie sich nicht sicher sind, so überprüfen Sie die Eintragungen für die Festplatte im BIOS; i.d.R. sind hier keine Änderungen notwendig
- Rechner hochfahren und die SAM-Datei auf der **zweiten Festplatte** löschen oder umbenennen in **old_sam**; Verzeichnis e:\Windows\System32\config (%windir% und Laufwerk sind natürlich entsprechend anzupassen)
- Rechner herunterfahren und die zweite Festplatte wieder ausbauen
- Festplatte im 1. Rechner einbauen und den 1. Rechner hochfahren
- in der Anmeldemaske des Windowsrechners als Benutzernamen **Administrator** eingeben, das **Passwortfeld** bleibt **leer**
- danach können die »normalen« Benutzerkonten wieder neu eingerichtet werden

5. zusätzliche Variante - Administrator Passwort vergessen

Die Hinweise für Datenträger mit dem Dateisystem **ntfs** gelten hier analog.

Achtung: Diese Variante funktioniert nur mit Betriebssystem-Versionen die

mindestens eines der folgenden Programme installiert haben:
ADDUSERS.EXE, **USRMGR.EXE** oder **MUSRMGR.EXE**. Diese Programme sind **nicht** in jeder Betriebssystem-Version verfügbar. Außerdem sollten die Sicherheitseinstellungen den Aufruf dieser Programme noch vor einer »ordentlichen« Benutzeranmeldung erlauben.

1. Booten mit einem fremden Betriebssystem - DVD, CD-ROM, Linux-Livesystem (z.B. KNOPPIX)
2. Wechseln ins Verzeichnis c:\winnt\system32
3. die Datei **logon.scr** in **logon.old** umbenennen - **rename logon.scr logon.old**
4. die Datei **cmd.exe** in **logon.scr** umbenennen - **rename cmd.exe logon.scr**
5. Reboot des Rechners – DVD bzw. CD-ROM vorher entfernen
6. die Windows-Anmeldemaske ca. 15 Minuten stehen lassen (Bildschirmschoner springt dann an; jetzt aber die DOS-Box)
7. Nach ca. 15 Minuten öffnet sich eine DOS-Box. Dort folgendes eingeben **ADDUSERS.EXE**, **USRMGR.EXE** oder **MUSRMGR.EXE**.
8. Usermanager öffnet sich. Dort einen Admin oder einen Benutzer mit Administratorrechten (**siehe auch weiter oben: 2. Administrator-Passwort vergessen**, aber es existiert noch ein Benutzer mit Administrator-Rechten) einrichten.
9. Reboot
10. Alle Änderungen von Punkt 3 und 4 rückgängig machen - d.h. die Datei **logon.scr** in **cmd.exe** und die Datei **logon.old** in **logon.scr** umbenennen.
11. Reboot

Anmerkung: Es gibt zwar auch Programme mit denen man die SAM-Datei entschlüsseln kann, dies kann aber bei ausreichend sicheren Passwörtern mehrere Wochen oder länger dauern. Für weitere Infos steht Ihnen das gesamte Internet zu Verfügung.

siehe auch: Internet-Suchmaschine: **Administrator Passwort vergessen**

B: Betriebssystem Windows XP, Windows Vista, Windows 7, Server 2003 und Server 2008

Bei den Betriebssysteme ab Windows XP führt die Umbenennung oder Löschung der SAM-Datei (SAM ... Security Access Management; %Systemroot%\winnt\system32\config\sam bzw. %Systemroot%\WINDOWS\system32\config\sam) nicht mehr zum Erfolg, um Zugriff auf Rechnerressourcen zu erhalten.

Für diese Betriebssysteme gibt es im Internet spezialisierte ISO-Images für CD-ROM's oder Windows-Programme.

1. NT Password Changer

Das ISO-Image kann auf der Webseite des Entwicklers Petter Nordahl-Hagen <http://home.eunet.no/~pnordahl/ntpasswd/> oder <http://pogostick.net/~pnh/ntpasswd/> oder im Internet (Suchmaschine: NT Password Changer) heruntergeladen werden.

Der NT Password Changer wurde bei den Betriebssystemen NT 3.51, NT 4, Windows 2000, Windows XP, Windows 2003 Server, Vista und Server 2008 erfolgreich getestet.

Die Oberfläche des NT Password Changer ist ein wenig gewöhnungsbedürftig, aber nach einigen Testläufen findet man sich relativ schnell zu recht.

Bei der Neuvergabe eines Passwortes für einen Benutzer mit Administratorrechten (nicht vom Benutzer »Administrator«) ist zu beachten, dass für diesen Benutzer ein leeres Passwort zu vergeben ist (Enter-Taste).

2. LessLinux Search and Rescue

LessLinux Search and Rescue ist ein bootfähiges System, dass von Matthias Schlenker entwickelt wurde.

LessLinux Search and Rescue ist hilfreich bei vielen Wartungs- und Rettungsarbeiten an Windows- und Linux-Rechnern. Der Start ist sowohl von USB-Stick oder -Festplatte als auch von einer bootfähigen CD oder DVD möglich.

Download: <http://blog.lesslinux.org/>

C: Betriebssystem Windows 7

Für Windows 7 gibt es noch eine alternative Methode um vergessene Passwörter neu zu vergeben.

1. Booten mit einem Fremdsystem von CD oder USB (z.B. Knoppix, PartedMagic)
2. \Windows\System32\Utilman.exe beliebig umbenennen z.B. UUUUtilman.exe (64-Bit Systeme: statt \Windows\System32 → \Windows\SysWOW64)
3. \Windows\System32\cmd.exe umbenennen in Utilman.exe (64-Bit Systeme: statt \Windows\System32 → \Windows\SysWOW64)

4. Windows starten: im linken unteren Bildschirmbereich kann die Utilman.exe (jetzt: CMD) über das entsprechende Icon aufgerufen werden
5. diese CMD besitzt höhere Rechte als der normale Administrator (siehe Befehl: **whoami**)
6. der Befehl **net user** zeigt alle lokalen Benutzer an
7. mit **net user <Benutzer> *** oder **net user <Benutzer> <neues Passwort>** ein neues Passwort vergeben
8. Booten mit einem Fremdsystem von CD oder USB (z.B. Knoppix, PartedMagic); die Umbenennung der beiden EXE-Dateien ist wieder rückgängig machen
9. Windows neu starten

D: Windows-Programm: grantAdminPriv (gAP)

Das Tool grantAdminPriv (gAP) kann sogar dem Benutzer Gast Administrator-Rechte zuweisen. Es muss dazu nur die Benutzerkontrolle aufgerufen werden.

siehe auch: www.youtube.com

E: Ergänzungen und Hinweise

1. Zugang zum Benutzerkonto des Benutzers »Administrator«

Windows XP: Fehlt unter XP die Eingabemaske für den Benutzer und dessen Passwort, so ist am Startbildschirm **zweimal** direkt hintereinander der 3-Finger-Salut ([Strg] + [Alt] + [Entf]) auf der Tastatur einzugeben.

Windows 7: Der Administrator von Windows 7 ist per Vorgabe deaktiviert, dieser kann aber von einem Benutzer mit Administratorrechten (CMD mit Administratorrechten aufrufen: Programme → Zubehör → Kommandozeile; rechte Maustaste etc.) wieder aktiviert werden.

net user Administrator /active:yes oder **no**

Nach der Aktivierung des Administrators ist dieser auf dem Startbildschirm wieder aufrufbar.

2. Aktivierungsschlüssel von Windows XP für Neuinstallation sichern

Um nach einer Neuinstallation die Produktaktivierung zu umgehen, ist nach der ersten Aktivierung die Datei WPA.DBL auf einem externen Datenträger

oder auf eine andere Partition zu sichern.

Die Datei ist vor der Produktaktivierung ca. 2 KB groß, nach der Aktivierung ca. 13 KB. Die Datei befindet sich im Ordner \Windows\System32.

Nach der nächsten Neuinstallation ist die Datei WPA.DBL durch die gesicherte Version zu überschreiben und schon hat man wieder ein aktiviertes Windows XP.

Das ganze funktioniert natürlich nur bei einer Neuinstallation auf dem gleichen Rechner, sofern bei der Hardware keine relevanten Veränderungen vorgenommen wurden.

3. Bootsektor von Windows XP wiederherstellen

Sie haben nach Windows XP ein weiteres Betriebssystem installiert und damit den MBR (Master Boot Record) überschrieben. Nun startet zwar das zuletzt installierte Betriebssystem aber Windows XP nicht mehr.

MBR von Windows XP wiederherstellen

Mit der Windows XP-CD starten, zur Wiederherstellungskonsole wechseln und folgende Befehle in dieser Reihenfolge eingeben:

FIXMBR C:

FIXBOOT C:

COPY x:\I386\NTLDR C:

COPY x:\I386\NTDETECT.COM C:

Ersetzen Sie das **x** in den Kopierbefehlen durch den Buchstaben des CD-ROM-Laufwerks und das **c** gegebenenfalls durch Ihren Laufwerksbuchstaben. Dadurch werden die Einstellungen des Bootloaders von Windows XP wieder zurückgesetzt.

4. Dateisystem von FAT zu NTFS konvertieren

Auch unter Windows XP ist es möglich eine bereits bestehende FAT-Festplatten- Partition in das NTFS - Dateisystem zu konvertieren. Der Befehl **convert** hat folgende Syntax:

Start → Ausführen → und **cmd** eingeben

convert c: /fs:ntfs /V ... »c« steht für das zu konvertierende Laufwerk!!

Nach einem Neustart des Rechners wird das Laufwerk in das NTFS-Dateisystem konvertiert - ohne Datenverlust.

siehe auch: INTERNET → Ntfsprogs-Paket. Das Paket enthält einige Programme zum Auslesen und Bearbeiten von NTFS formatierte Partitionen (ntfsls, ntfsnp, ntfsinfo, ...).

5. Windows 7 - Aktivierungsschlüssel sichern

1. Aktivierung sichern

Die zwei Dateien "tokens.dat" und "pkeyconfig.xrm-ms" müssen zuerst einmal gesichert werden (USB Stick, externe HDD etc ...). Die Dateien sind versteckt, also anzeigen lassen.

tokens.dat:

"c:\windows\ServiceProfiles\NetWorkService\AppData \Roaming\Microsoft\SoftwareProtectionPlatform"

pkeyconfig.xrm-ms:

"c:\windows\System32\spp\tokens\pkeyconfig"

2. Aktivierung wiederherstellen

Windows 7 installieren (ohne Eingabe des Produkt-Keys).

Im frisch installierten Windows 7 ist vorübergehend der DIENST "SoftwareProtection" zu deaktivieren.

Eingabeaufforderung (cmd) als Admin öffnen:

Dort "net stop sppsvc" eingeben und mit Enter bestätigen. Das Fenster nicht schließen!

Die beiden Dateien lassen sich nur ersetzen, wenn der DIENST "SoftwareProtection" beendet wurde, oder nicht aktiv läuft.

3. Dateirechte aneignen, Dateien löschen und gesicherte Dateien kopieren

Nun müssen die beiden Dateien gelöscht werden. Die Benutzerrechte der Dateien müssen auf einen selbst übertragen werden, damit sie gelöscht werden können. Nun sind die beiden Dateien zu löschen und durch die zuvor gesicherten Dateien zu ersetzen.

Falls eine 64 bit Windows Version eingesetzt wird, muss auch eine zweite Datei "pkeyconfig.xrm-ms" gelöscht werden. Diese befindet sich im Verzeichnis: "c:\windows\SysWOW64\spp\tokens\pkeyconfig".

4. Windows aktivieren

Nun kann der DIENST "Software Protection" wieder aktiviert werden. In der noch offenen Eingabeaufforderung ist folgender Befehl einzugeben:

"net start sppsvc" und mit Enter abschließen.

5. Aktivierung durchführen

Aktiviere nun Windows mit dem Befehl : "slmgr.vbs -ipk XXXXX-XXXXX-XXXXX-XXXXX".

Ersetze die X durch den Produkt-Key der Windows 7 Version. Bis sich das Bestätigungsfenster öffnet, vergehen einige Sekunden. Anschließend ist der Rechner neu zu starten.

Windows 7 ist nun aktiviert und muss nicht mehr über die Webseite der Redmonder Software-Schmiede aktiviert werden.

Einbruchserkennung - Claymore

Info-Stand: 2005

Das Aktualisieren von Programmpaketen dient dazu, Einbrüche dadurch zu erschweren, dass Sie Programme mit bekannten Sicherheitslücken durch korrigierte Versionen ersetzen.

Eine absolute Sicherheit vor Hackern kann aber auch dies nicht bieten. Wenn es zu Einbrüchen in den von Ihnen betreuten Systemen kommen sollte, sollten Sie diese möglichst schnell erkennen.

Einbrüche lassen sich nicht immer ganz einfach erkennen, da die Einbrecher oft Systemprogramme durch veränderte Versionen ersetzen. Beliebte Veränderungen an den Programmen ps und ls, damit diese die Verzeichnisse und Programme der Einbrecher nicht anzeigen.

Ein recht einfaches, aber durchaus wirkungsvolles System der Einbruchserkennung besteht daher darin, Prüfsummen der wichtigsten Systemdateien zu erstellen und diese regelmäßig zu vergleichen. Wenn Einbrecher Systemdateien verändern, ändern sich die Prüfsummen, was eindeutig auf einen Einbruch hinweist.

Das Programm Claymore zur Einbruchserkennung (intrusion detection), das Sie von <http://www.securityfocus.com/tools/1675> kostenlos laden können, ist empfehlenswert.

wget <http://www.securityfocus.com/data/tools/claymore03.tar.gz>

Das Perl-Programmpaket ist sehr klein. Entpacken Sie das Archiv mit

tar xvfz claymore.tar.gz

Dabei entsteht ein Verzeichnis claymore-0.3 (die Versionsnummer kann sich ändern), in das Sie mit

cd claymore-0.3

wechseln.

Kopieren Sie das Programm in das Verzeichnis /root/bin

cp claymore.pl /root/bin

Das Programm arbeitet mit zwei Dateien

light.list
light.db

Die erste Datei enthält eine Liste der zu überwachenden Programme mit vollständiger Pfadangabe und die zweite Datei zusätzlich die jeweiligen Prüfsummen. In diese Prüfsummen gehen sowohl der Dateiinhalt als auch das Dateidatum mit ein, so dass Veränderungen sofort zu erkennen sind.

Beide Dateien legt das Programm im Home-Verzeichnis des aufrufenden Benutzers ab, also in /root/claymore-0.3. Legen Sie also bitte dieses Verzeichnis an.

mkdir /root/claymore-0.3

Das Programm schlägt eine Liste der zu überwachenden Dateien vor, wenn Sie den Parameter **-m** mit angeben. Diese Liste können Sie so an die richtige Stelle bringen:

/root/bin/claymore.pl -m > /root/claymore-0.3/light.list

Dann müssen Sie die Datei mit den Prüfsummen initialisieren:

/root/bin/claymore.pl -r

Das dauert jetzt etwas, da sehr viele Dateien in der Liste stehen.

Jedes Mal, wenn Sie hinfert

/root/bin/claymore.pl

aufrufen, erzeugt das Programm für jede Datei in der light.list eine Prüfsumme und vergleicht diese mit dem in der Datei light.db gespeicherten Wert.

Sowie es eine Abweichung gibt, warnt Claymore an der Konsole und an die konfigurierbare Mail-Adresse.

In dem Programm können Sie ein paar Einstellungen leicht verändern, vor allem den Mail-Empfänger für die Virenwarnungen.

claymore.pl (Auszug ab Zeile 21)

info to

```

customize
$USER = "; # (optional) address to email warnings, try 'root@localhost'
##### PATHs,
these should be adjusted to match your system
$DB_FILE = "$ENV{'HOME'}/claymore-$::VERSION/light.db";
$LIST_FILE = "$ENV{'HOME'}/claymore-$::VERSION/light.list";
$MAIL = '/bin/mail';

```

Geben Sie in der Variablen \$USER eine sinnvolle Mail-Adresse für die Warnungen an, möglichst eine auf einem anderen Rechner!

Um Einbrechern das Auffinden des Programms zu erschweren, sollten Sie die Dateinamen für die Listen und das Programm selbst ändern.

Wenn das Programm zu Ihrer Zufriedenheit konfiguriert ist, sollten Sie es per Crontab regelmäßig aufrufen lassen. Mit

05 * * * * /root/bin/claymore.pl

veranlassen Sie eine stündliche Überprüfung der Systemdateien. Sie müssen nun aber bei jedem Online-Update daran denken, dass Sie die Datenbank von Claymore mit

/root/bin/claymore.pl -r

neu erzeugen, da Sie sonst nach dem Update stündlich eine Fehlermeldung bekommen.

Hinweis: Auch das Programm Claymore und ähnliche Programme bieten keine absolute Sicherheit. Allein schon diese Beschreibung macht das System unsicherer, weil bekannter.

Festplatten und Partitionen mit Ghost for Linux (g4l) sichern

In den nachfolgenden Zeilen wird davon ausgegangen, dass ausreichend Wissen über Laufwerksbezeichnungen (Festplatten, Partitionen) unter Linux und über die verschiedenen Dateisysteme (NTFS, VFAT, EXT3, EXT4 ...) vorhanden sind.

G4L (Ghost for Linux) ist Bestandteil der Live-CD **PartedMagic**. Die ISO-Datei finden Sie auf der Webseite <http://partedmagic.com>. Die heruntergeladene ISO-Datei können Sie dann mit einem beliebigen Brennprogramm als Abbild auf CD brennen.

Hinweis: G4L (Ghost for Linux) gibt es auch als Live-CD. Das ISO-Image bekommt man unter anderen von Sourceforge.net (Internetadresse: <http://sourceforge.net/projects/g4l/files/g4l%20ISO%20images/>).

Der Funktionsumfang von G4L reicht vom Klonen einer Partition oder Festplatte bis zum Erstellen von komprimierten Partitionsabbildern, die auf einem FTP-Server gespeichert werden können.

Im nachfolgenden Beispiel wird davon ausgegangen, dass im Rechner 2 Festplatten eingebaut sind. Die 1. Festplatte wird als komprimiertes Image (mit Master Boot Record und allen Partitionstabellen) auf der 2. Festplatte gesichert. Als 2. Festplatte kann auch ein ausreichend bemessener externer Datenträger (USB-Festplatte; Dateisystem: EXT3, EXT4, etc.) verwendet werden. Das Dateisystem auf der USB-Festplatte sollte ein Dateigröße von mehr als 4 Gbyte erlauben, damit scheidet das Dateisystem VFAT aus.

1. Backup und Restore von einer Festplatte (hda)

Die Vorgehensweise für die Erstellung eines Backup-Images bzw. für die Wiederherstellung (Restore) einer Festplatte ist im ersten Teil dieselbe.

Starten Sie den Rechner mit der PartedMagic-Live-CD. Nach erfolgreichem Start können Sie die CD entfernen, denn PartedMagic befindet sich nun vollständig im Hauptspeicher des Rechners.

Als Erstes ändern Sie das Tastaturlayout durch einen Mausklick auf das entsprechende Icon auf dem Desktop.

Achtung: Sie besitzen volle Rechte, d.h. SIE SOLLTEN WISSEN WAS SIE TUN.

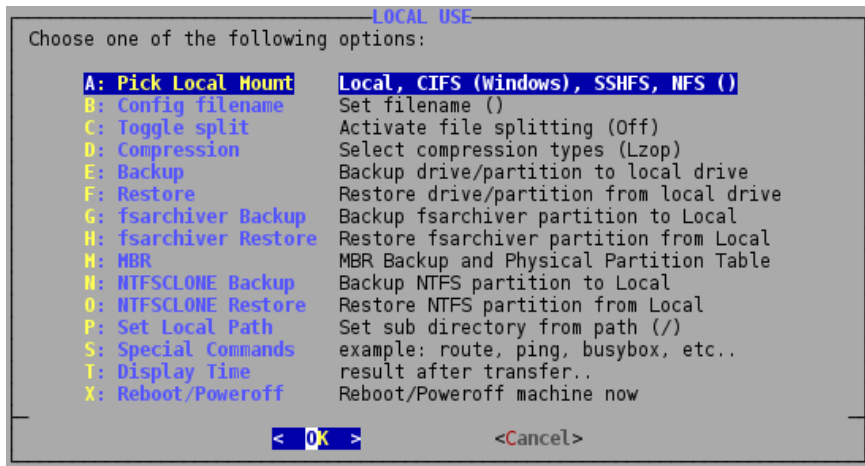
G4L starten Sie innerhalb eines Terminals mit dem Befehl **g4l**, außerdem

finden Sie G4L im Hauptmenü unter den »System Tools«.
 Nach dem Begrüßungsbildschirm von G4L, wählen Sie in den einzelnen Menüs jeweils eine der nachfolgenden Zeilen aus. Die Reihenfolge der aufgelisteten Menüzeilen ist einzuhalten.

Die Navigation innerhalb eines Menüs erfolgt mittels der Pfeiltasten.

- **RAW Mode** **ANY filesystem, every bit, local+ftp**
 Hinweis: Beim RAW-Mode brauchen Sie sich um das verwendete Dateisystem ext3, ext4, FAT, NTFS etc. keine Gedanken machen. G4L sichert Partitionen im RAW-Mode als blockweises Abbild.
- **Local use** **Backup/Restore to/from local drive**

Nach dem zweiten Auswahlmenü sehen Sie jetzt das Hauptmenü von G4L (Ghost for Linux). Für das Backup bzw. für das Restore der gesamten Festplatte interessieren uns nur die Punkte A, B, E und F.
 Bei den Punkten C und D bleiben die Standardeinstellungen von G4L unverändert.



2. Backup: Erstellung eines Images der gesamten 1. Festplatte

- **A: Pick Local Mount** **Local, CIFS (Windows), SSHFS, NFS ()**
 Wo soll gespeichert werden? In den nachfolgenden Menüs wählen Sie das Laufwerk aus, auf dem das Backup-Image gespeichert werden soll.

Die Navigation innerhalb eines Menüs erfolgt mittels der Pfeiltasten. Für die Auswahl der Partition auf der 2. Festplatte (hdb1) benutzen Sie bitte die Leertaste.

- **B: Config filename Set filename ()**
Geben Sie dem Backup-Image einen aussagekräftigen Namen!
Beispiel: linux_mint_15-64_bit-hda-2014-04-01.img
- **C: Toggle split Activate file splitting (Off)**
keine Änderungen
- **D: Compression Select compression types (Lzop)**
keine Änderungen
- **E: Backup Backup drive/partition to local drive**
Was soll gespeichert werden? In den nachfolgenden Menüs wählen Sie das Laufwerk aus, von dem das Backup-Image erstellt werden soll.

Die Navigation innerhalb eines Menüs erfolgt mittels der Pfeiltasten. Für die Auswahl der gesamten 1. Festplatte (hda – ohne Nummer) benutzen Sie bitte die Leertaste.

Anschließend werden Sie von G4L gefragt:

Are you sure?

Sind Sie sicher?

Überprüfen Sie die Angaben sorgfältig und starten dann das Backup.

- **G4L is backing up your drive.**

Sie können sich jetzt einige Minuten zurücklehnen ...

Das Backup-Image von der gesamten 1. Festplatte (mit Master Boot Record und allen Partitionstabellen) finden Sie im unseren Beispiel im Wurzelverzeichnis der 2. Festplatte (hdb).

3. Restore: Wiederherstellung der 1. Festplatte durch ein Backup-Image

- **A: Pick Local Mount Local, CIFS (Windows), SSHFS, NFS ()**
Wo liegt das Backup-Image? In den nachfolgenden Menü wählen Sie das Laufwerk aus, auf dem das Backup-Image für die 1. Festplatte gespeichert ist.

Die Navigation innerhalb eines Menüs erfolgt mittels der Pfeiltasten. Für die Auswahl der Partition auf der 2. Festplatte (hdb1) benutzen Sie bitte die Leertaste.

- **B: Config filename Set filename ()**
Wählen Sie das zu verwendende Backup-Image für die 1. Festplatte aus.
Beispiel: linux_mint_15-64_bit-hda-2014-04-01.img
- **C: Toggle split Activate file splitting (Off)**
keine Änderungen
- **D: Compression Select compression types (Lzop)**
keine Änderungen
- **F: Restore Restore drive/partition from local drive**
Was soll überschrieben, wiederhergestellt werden? In den nachfolgenden Menü wählen Sie das Laufwerk aus, dass durch das Backup-Image wiederhergestellt wird.

Die Navigation innerhalb eines Menüs erfolgt mittels der Pfeiltasten. Für die Auswahl der Partition auf der 1. Festplatte (hda – ohne Nummer) benutzen Sie bitte die Leertaste.

Anschließend werden Sie von G4L gefragt:

Are you sure?

Sind Sie sicher?

Überprüfen Sie die Angaben sorgfältig und starten dann das Restore (Wiederherstellung der 1. Festplatte).

G4L is locally restoring your drive.

Sie können sich jetzt einige Minuten zurücklehnen ...

Die 1. Festplatte (hda) wurde mit dem Master Boot Record und

allen Partitionstabellen durch das Backup-Image komplett überschrieben und damit auch wiederhergestellt.

Ergänzende Informationen

1. MBR – Master Boot Record

Der Master Boot Record (MBR) ist der erste Datenblock eines in Partitionen aufgeteilten, bootfähigen Speichermediums.

Dieser Bereich der Festplatte, der sich im ersten Sektor der Platte befindet, ist so aufgeteilt, dass darin sowohl der Bootcode (Boot-Loader), die MBR-Signatur und die Partitionstabelle der Festplatte untergebracht sind. Die Aufgabe des Bootcodes besteht nun entweder darin,

- das Betriebssystem zu starten, das sich auf der als aktiv markierten Partition befindet,
- oder dem Benutzer zunächst die Möglichkeit zu bieten, eines von mehreren Betriebssystemen – oder einen von mehreren Kernen – auszuwählen und zu starten.

Ist eine MBR-Signatur vorhanden, so geht das BIOS davon aus, dass ein gültiger Master Boot Record vorhanden ist. Wird die Signatur nicht gefunden, vermutet das BIOS einen neuen bzw. gelöschten Datenträger. Der Bootvorgang wird dann abgebrochen und eine Fehlermeldung ausgegeben. Natürlich ist eine korrekte Signatur keine Garantie für gültigen Boot-Code im MBR. Sie dient lediglich dazu, zu verhindern, dass leere MBRs oder MBRs mit Zufallsdaten ausgeführt werden.

2. Extended Partition Boot Record (EPBR oder EBR)

Jede logische Partition entspricht einer physischen Festplatte und auf jeder logischen Festplatte nimmt ein EBR die gleiche Position wie der MBR auf einer physischen Festplatte ein.

Für die erweiterte Partition, die alle logischen Laufwerke beinhaltet, wird in der im Master Boot Record abgelegten Partitionstabelle lediglich die nächste erweiterte Partitionstabelle adressiert. Dieser Extended Boot Record (EBR), enthält dann die Adresse des ersten logischen Laufwerks und den nächsten Extended Boot Record (EBR) eines weiteren logische Laufwerkes. Das ganze kann dann im Prinzip immer so weitergehen und wird lediglich durch den Platz auf der Platte bzw. durch die verfügbaren Buchstaben für die Laufwerke limitiert. Demnach können durch diese »verkettete Liste« im Prinzip beliebig viele logische Partition adressiert werden.

Jedem logischem Laufwerk geht demnach ein EBR voraus und die Adresse

des ersten EBR ist im MBR hinterlegt.

3. G4L und der Master Boot Record

Wird mit G4L die gesamte Systemfestplatte (hda – ohne Nummer) gesichert, so wird auch der Master Boot Record (MBR) und ein evtl. vorhandener Extended Partition Boot Record (EBR) ebenfalls mitgesichert.

Wird mit G4L aber nur eine Partition (hda1) gesichert, so muss der MBR in einem zweiten Backup-Image zusätzlich gesichert werden (siehe G4L-Menü: MBR). Gibt es auf der Festplatte neben einer primären Partition noch eine erweiterte Partition, so ist der EBR ebenfalls noch in einem zusätzlichen Backup-Image zu sichern.

Verkleinerung des Backup-Images durch Ausnullen

G4L erstellt im RAW-Mode von der Festplatte ein blockweises Abbild. Dies bedeutet, die vom System freigegebene Sektoren (gelöschte Daten) werden nicht als leer interpretiert. Mit dem Ausnullen dieser »leeren« Bereiche wird erreicht, dass G4L die freigegebenen Sektoren auf der Festplatte bedeutend stärker komprimieren kann.

Linux: Ausnullen mit dd

Die nulldatei.000 kann so groß werden, wie der leere Bereich der aktuellen Partition werden. Hat die Datei ihre maximale Größe erreicht bricht das System den Vorgang mit einer Fehlermeldung ab. Anschließend wird diese Datei, nach dem Synchronisieren, gelöscht.

dd if=/dev/zero of=nulldatei.000

sync

rm nulldatei.000

sync ... Normalerweise verwendet Linux einen Puffer (Cache) im Arbeitsspeicher, in dem sich ganze Datenblöcke eines Massenspeichers befinden. So werden Daten häufig temporär erst im Arbeitsspeicher verwaltet, da sich ein dauernd schreibender Prozess äußerst negativ auf die Performance des Systems auswirken würde.

Mit dem Kommando sync können Sie nun veranlassen, dass veränderte Daten sofort auf die Festplatte (oder auf jeden anderen Massenspeicher) geschrieben werden.

Windows: Ausnullen mit einer Batchdatei

Datei: ausnullen.bat

[illegible]

Die Datei ausnullen.bat ist an der Kommandozeile (cmd) aufzurufen. Die Abarbeitung der Befehlsreihe, kann je nach Größe der Partition einige Zeit in Anspruch nehmen.

Die `ausnullen.bat` schreibt solange Nullen in die `nulldatei.000` bis der

Leerbereich der Partition gefüllt ist und das System die Abarbeitung der Befehle abbricht. Anschließend wird die Datei nulldatei.000 und die Datei 10 durch die Batchdatei wieder gelöscht.

Alternativ kann man auch an der Windows-Kommandozeile (cmd) das Programm cipher.exe (falls vorhanden) aufrufen:

CIPHER /W:C:

Der Laufwerksbuchstabe C: ist gegebenenfalls anzupassen. Das Programm cipher.exe verschlüsselt die gelöschten aber noch vorhandenen Daten in den freigegebenen Sektoren. In der ersten Runde werden diese Sektoren ausgenullt und an dieser Stelle ist das Programm cipher.exe einfach abzubrechen ([Strg] + [C]), bevor es mit der Verschlüsselung der freigegebenen Sektoren beginnt.

Partimage – eine Alternative

Partimage (Installation: `sudo apt-get install partimage`) ist ein Kommandozeilenprogramm, mit dem sich ganze Partitionen sichern lassen.

Durch Partimage werden nur benutzte Sektoren der Partition gesichert und die Daten können auch komprimiert werden. Die entstehenden Image-Dateien benötigen daher weniger Speicherplatz als die zu sichernde Partition selbst. Auch Windows-Partitionen können gesichert werden. Es können nur Partitionen gesichert werden, die nicht gerade im schreibbaren Zustand verwendet werden. Eine Sicherung der Systempartition ist daher ohne weiteres nur von einem separaten Wartungssystem oder einer Live-CD (z.B. LessLinux, Knoppix) aus möglich.

Partimage wird in einem Terminal aufgerufen:

sudo partimage

Die grafische Oberfläche wird ausschließlich per Tastatur bedient. Im ersten Schritt muss zunächst mit den Pfeiltasten die zu sichernde Partition gewählt werden. Dann geht es mit der Tab-Taste weiter zur Eingabe von Ort und Namen der Sicherungsdatei. Für eine externe USB-Festplatte könnte die Pfadangabe z.B. so aussehen: `/mnt/sdb1/partimage.gz`, wobei partimage.gz hierbei den frei wählbaren Dateinamen und die Komprimierung (gzip) angibt. Die Laufwerksbezeichnung kann aus der Auflistung der von Partimage erkannten Partitionen entnommen werden.

siehe auch: man partimage

Tipps und Tricks

- **abgestürzte KDE-Kontrollleiste neu starten**

Normale Programme die abgestürzt sind ruft man über die Kontrollleiste einfach wieder neu auf. Aber was tun, wenn die Kontrollleiste abgestürzt ist?

KDE: Schnellstartfenster über Tastenkombination **[Alt] + [F2]** öffnen und **kicker** eingeben.

- **abgestürztes Gnome-Panel neu starten**

Tauchen im Gnome-Panel fehlerbehaftete Anzeigen auf, so kann man das Panel mit **pkill** beenden. Anschließend wird das Gnome-Panel automatisch neu gestartet.

Gnome: **pkill gnome-panel** in ein Terminal eingeben

- **abgestürzter Cinnamon-Desktop neu starten**

Funktioniert etwas auf dem Cinnamon-Desktop von Linux-Mint nicht so wie soll, so kann man den Desktop neu starten - ohne ein System-Neustart durchzuführen.

Cinnamon: Schnellstartfenster über Tastenkombination **[Alt] + [F2]** öffnen und den Buchstaben **r** eingeben.

- **Kwrite: Text in Groß- oder Kleinbuchstaben umwandeln**

Text in **Großbuchstaben** umwandeln: Den Text markieren oder um den gesamten Text zu markieren, die Tastenkombination **[Strg] + [a]** drücken und anschließend die Tastenkombination **[Strg] + [u]** drücken.

Text in **Kleinbuchstaben** umwandeln: Den Text markieren oder um den gesamten Text zu markieren, die Tastenkombination **[Strg] + [a]** drücken und anschließend die Tastenkombination **[Strg] + [Umschalt] + [u]** drücken.

- **Programm-Fenster verschieben und skalieren**

Fast jeder Benutzer hat es schon einmal geschafft, ein Fenster so zu platzieren, dass er an die Fensterleiste nicht mehr herankommt. Um das Fenster trotzdem zu verschieben, drücken Sie die Taste **[Alt]** und klicken

mit der **linken Maustaste** in das Fenster, dass Sie nun verschieben können, solange Sie die linke Maustaste gedrückt halten.

- **KDE: Tastenkombinationen**

[Alt] + [F1] ... Kontrollleiste → Aufklappmenü (Startmenü) öffnen

[Alt] + [F2] ... Befehl ausführen

[Alt] + [F3] ... Fenster-Aktions-Menü öffnen; mit [Alt] wieder schließen

[Alt] + [F4] ... aktuelles Fenster schließen

[Alt] + [F5] ... Fensterliste anzeigen; mit [Alt] wieder schließen

[Alt] + [F12] ... Mausemulation, d.h. Bedienung der Maus über die Pfeiltasten; mit [Alt] Emulation wieder beenden

[Alt] + [Tab] ... zwischen den Fenstern wechseln

[Alt] + [Druck bzw. Print] ... Bildschirmfoto vom aktuellen Fenster erstellen, anschließend mit [Strg] + [v] das Bildschirmfoto z.B. im Home-Verzeichnis einfügen - vor dem endgültigen Einfügen ist über ein Dialogfenster noch der Dateiname zu vergeben → z.B. Snapshot1.png (PNG ist das Standardformat)

[Strg] + [Druck bzw. Print] ... Bildschirmfoto von der Arbeitsfläche, Desktop erstellen, anschließend mit [Strg] + [v] das Bildschirmfoto z.B. im Home-Verzeichnis einfügen - vor dem endgültigen Einfügen ist über ein Dialogfenster noch der Dateiname zu vergeben → z.B. Snapshot1.png (PNG ist das Standardformat)

[Alt] + [Strg] + [d] ... Arbeitsfläche anzeigen an/aus

[Alt] + [Strg] + [L] ... aktuelle Sitzung sperren

[Strg] + [Esc] ... Prozessmanager anzeigen

[Strg] + [Alt] + [Backspace bzw. Rückschritt] ... Reagiert der Desktop auch nach einer Wartezeit nicht, »schießen« Sie die grafische Oberfläche mit dieser Tastenkombination ab. Haben Sie damit Erfolg, wird der Bildschirm schwarz, und Linux zeigt nach wenigen Sekunden den Login-Manager an, wo Sie sich erneut anmelden können.

[Strg] + [Alt] + [Esc] ... Verwandelt den Mauszeiger in einen Totenkopf, mit dem man Programm-Fenster die nicht mehr reagieren wollen - gewaltsam »abschießt«.

[Strg] + [Alt] + [Entf] ... öffnet den KDE-Abmeldedialog.

[Strg] + [Alt] + [d] ... minimiert alle Fenster bzw. stellt sie wieder her.

weitere Tastenkürzel finden Sie im Kontrollzentrum:

«Kontrollzentrum» «Regionaleinstellungen & Zugangshilfen»
«Tastenkürzel»

- **Unity: Tastenkombinationen unter Ubuntu**

[Super] ... Unitys Launcher starten

[Super] ... Super-Taste längere Zeit gedrückt halten – kurze Übersicht über die Tastaturkürzel anzeigen
[Super] + [d] ... alle Fenster minimieren beziehungsweise wiederherstellen
[Super] + [1-0] ... entsprechende Anwendung in der Seitenleiste öffnen
[Super] + [Shift] + [1-0] ... Entsprechende Anwendung öffnen, wenn bereits gestartet
[Super] + [a] ... Applikations-Menü öffnen
[Super] + [s] ... alle Arbeitsflächen zeigen
[Super] + [f] ... Dateien und Ordner öffnen
[Super] + [t] ... Papierkorb öffnen
[Super] + [w] ... alle Fenster im Überblick (Expo-Modus)
[Alt] + [F1] ... öffnet den Launcher und verwendet Pfeiltasten zum Navigieren
[Alt] + [F2] ... einen Befehl ausführen
[Strg] + [Alt] + [t] ... Terminal öffnen
[Strg] + [Alt] + [Numpad 1] ... Fenster in der unteren linken Hälfte platzieren (mehrfache Ausführung ändert Schrittweise die Größe, gilt für 1 bis 0)
[Strg] + [Alt] + [Numpad 2] ... Fenster in der unteren Hälfte platzieren
[Strg] + [Alt] + [Numpad 3] ... Fenster in der unteren rechten Hälfte platzieren
[Strg] + [Alt] + [Numpad 4] ... Fenster in der linken Hälfte platzieren
[Strg] + [Alt] + [Numpad 5] ... Fenster zentrieren beziehungsweise maximieren
[Strg] + [Alt] + [Numpad 6] ... Fenster in der rechten Hälfte platzieren
[Strg] + [Alt] + [Numpad 7] ... Fenster in der oberen linken Hälfte platzieren
[Strg] + [Alt] + [Numpad 8] ... Fenster in der oberen Hälfte platzieren
[Strg] + [Alt] + [Numpad 9] ... Fenster in der oberen rechten Hälfte platzieren
[Strg] + [Alt] + [Numpad 0] ... Fenster maximieren
[Strg] + [Alt] + [→] ... eine Arbeitsfläche nach rechts
[Strg] + [Alt] + [←] ... eine Arbeitsfläche nach links
[F10] ... erstes Menü im Panel öffnen
[Druck] ... Screenshot / Schnappschuss des gesamten Bildschirms (bei Notebooks ist zusätzlich die Taste [Fn] zu betätigen: [Fn] + [Druck])
[Alt] + [Druck] ... Screenshot / Schnappschuss des aktiven Fensters (bei Notebooks ist zusätzlich die Taste [Fn] zu betätigen: [Alt] + [Fn] + [Druck])

Dateimanager »Nautilus«

[Strg] + [N] ... neues Programmfenster öffnen
[Strg] + [T] ... neuen Reiter, Tab öffnen
[Strg] + [W] ... aktuelles Programmfenster schließen
[Alt] + [F4] ... aktuelles Programmfenster schließen
[Strg] + [Q] ... alle Programmfenster schließen
[Alt] + [↑] ... ins übergeordnete Verzeichnis wechseln
[Alt] + [←] ... zurück; ins vorherige Verzeichnis wechseln
[Strg] + [F] ... nach Dateien suchen
[Strg] + [H] ... versteckte Verzeichnisse und Dateien (Verzeichnis- und Dateinamen die mit einem Punkt beginnen) anzeigen / verbergen
[Strg] + [+] ... Symbole für Verzeichnisse und Dateien vergrößern
[Strg] + [-] ... Symbole für Verzeichnisse und Dateien verkleinern
[Strg] + [0] ... Symbole für Verzeichnisse und Dateien in Originalgröße anzeigen
[Strg] + [L] ... Adressleiste anzeigen, für die direkte Eingabe von Verzeichnispfaden und Serveradressen (FTP, SSH);
 Verzeichnispfad: **/var/log**
 FTP-Server: **ftp://Benutzername@ftp.servername.de**
 SSH-Server: **ssh://Benutzername@ssh.servername.de**

Tricks mit der Maus

Unity lässt sich nicht nur mittels Tastenkombinationen / Shortcuts schneller bedienen. Auch mit der Maus können Sie einiges bewirken.

Mauszeiger in obere linke Ecke ... Launcher / Startleiste einblenden, wenn versteckt

Fenster zum oberen Bildrand ziehen ... Fenster maximieren

Mittlere Maustaste auf Maximier-Schaltfläche des Fensters ... Höhe maximieren

Rechte Maustaste auf Maximier-Schaltfläche des Fensters ... Breite maximieren

Fester zum rechten Bildrand ziehen ... in der rechten Hälfte platzieren

Fester zum linken Bildrand ziehen ... in der linken Hälfte platzieren

- **Cinnamon: Tastenkombinationen unter Linux Mint**

[Super] ... Hauptmenü öffnen

[Strg] + [Alt] + [Entf] ... Benutzer abmelden

[Alt] + [Strg] + [L] ... aktuelle Sitzung sperren oder

[Strg] + [Alt] + [L] ... aktuelle Sitzung sperren

[Super] + [d] ... alle Fenster minimieren / wiederherstellen

[Strg] + [Alt] + [→] ... eine Arbeitsfläche nach rechts

[Strg] + [Alt] + [←] ... eine Arbeitsfläche nach links

[Strg] + [Alt] + [↑] ... Übersicht aller Arbeitsbereiche anzeigen
[Strg] + [Alt] + [↓] ... Übersicht aller Programmfenster anzeigen
[Alt] + [F1] ... Übersicht aller Arbeitsbereiche anzeigen / ausblenden
[Alt] + [F4] ... aktuelles Fenster schließen
[Alt] + [F2] ... einen Befehl ausführen (z.B. gksu nemo)
[Strg] + [Alt] + [t] ... Terminal öffnen
[Strg] + [Alt] + [Numpad 1] ... Fenster in der unteren linken Hälfte platzieren (mehrfache Ausführung ändert Schrittweise die Größe, gilt für 1 bis 0)
[Strg] + [Alt] + [Numpad 2] ... Fenster in der unteren Hälfte platzieren
[Strg] + [Alt] + [Numpad 3] ... Fenster in der unteren rechten Hälfte platzieren
[Strg] + [Alt] + [Numpad 4] ... Fenster in der linken Hälfte platzieren
[Strg] + [Alt] + [Numpad 5] ... Fenster maximieren
[Strg] + [Alt] + [Numpad 6] ... Fenster in der rechten Hälfte platzieren
[Strg] + [Alt] + [Numpad 7] ... Fenster in der oberen linken Hälfte platzieren
[Strg] + [Alt] + [Numpad 8] ... Fenster in der oberen Hälfte platzieren
[Strg] + [Alt] + [Numpad 9] ... Fenster in der oberen rechten Hälfte platzieren
[Strg] + [Alt] + [Numpad 0] ... Fenster minimieren
[Alt] + [Tab] ... Fensterliste anzeigen
[F10] ... erstes Menü eines geöffneten Programmfensters öffnen / schließen

Terminalfenster

[Strg] + [+] ... aktiviertes Terminalfenster und Schrift vergrößern; Plus-Zeichen auf dem Haupttastenblock verwenden, **nicht** den Nummern-Block; stufenweises Vergrößern durch mehrmaliges Betätigen der 2-Tastenkombination
[Strg] + [-] ... aktiviertes Terminalfenster und Schrift verkleinern; Minus-Zeichen auf dem Haupttastenblock verwenden, **nicht** den Nummern-Block; stufenweises Verkleinern durch mehrmaliges Betätigen der 2-Tastenkombination
[Strg] + [0] ... aktiviertes Terminalfenster und Schrift in Originalgröße anzeigen; die Null (0) auf dem Haupttastenblock verwenden, **nicht** den Nummern-Block

Dateimanager »Nemo«

[Strg] + [N] ... neues Programmfenster öffnen
[Strg] + [T] ... neuen Reiter, Tab öffnen

[Strg] + [W] ... aktuelles Programmfenster schließen
[Alt] + [F4] ... aktuelles Programmfenster schließen
[Strg] + [Q] ... alle Programmfenster schließen
[Alt] + [↑] ... ins übergeordnete Verzeichnis wechseln
[Alt] + [←] ... zurück; ins vorherige Verzeichnis wechseln
[Strg] + [F] ... nach Dateien suchen
[Strg] + [H] ... versteckte Verzeichnisse und Dateien (Verzeichnis- und Dateinamen die mit einen Punkt beginnen) anzeigen / verbergen
[Strg] + [+] ... Symbole für Verzeichnisse und Dateien vergrößern
[Strg] + [-] ... Symbole für Verzeichnisse und Dateien verkleinern
[Strg] + [0] ... Symbole für Verzeichnisse und Dateien in Originalgröße anzeigen

[Strg] + [L] ... Adressleiste anzeigen, für die direkte Eingabe von Verzeichnispfaden und Serveradressen (FTP, SSH);

Verzeichnispfad: **/var/log**

FTP-Server: **ftp://Benutzername@ftp.servername.de**

SSH-Server: **ssh://Benutzername@ssh.servername.de**

Tricks mit der Maus

Mauszeiger in obere linke Ecke ... Übersicht aller Arbeitsbereiche anzeigen

- **Fensterverwaltung – ein Programm per Mausklick beenden**
 Falls das Programm xkill nicht die gewünschte Wirkung zeigt (z.B. bei Skripts, die auch ohne grafisches Frontend im Hintergrund weiter laufen), so kann folgendes in einem Terminal versucht werden.

```
kill -9 $(xprop _NET_WM_PID | sed -ne 's/[^0-9]*\([0-9]\+\)/\1/p')
```

Das Programm xprop ermittelt mit Hilfe von sed die Prozesskennung.

Beim Aufruf dieser Befehlskette verwandelt sich den Mauszeiger in ein Fadenkreuz. Mit einem Mausklick in das nicht mehr reagierende Programmfenster, wird das Programm mit einem kill -9 Signal auf die harte Tour beendet.

- **Namen eines laufenden Programms mit xprop ermitteln**
 Eine gezielte Möglichkeit, den Namen eines laufenden Programms zu ermitteln, ist der folgende Terminal-Befehl:

xprop | awk '/CLASS/'

Wenn man den in ein Kreuz verwandelten Mauszeiger nun in das Programmfenster setzt, so zeigt das Terminal den Programmnamen.

- **Batteriekapazität und Stromverbrauch des Notebooks anzeigen**

watch -n 1 cat /proc/acpi/battery/*/state

Die Angaben werden im Sekundentakt aktualisiert. Das Programm wird über die Tastenkombination [Strg] + [C] beendet.

Notizen

Stichwortverzeichnis

- 3
- 32-Bit-CPU ... 270
- 3D-Beschleunigung ... 138
- 3D-Konfiguration ... 138
- 3Ddiag ... 138
- 6
- 64-Bit-CPU ... 270
- 64-Bit-Erweiterungen ... 270
- 7
- 7z ... 248
- 7zr ... 248
- A
- ab ... 12
- abgestürzte KDE-Kontrollleiste neu starten ... 553
- abgestürzter Cinnamon-Desktop neu starten ... 553
- abgestürztes Gnome-Panel neu starten ... 553
- Absturz der grafischen Oberfläche ... 6
- Access Points ... 167
- Address Resolution Protocol ... 236
- Administrator-Passwort vergessen ... 531
- AES256 ... 392
- agrep ... 86
- Aktivierungsschlüssel ... 538, 540
- Alias-Definition ... 315
- Anmeldeversuche ... 120
- antiword ... 12
- Apache Benchmark ... 12
- aplay ... 299
- apropos ... 18, 175
- apt-cache ... 15
- apt-get ... 13, 270
- Arithmetik ... 497, 515
- arp ... 18
- ARP ... 236
- ARP-Cache-Poisoning-Attacken ... 168
- Arrays ... 507
- ASCII-Zeichen in Hexadezimalschreibweise ... 468
- at ... 17
- atool ... 419
- Audacity ... 417
- Audiostream einer DVD in eine MP3-Datei umwandeln ... 226
- aunpack ... 419
- Ausführungsberechtigung ... 409
- Ausnullen ... 550
- Authentifizierungs- und sonstige Weiterleitung ... 308
- Autorisierung über ssh ... 436
- Autostart mit KDE ... 23
- Autostart mit ROOT-Rechten ... 20
- avconv ... 23
- avconv2theora ... 32
- B
- Backticks ... 501
- Backup mit tar ... 436
- backup_incremental.sh ... 443
- backup_tar.sh ... 439
- basename ... 56
- Batteriekapazität und Stromverbrauch des Notebooks anzeigen ... 559
- bc ... 56
- Bearbeitung von Texten mit sed ... 284
- Bekanntmachung des Schlüssels und Widerruf – Artikel Nr. 4 ... 389
- Benutzer-Login ... 322
- Benutzergruppe ... 409
- Benutzerverwaltung ... 368
- Bibliotheken ... 197
- Bilder erstellen und bearbeiten ... 466
- Bildschirmauflösungen ... 406
- Bildschirmfotos ... 172

| | | | |
|-------------------------------|--------------|--|--------------|
| Bildschirmschoner | ... 407 | clear | ... 72 |
| BIOS | ... 57 | CMOS-Uhr | ... 79 |
| blkid | ... 57 | COMMAND | ... 264 |
| Blockgröße | ... 314 | Command Substitution | ... 501 |
| Blowfish | ... 185 | continue | ... 514 |
| BLOWFISH | ... 392 | convert | ... 170, 466 |
| BLOWFISH-Verschlüsselung | ... 267 | cp | ... 73 |
| boot.local | ... 23 | CPU | ... 269 |
| Bootloader | ... 138 | cpuinfo | ... 76 |
| Bootsektor | ... 539 | cron | ... 17 |
| break | ... 515 | Cronjobs einrichten | ... 448 |
| bsd-mailx | ... 209 | crontab | ... 73 |
| C | | crypt | ... 76 |
| cal | ... 58 | Crypt-Funktion | ... 432 |
| Calibre | ... 417 | curlftpfs | ... 76 |
| case | ... 515 | cut | ... 76 |
| CAST5 | ... 391 | D | |
| cat | ... 58 | Darktable | ... 416 |
| cat /proc/cpuinfo | ... 269 | Das kleine GnuPG Intro - Artikel Nr. 2 | ... 381 |
| cat /proc/filesystems | ... 269 | date | ... 78 |
| cat /proc/interrupts | ... 269 | Datei löschen | ... 276 |
| cat /proc/pci | ... 269 | Dateien umbenennen und | |
| cat /proc/version | ... 269, 374 | Zeichenersetzungen | ... 460 |
| cclive | ... 61 | Dateien: versehentlich gelöschte | |
| cd | ... 59 | Dateien | ... 80 |
| cd ~ | ... 281 | Dateisysteme | ... 104, 269 |
| cdparanoia | ... 300, 462 | Datenspuren | ... 84 |
| CDs und DVDs erstellen | ... 59 | DCF77 | ... 80 |
| CentOS | ... 106, 271 | dd | ... 87 |
| chgrp | ... 63 | ddrescue | ... 101 |
| chkconfig | ... 107 | DEB-Paketen | ... 270 |
| chmod | ... 62, 465 | Debian | ... 105, 270 |
| chown | ... 63 | DES | ... 185 |
| chpasswd | ... 60 | DES-Verschlüsselung | ... 267, 432 |
| chroot | ... 64 | df | ... 104 |
| chroot in einer Live-CD/DVD | | Dienste starten, anhalten und | |
| Umgebung | ... 65 | deaktivieren | ... 105 |
| chroot-jail | ... 64 | diff | ... 84 |
| chroot-Umgebung für SSH | ... 68 | Diffuse | ... 417 |
| Cinnamon: Tastenkombinationen | | ding | ... 85 |
| unter Linux Mint | ... 556 | display | ... 171 |
| ClamAV | ... 374 | Display Power Management | |
| | | Signaling | ... 407 |
| | | DMA-Modus | ... 152 |

| | | | |
|--|--------------|--|--------------|
| dmesg | ... 107 | exiftool | ... 117 |
| DNS | ... 110 | exit | ... 282 |
| Domain Name Registry | ... 399 | Exit Status | ... 493 |
| Domain- und Namensauflösung | ... 110 | exiv2 | ... 115 |
| DOS / Linux - Befehlsunterschiede | ... 108 | export | ... 471 |
| DOS ins UNIX-Format | ... 129 | ext2 | ... 80 |
| dos2unix | ... 108 | EXT2-Partition auf einem USB-Stick erstellen | ... 459 |
| dosbox | ... 109 | ext3 | ... 120 |
| dpkg | ... 112 | Extended Partition Boot Record | ... 549 |
| DPMS | ... 407 | F | |
| Drucker | ... 113 | faillog | ... 120 |
| Druckertreiber | ... 353 | fdisk | ... 120 |
| du | ... 105 | Fedora | ... 271 |
| Dukto R6 | ... 419 | Fensterverwaltung – ein Programm per Mausklick beenden | ... 558 |
| Dvdauthor | ... 418 | Festplatte mounten | ... 124 |
| dvdbackup | ... 417 | ffmpeg2theora | ... 32 |
| DVDStyler | ... 418 | FIFOs | ... 411 |
| E | | file | ... 125, 161 |
| EBR | ... 549 | filtered | ... 237 |
| echo | ... 114 | find | ... 64, 126 |
| Editor | ... 394 | finger | ... 128 |
| EFI | ... 114, 139 | fingerprint | ... 387 |
| EIGamal | ... 381 | Fingerprint | ... 241 |
| Eigentümer | ... 409 | Firewall | ... 237 |
| Ein-/Ausgabe-Umleitungen | ... 518 | FLAC | ... 300 |
| Einbruchserkennung - Claymore | ... 542 | Focuswriter | ... 417 |
| Einführung in die Shell-Programmierung - Artikel Nr. 2 | ... 483 | for...do | ... 513 |
| Einführung in die Shell-Programmierung - Artikel Nr. 3 | ... 504 | free | ... 129 |
| Einführung in die Shellprogrammierung | ... 473 | Freigaben | ... 290 |
| Einleitung: Linux-Kurzreferenz | ... 9 | fromdos | ... 129 |
| EncFS | ... 419 | ftp | ... 128 |
| Entpacker | ... 419 | Funktionen | ... 516 |
| Ersetzer sed | ... 283 | fuser | ... 130 |
| eth0 | ... 275 | G | |
| execute | ... 409 | G4L | ... 545 |
| Exif-Informationen | ... 115 | Gateway | ... 274 |
| | | Gateway-Routers | ... 275 |
| | | gdebi | ... 112 |
| | | GeeXboX | ... 529 |
| | | General Public License | ... 132 |
| | | GEO-Targeting | ... 359 |
| | | Gerätedateien | ... 411 |

| | | | |
|--------------------------------|-------------------------------------|---------------------------------------|----------|
| Ghost for Linux (g4l) ... | 545 | HTML-Injection ... | 469 |
| GID ... | 322 | htop ... | 159 |
| Gigolo ... | 419 | HTTrack ... | 416 |
| gksu ... | 132 | hwclock ... | 79 |
| glxinfo ... | 138 | I | |
| GNU ... | 132 | iconv ... | 160 |
| Gnu Privacy Guard ... | 381 | id ... | 6, 175 |
| Google ... | 421 | id3 ... | 162 |
| gpasswd ... | 132 | ID3-Tags ... | 162 |
| gpg ... | 192, 377, 382, 386f., 389, 391, 393 | id3v2 ... | 163 |
| GPL ... | 132 | identify ... | 170 |
| grabcc ... | 136 | IEEE ... | 236 |
| Grafikkarte ... | 138 | if ... | 164 |
| grep ... | 82, 136, 522 | ifconfig ... | 165, 260 |
| group ... | 409 | ImageMagick ... | 169 |
| groupadd ... | 133 | import ... | 172 |
| groupdel ... | 134 | inetd - der Superdämon ... | 177 |
| groupmod ... | 135 | info ... | 174 |
| groups ... | 63, 135 | init (SysVinit) ... | 175 |
| Grub 2 ... | 138 | init-Programm ... | 176 |
| Grundlagen: Shellskripte ... | 473 | initctl ... | 106, 366 |
| gunzip ... | 135, 345 | Inkrementeller Modus ... | 187 |
| gzip ... | 135, 344 | Inkscape ... | 416 |
| H | | Inodes ... | 314 |
| Hacker ... | 429 | Installation ... | 208 |
| halt ... | 6, 151 | Institute of Electronic Engineers ... | 236 |
| hardinfo ... | 202 | Integrität eines Verzeichnisses ... | 462 |
| Hardware ... | 202 | Integrität von Dateien ... | 463 |
| Hardware Adresse ... | 236 | Integritätsprüfung ... | 213 |
| hdparm ... | 151 | Interrupts ... | 269 |
| head ... | 151, 343 | ip ... | 168, 237 |
| Here-Dokument ... | 210 | iso-8859-1 ... | 189 |
| Herstellerkennung ... | 236 | ISO-8859-1 ... | 161 |
| Highlighting ... | 153 | iso-8859-15 ... | 190 |
| Hintergrundprozess starten ... | 154 | ISO-Dateien mounten ... | 217 |
| history ... | 153, 281 | J | |
| History-Datei ... | 9 | Java ... | 182 |
| host ... | 155, 359 | jobs ... | 155 |
| hostname ... | 151 | john ... | 182 |
| hosts ... | 111 | John the Ripper ... | 185 |
| hosts.allow ... | 155 | join ... | 188 |
| hosts.deny ... | 155 | journalctl ... | 339 |

| | | | | | |
|-----------------------------------|-----|---------------|------------------------------|-----|---------------|
| journal | ... | 339 | Linux-Dateisystem | ... | 193 |
| K | | | Linux-Systemzeit | ... | 79 |
| KDE: Tastenkombinationen | ... | | Linux-Verzeichnishierarchie | ... | |
| 554 | | | 196 | | |
| kdesu | ... | 132 | Literaturverzeichnis | ... | 427 |
| kdiallog | ... | 460 | In | ... | 198 |
| KEINE MELDUNG | ... | 9 | locale | ... | 160, 220, 320 |
| Kernelversion | ... | 189, 269 | locate | ... | 199 |
| Keyserver | ... | 389 | loop | ... | 96 |
| kill | ... | 190 | lpr | ... | 205 |
| kill -9 | ... | 191, 558 | ls | ... | 200 |
| kill -HUP | ... | 190 | ls -IRa --full-time | ... | 430 |
| kill -KILL | ... | 191 | lsdvd | ... | 202 |
| killall | ... | 191 | lshw | ... | 202 |
| KNOPPIX | ... | 531 | lsdf | ... | 203, 431 |
| Kodierung | ... | 125 | lsusb | ... | 202 |
| Kommandos Gruppieren | ... | | Lynx | ... | 206 |
| 492 | | | M | | |
| Kommandosubstitution | ... | 507 | MAC | ... | 236 |
| Kommandozeilenfenster | ... | | MAC-ACL | ... | 167 |
| 277 | | | MAC-Adressen | ... | 167 |
| Konvertierung | ... | 189, 207, 397 | mail | ... | 209 |
| Konvertierung von Video-, Audio- | | | mailutils | ... | 209 |
| oder Bildformate | ... | 23 | make | ... | 208 |
| kopieren | ... | 73 | make install | ... | 208 |
| Kryptographie mit GnuPG - Artikel | | | makepasswd | ... | 208 |
| Nr. 1 | ... | 377 | man | ... | 211 |
| Kurzreferenz der gängigsten gpg- | | | Master Boot Record | ... | 549 |
| Befehle - Artikel Nr. 3 | ... | 386 | MBR | ... | 549 |
| Kwrite: Text in Groß- oder | | | mc | ... | 212 |
| Kleinbuchstaben umwandeln | ... | | MD5 | ... | 185 |
| 553 | | | MD5-Verschlüsselung | ... | 267, 432 |
| L | | | md5deep | ... | 213 |
| lame | ... | 298 | md5sum | ... | 213, 318 |
| LAME-Paket | ... | 298 | Medium Access Control | ... | |
| last | ... | 197 | 236 | | |
| latin1 | ... | 160 | Metadaten von Digitalbildern | ... | |
| ldd | ... | 156, 197 | 115 | | |
| Leseberechtigung | ... | 409 | Midnight Commander | ... | 212 |
| less | ... | 197 | mkdir | ... | 215 |
| LessLinux Search and Rescue | ... | | mke2fs | ... | 123 |
| 537 | | | mkfs.ext2 | ... | 459 |
| Link | ... | 411 | mkfs.ext4 | ... | 196 |
| Links | ... | 423 | mkpasswd | ... | 208 |
| Linux | ... | 193 | Monatskalender | ... | 58 |

| | | | |
|--------------------------------|-------------------|------------------------------------|--------------|
| Monitor | ... 406 | offenen Ports | ... 237 |
| Moovida-Media-Center | ... 418 | Ogg Vorbis | ... 299 |
| more | ... 215 | ogg123 | ... 299 |
| mount | ... 216 | oggdec | ... 299 |
| MP3 | ... 298 | oggenc | ... 299 |
| MP3-Dateien | ... 461 | Okular | ... 416 |
| mp3gain | ... 218 | OpenDocument | ... 247 |
| mp3splt | ... 220 | OpenShot | ... 418 |
| mp3wrap | ... 221 | OpenSSH | ... 302 |
| MPlayer | ... 222 | OpenSuse | ... 107, 271 |
| mtr | ... 228 | Operating System | ... 241 |
| Multimedia-Live-CD | ... 529 | Opus | ... 301 |
| Multimedia-Player | ... 416 | opusdec | ... 302 |
| mv | ... 229 | opusenc | ... 301 |
| My Traceroute | ... 228 | opusinfo | ... 302 |
| N | | other | ... 409 |
| Name Based Hosting und SSH | ... 309 | owner | ... 409 |
| Namen eines laufenden | | P | |
| Programms mit xprop ermittelt | ... 558 | p7zip | ... 248 |
| Nameserver | ... 110 | Parameter | ... 508 |
| ncal | ... 58 | Parametererweiterung | ... 510 |
| netstat | ... 231 | PartedMagic | ... 375 |
| Netzwerk manuell aufsetzen | ... 235 | Partimage | ... 552 |
| Netzwerkanalyse-Programm | ... 345 | Partitionen | ... 104 |
| Netzwerkgeräte | ... 231 | Partitionsmanager | ... 120 |
| Netzwerkkarte | ... 236 | passwd | ... 183, 264 |
| Netzwerkverbindungen | ... 231 | Passwortüberprüfung – Artikel Nr.1 | ... 182 |
| NIC | ... 399 | Passwortüberprüfung – Artikel Nr.2 | ... 184 |
| nice | ... 245 | paste | ... 270 |
| nicht ausdruckbaren Zeichen | ... 58 | Pattern | ... 10 |
| nl | ... 237 | PDF | ... 250 |
| nmap | ... 237 | PDF Chain | ... 250 |
| Normale Dateien | ... 411 | PDF-Arranger | ... 250 |
| nostromo | ... 267, 370, 435 | PDF-Betrachter | ... 416 |
| Notizen | ... 560 | pdfarranger | ... 249 |
| NTFS | ... 539 | pdffonts | ... 256 |
| NUM-Lock beim Start aktivieren | ... 246 | pdfimages | ... 258 |
| O | | pdfinfo | ... 255, 456 |
| odt2txt | ... 247 | pdftk | ... 249 |
| | | pdftohtml | ... 257 |
| | | pdftoppm | ... 258 |
| | | pdftotext | ... 256 |
| | | Peer-to-Peer-Dateiaustausch | ... |

| | | | |
|------------------------------------|-------------|-----------------------------------|-------------------|
| 419 | | R | |
| Permissions | ... 409 | RAR-Dateien | ... 372 |
| Pfadvariable | ... 471 | read | ... 409, 481 |
| pgrep | ... 261 | reboot | ... 277 |
| photofilmstrip - Slideshow creator | | Rechneruhr | ... 79 |
| mit den »Ken Burns Effekt« | ... | recode | ... 221, 274, 321 |
| 418 | | Regelmäßige Prüfung | ... 188 |
| PID | ... 264 | Regeln für Dateinamen | ... 10 |
| ping | ... 259 | Registrierungsstellen | ... 399 |
| ping6 | ... 259 | reguläre Ausdrücke | ... 519 |
| Pipe | ... 361 | remove | ... 276 |
| Pipes | ... 518 | rename | ... 274 |
| pkill | ... 262 | Rescatux | ... 150 |
| Platzverbrauch | ... 105 | reset | ... 277 |
| play | ... 300 | Rest der Welt | ... 409 |
| Playonlinux | ... 417 | rm | ... 276 |
| Portfwd | ... 416 | rmdir | ... 276f. |
| Ports | ... 268 | route | ... 231, 274 |
| Portscanner | ... 237 | Router | ... 274 |
| printenv | ... 471 | RPM-Pakete | ... 271 |
| Priorität | ... 245 | RSA | ... 381 |
| proc-Dateisystem | ... 269 | rsync | ... 277 |
| process status | ... 263 | Runlevels | ... 175, 177 |
| procinfo | ... 269 | S | |
| Programm-Bibliotheken | ... 197 | S-Bit | ... 429f. |
| Programm-Fenster verschieben | | SAM | ... 531 |
| und skalieren | ... 553 | Samba | ... 290 |
| Programme finden, installieren und | | Schnellanleitung Ver- und | |
| entfernen | ... 270 | Entschlüsselung – Artikel Nr. 5 | ... |
| Prozess-ID | ... 264 | 392 | |
| Prozesse | ... 263 | Schreibberechtigung | ... 409 |
| Prozessor-Flags | ... 270 | scp | ... 305 |
| ps | ... 263 | Screenshot | ... 172 |
| pstree | ... 263 | Security Access Management | ... |
| Public-Key-Verfahren | ... 380 | 531 | |
| pv | ... 91, 100 | sed | ... 283, 524 |
| pwd | ... 262 | service | ... 105, 178 |
| pwgen | ... 262 | sfill | ... 291 |
| Q | | SFTP | ... 68 |
| QR-Code | ... 272 | sftp - sicherere Dateiübertragung | |
| qrencode | ... 272 | ... 306 | |
| qtparted | ... 120 | SGID-Bit | ... 413 |
| qtqr | ... 273 | sha1sum | ... 318 |
| Quota's überprüfen | ... 459 | shadow | ... 183 |
| Quoting | ... 491 | Shell | ... 281 |

| | |
|---|--|
| Shellskripts ... 473 | Subshells ... 487, 489 |
| Shellskripts - Einfache | substitute user ... 322 |
| Batchabläufe erstellen - Artikel Nr. 1 ... 477 | Suchen nach Dateien bzw. Dateiinhalte ... 463 |
| shred ... 292 | Suchmaschinen ... 421 |
| shutdown ... 293 | sudo ... 323 |
| Shutdown ... 277 | sudo - /etc/sudoers bearbeiten ... 326 |
| siege ... 293 | sudo – Privilegien gewähren ... 329 |
| Sitecopy ... 294 | SUID-Bit ... 412 |
| skill ... 296 | Super-GRUB2-Disk ... 149 |
| Skript-Listings ... 429 | SWAP - Auslagerungsspeicher ... 331 |
| Skript: cronjobs ... 429 | symbolische Links ... 411 |
| Skript: Crypt ... 432 | Symmetrische Verschlüsselung mit gpg ... 391 |
| Skript: Metadaten in PDF-Dateien löschen ... 454 | synchronisieren ... 277 |
| Skript: Start-/Stop-Skript für den Entwicklungsserver XAMPP ... 457 | systemctl ... 107, 339 |
| Sniffer ... 345 | systemd – Das Init-System ... 336 |
| Sockets ... 231, 411 | Systemmonitor ... 159, 349 |
| Sonderzeichen ... 342 | T |
| sort ... 312, 362 | Tabellen ... 468 |
| Sound ... 297 | tac ... 343 |
| SoundConverter ... 417 | tail ... 343 |
| Spezialparameter ... 509 | tar ... 343, 438 |
| Spiegeln von Webseiten ... 398 | Tastatur ... 342 |
| split ... 297 | Tastenkombinationen ... 342 |
| SSH - secure shell ... 302 | TCP ... 231 |
| ssh -XC ... 306 | TCP-Wrapper ... 156 |
| SSH-Authentifizierungsmechanismen ... 307 | tcpd ... 156 |
| ssh-keygen ... 437 | tcpdump ... 345 |
| sshd ... 303 | tee ... 348 |
| Start-/Stop-Skript erstellen ... 20 | Terminalkommandos aus einer Textdatei in ein Terminalfenster einfügen (grafische Oberfläche) ... 282 |
| stat ... 314 | test ... 164, 348 |
| STAT ... 264 | Tests, Verzweigungen und Schleifen ... 512 |
| steghide ... 315 | textbasierte Shell ... 281 |
| Steuerzeichen ... 58 | textbasierter Browser ... 397 |
| Sticky-Bit ... 414 | Textmanipulationen ... 519 |
| streamripper ... 220, 320 | Textmeldung auf den Bildschirm |
| strings ... 321 | |
| su ... 322 | |

| | |
|---|--|
| ... 460 | 332 |
| TIME ... 264 | UNIX ins DOS-Format ... 349 |
| Tintenstrahldrucker ... 353 | unix2dos ... 365 |
| Tipps und Tricks ... 553 | unrar ... 372 |
| todos ... 349 | unzip ... 372 |
| Tonspur mit Mplayer extrahieren ... 226 | update-rc.d ... 106, 178 |
| top ... 246, 349 | updatedb ... 200 |
| tote Taste ... 342 | Upstart ... 365 |
| touch ... 349 | USB-Geräte ... 202 |
| tr ... 351 | useradd ... 368, 437 |
| traceroute ... 353 | userdel ... 370 |
| tree ... 350 | usermod ... 371 |
| tripwire ... 215 | UTF-8 ... 161, 190, 372 |
| TTY ... 264 | uudecode ... 373 |
| tune2fs ... 120 | uuencode ... 372 |
| TurboPrint ... 353 | UUID ... 57, 332 |
| TWOFISH ... 392 | V |
| U | Variablen ... 471 |
| Ubuntu ... 106 | Variablen und Quoting ... 505 |
| UDP ... 231 | Verbindungsdaten ... 231 |
| UEFI ... 114 | Verbindungstypen ... 231 |
| UFW ... 355 | Verschlüsseln und Entschlüsseln ... 387 |
| UFW BLOCK ... 359 | Verschlüsselung ... 377 |
| UID ... 322 | Version des Kernels ... 374 |
| umask ... 363 | Verzeichnis entfernen ... 276 |
| Umleiten in eine Datei ... 360 | Verzeichnisse ... 411 |
| Umleitung an ein Programm ... 361 | vi ... 394 |
| Umleitung auf Geräte ... 360 | Videobearbeitung ... 53 |
| Umleitung von Befehlen ... 360 | VideoLAN ... 416 |
| Umleitung von TCP/IP-Verbindungen ... 308 | videotrans - DVD-Erstellungswerkzeuge ... 419 |
| umount ... 218, 364 | Virens Scanner ... 374 |
| uname -a ... 374 | Virens Scanner ClamAV von CD-ROM starten ... 375 |
| Uncomplicated Firewall ... 355 | Virtuelle Netzwerkkarten einrichten ... 166 |
| Unetbootin ... 420 | visudo ... 327 |
| Unicode ... 342 | vlc ... 416 |
| Unified Extensible Firmware Interface ... 114 | vobcopy ... 395 |
| uniq ... 365 | Vorwort ... 5 |
| Unity: Tastenkombinationen unter Ubuntu ... 554 | W |
| Universally Unique Identifiers ... | w ... 397 |
| | W3M ... 397 |
| | WAV ... 299 |

| | | | |
|------------------------------------|----------|-----------------------------|----------|
| Wave-Dateien ... | 462 | Zugriffsrechte ... | 363, 409 |
| wc ... | 201, 397 | Zugriffsrechte ändern ... | 465 |
| Webcam mit MPlayer ... | 227 | zypper ... | 271 |
| wget ... | 398 | . | |
| whatis ... | 399 | .bash_rc ... | 315 |
| when ... | 401 | [| |
| whereis ... | 398 | [Alt] + [.] ... | 9 |
| which ... | 399 | [Alt] + [F7] ... | 281 |
| while und until ... | 514 | [Strg] + [C] ... | 9, 281f. |
| whoami ... | 400 | [Strg] + [d] ... | 282 |
| whois ... | 399 | [Strg] + [L] ... | 281 |
| WHOIS-Server ... | 400 | [Strg] + [q] ... | 282 |
| Wildcards ... | 10 | [Strg] + [s] ... | 282 |
| Wildcards und Pattern Matching ... | 493 | [Strg] + [Shift] + [V] ... | 282 |
| Wine ... | 402 | [Strg] + [u] ... | 281 |
| WinNT LM ... | 185 | / | |
| WLAN ... | 167 | /etc/default/useradd ... | 368 |
| Wortlistenmodus ... | 187 | /etc/hostname ... | 236 |
| write ... | 409 | /etc/hosts ... | 236 |
| X | | /etc/network/interfaces ... | 235 |
| XAMPP ... | 404, 457 | /etc/pam.conf ... | 368 |
| xkill ... | 406, 558 | /etc/passwd ... | 264, 368 |
| xmessage ... | 460 | /etc/services ... | 268 |
| Xnviewmp ... | 417 | /etc/shadow ... | 264, 368 |
| Xpdf ... | 416 | /etc/skel ... | 368 |
| xprop ... | 558 | /etc/sudoers ... | 327 |
| xrandr ... | 406 | /etc/sysconfig/language ... | ... |
| xset ... | 407 | 160 | |
| xwininfo ... | 47, 173 | /sbin/route ... | 275 |
| Y | | /var/log/kern.log ... | 236 |
| yapet ... | 408 | /var/log/messages ... | 236 |
| yum ... | 271 | # | |
| Yumi ... | 420 | #!/bin/bash ... | 473 |
| Z | | \$ | |
| Zeilennummerierung ... | 58 | \$HISTFILE ... | 471 |
| Zeilennummern ... | 237 | \$HISTSIZe ... | 471 |
| Zeilenumbrüche ... | 58 | \$HOME ... | 471 |
| Zeitdrift ... | 79 | \$MAIL ... | 471 |
| Zeitstempel ... | 349 | \$MAILCHECK ... | 471 |
| zenity ... | 460 | \$PATH ... | 471 |
| Zip-Dateien ... | 372 | \$RANDOM ... | 471 |
| zless ... | 198 | \$SHELL ... | 471 |
| | | \$UID ... | 471 |

Diese Kurzreferenz möchte ein kleiner Helfer für den alltäglichen Umgang mit Linux sein. Sie ist keine allumfassende vollständige Referenz.

Die Kurzreferenz bezieht sich im wesentlichen auf die Linux-Distributionen Linux Mint und Ubuntu. Bei den anderen Linux-Distributionen wie Xubuntu, Debian, Fedora, CentOS, Lubuntu, Mageia, OpenSuse, Symphony, Ark Linux, Manjaro, PCLinuxOS, Siduction, Bhodi Linux etc. sollte vieles sehr ähnlich funktionieren.

Die in dieser Linux-Kurzreferenz aufgeführten Befehle und Funktionen sind nicht vollständig in dieser Kurzreferenz dokumentiert, es sind nur die am häufigsten benutzten oder gesuchten Optionen zu den einzelnen Befehlen aufgelistet.

Wer hier einige Optionen oder einzelne Befehle vermisst, kommt nicht umhin die Kurzhilfe – über

<Befehlsname> --help die Infodateien – über
info <Befehlsname> - oder die Manualseiten – über
man <Befehlsname> zu konsultieren.

Bei der intensiven Beschäftigung mit Linux sollte zu Beginn folgendes unbedingt gelesen werden: das Vorwort, die Einleitung und der Abschnitt der sich mit den Zugriffsrechten (siehe Buchstabe »Z«) beschäftigt.

Verwenden Sie eine aktuellere Linux-Distribution, so werden einige Hinweise mitunter nicht zutreffend sein. Sie können dann die Kurzreferenz Ihren Bedürfnissen anpassen. Nach Möglichkeit sollten Sie Ihre Version der Linux-Kurzreferenz der Open-Source-Gemeinde wieder zur Verfügung stellen.

Viel Spaß !!!