Dinge die das Leben erleichtern

Alle Mails an Root an echte Mailadresse weiterleiten

Viele Programm senden default Servicemails an Root (hier Ubuntu 16.04). Nur hat Root ja keine echte Adresse (root@localhost). Also bedienen wir hier uns Aliases. Zum einrichten der Funktion geht man wie folgt vor:

```
nano /etc/aliases
```

```
# See man 5 aliases for format
postmaster:    root
backuppc: root
root: technik@meinedomain.at
```

oben sieht man user die alle an „root“ senden. Und am Schluss eben wer Root eigentlich ist. Die neuen Aliases liest man mit **„newaliases“** ein. Jetzt bestimmt man noch den Absendernamen. Das kann eine Adresse oder auch ein echter Personenname sein.

```
chfn -f "testrecher@hallo.meinedomain.at Root" Root
```

Nun muss man noch die Absenderdomain in der Postfixconfig angeben.

```
nano /etc/postfix/main.cf
```

```
myhostname = srv-backup.hallo.meinedomain.at
```

Jetzt noch den Postfix reloaden und schon gehts.

Dateiverwaltung

Befehl	Beschreibung
`for i in * ;do mv "$i" $(echo "$i" | tr [:blank:] _ ) ;done`	Bei allen Dateien im Verzeichnis das Leerzeichen entfernen und durch einen Unterstrich ersetzen
find / -nouser -or -nogroup	Sucht nach Dateien und Ordner im ganzen root die keine UID oder GID haben

Netzwerk und Hardware

Befehl	Beschreibung
rsync -e „ssh -p222“ -Pvz /backup/vzdump1.tar –bwlimit=100 nas1:/mnt/backup/12-10-2009/	Rsync auf einem bestimmten Port benutzen, mit Bandbreitenbeschränkung
ifconfig br0:1 10.55.1.100 netmask 255.255.255.0 up	Einer Netzwerkkarte eine zweite IP-Adresse zuweisen
route add default gw 192.168.178.1	Default-Gateway bestimmen
`nmap -p 22 172.16.10.* --open | grep Interesting`	nmap nur offene ports anzeigen
tar -czf /p-openvz-templates/ubuntu-8.0-standard_8.10_amd64.tar.gz .	Ein TAR.GZ erstellen
ethtool -s eth0 speed 1000 duplex full autoneq on	speed der Netzwerkkarte ändern
route add -net 11.22.0.0 netmask 255.255.0.0 dev ppp0	Route für z.B. eine VPN legen
ssh -L 8291:10.8.70.10:8291 user@172.30.1.10	SSH-Tunnel für z.B. Mikrotik auf localhost mappen
`less /proc/cpuinfo |grep -i "physical id" |uniq -c`	physikalisch vorandenen CPUs eruieren
`nwipe /dev/XXX`	Vernichtet alle Daten am angegebenen Datenträger

Ärger mit Prozessen und Diensten

Befehl	Beschreibung
jobs	Prozesse anzeigen die im Hintregrund sind
STRG + Z	Prozess anhalten und in den Hintergrund schieben
fg <jobnummer>	Bringt den Befehl mit der angegebenen Jobnummer in den Vordergrund
bg <jobnummer>	Bringt den Befehl mit der angegebenen Jobnummer in den Hintergrund
<Befehl> &	Startet den Befehl in den Hintergrund
nohup <Befehl>	Lässt den Prozess bei SSH-Abmeldung weiterlaufen
fuser -uv /mnt/sda2	Hier sieht man warum sich ein Device (z.B. CDrom) nicht aushängen lässt
fuser -k {device/Datei}	killen der Sperre
sudo ifconfig enp5s0:1 192.168.123.11/24 up	weitere IPadresse hinzufügen

Programme spezifisches

Befehl	Beschreibung
`echo 'Acquire::http { Proxy "http://10.69.99.10:3142"; };' | sudo tee /etc/apt/apt.conf.d/01proxy`	Setzt den Server „apt-cacher“ als Proxy für das Paketmanagement

Befehl	Beschreibung
`echo 'Acquire::http { Proxy "http://apt-cacher.osit.cc:3142"; };' | sudo tee /etc/apt/apt.conf.d/01proxy`	Setzt den Server „apt-cacher“ als Proxy für das Paketmanagement
dpkg –get-selections '*' > Paketliste.txt	Erstellen eines APT-Abbildes, zur Übertragung der gleichen Pakete auf einem anderen Rechner
dpkg –set-selections < Paketliste.txt INFO Synaptic	Setzen der Liste auf dem Zielrechner
apt-get dselect-upgrade INFO Synaptic	Die zuvor gesetzte Liste installieren
grep -R „blabla“ *	Von dort wo man sich befindet rekrusiv alle Dateien nach „blabla“ durchsuchen
aptitude search '~i linux-image'	Sucht nach allen installierten Linuxkernels auf Debian basierenden Systemen
sh -c '/usr/bin/nvidia-settings –load-config-only'	Nvidiaconfig beim Desktopstart laden
ldapsearch -D „cn=directory manager“ -w geheim -h localhost -b „dc=osit,dc=cc“	unter CentOS (kolab) ldapsearch durchführen
ldapsearch -h ldapserver.local -Z -x -D „cn=Manager,dc=osit,dc=cc“ -W	unter Gentoo mit TLS

IPV6 DNSserver

Servername	Adresse
Google	2001:4860:4860::8888 / 2001:4860:4860::8844
ns1.easyname.eu	2a02:1b8:ea59::2
Tunnelbroker	2001:470:20::2

Clonezilla

Zuerst das Image downloaden und auf einem FAT32 USBstick kopieren.

```
unzip clonezilla-live-1.0.10-8.zip -d /media/usb/
```

Danach muss man noch den Booloader installieren.

```
cd /media/usb/utils/linux
bash makeboot.sh /dev/sdd1
```

Hier alle Fragen mit „Ja“ beantworten. Das wars. Nun kann man sein Clonezilla vom USBstick booten.

- Clonezilla über PXE ausliefern

Mounten von Verzeichnissen und Laufwerken

Mounten mit den Rechten des Users auf beiden Seiten

```
sshfs#ml@app:/home/ml
/opt/openthinclient/server/default/data/nfs/home/ml/MYHOME fuse _netdev 0 0
```

Mounten mit 777 auf der Linuxseite, ideal für Zugriff meherer User, Gegenseite SAMBA

```
sshfs#root@data:/media/daten /mnt/data fuse
gid=100,umask=0,allow_other,_netdev 0 0
```

ACLs setzen

Hat ein Ordner einen Vorgabewert für z.B. die Gruppe „edv“ rwx, heist das nicht das Dateien und Ordner die mit dieser Gruppe angelegt wurden, oder dieser Gruppe gehören, dann auch dafür rwx haben, sondern das Mitlieder der Gruppe „edv“ in diesen Ordner sich befindliche Dateien und Unterordner verändern und auch Dateien und Ordner anlegen dürfen. Dies wäre die einfachste Möglichkeit ACLs zu geniessen.

A C H T U N G
Mit der Option **-d** werden immer die Default ACL gesetzt.

```
setfacl -d -m group:edv:rwx /var/iso/
```

Rechte rwx für die ganze Welt setzten. **-R wäre rekursiv**

```
setfacl -d -m other:rwx /var/iso/
```

Bei den Dateisystemen jfs und xfs können ACLs standardmäßig gesetzt werden. Bei den unter Linux üblichen Dateisystemen ext3, ext4 und reiserfs müssen ACLs aber explizit aktiviert werden. Dies geschieht durch die Option -o acl beim Einbinden der Partition oder direkt in /etc/fstab.

```
/dev/sda5  /home  ext3  defaults,nodev,acl 0 2
```

Entfernen der gesamten ACL, so dass nur die klassischen Unixrechte zurückbleiben:

```
setfacl -b DATEI ...
```

Mit dem folgenden Befehl wird die Default-ACL entfernt:

```
setfacl -k DATEI ...
```

Praxibeispiel

Anton möchte verhindern, dass sein Chef, der ebenfalls in der Gruppe schreiber ist, die Datei lesen kann. Gleichzeitig möchte er den Lektoren die Möglichkeit geben, seine Datei zu korrigieren. Jetzt

werden dem Nutzer chef alle Rechte genommen und der Gruppe lektoren die Schreib- und Leserechte eingeräumt.

```
setfacl -m u:chef:-,g:lektoren:rw roman.txt
```

Die Ausgabe von getfacl sieht jetzt so aus:

```
# file: roman.txt
# owner: anton
# group: schreiber
user::rw-
user:chef:---
group::r--
group:lektoren:rw
mask::rw-
other::r--
```

Wie man sieht, werden die ACLs für den Chef und die Lektoren jetzt angezeigt. Die Ausgabe von ls -l sieht jetzt so aus:

```
-rw-rw-r--+ 1 anton schreiber 825 2010-01-01 00:00 roman.txt
```

Das „+“ zeigt an, dass ACLs vorhanden sind. Welche dies sind, sieht man aber über ls nicht.

Möchte man einem User einer Ordnerhirachie zusätzlich Leserechte einräumen macht das z.B. so:

```
setfacl -R -m u:hansi:rx /pfad ## -d würde dies Vorgabe für weitere Dateien
machen
```

Prioritäten

Welcher Eintrag für die Zugriffsrechte entscheidend ist, bestimmen folgende Regeln:

- Die ACL wird von oben nach unten abgearbeitet.
- Die erste zutreffende Regel gilt.

Anton ist der Besitzer der Datei. Für ihn gelten die Rechte des Besitzers user::rw-. Der Eintrag user:anton:r– folgt später und wird daher ignoriert. Der Chef sei in der besitzenden Gruppe schreiber, welche lesen darf (group::r–). Trotzdem hat der Chef überhaupt keinen Zugriff, weil er weiter oben als benannter Benutzer ohne Rechte (user:chef:—) eingetragen ist.

- http://wiki.ubuntuusers.de/ACL

```
find /verzeichnis/ -type d -exec chmod 755 {} +
find /verzeichnis/ -type f -exec chmod 644 {} +
```

APT Paketverwaltung

Paket sperren: Beispiel Linuxkernel nicht mehr updaten.

```
echo linux-image-generic hold | dpkg --set-selections
echo linux-generic hold | dpkg --set-selections
```

Paket entsperren:

```
echo <paketname> install | dpkg --set-selections
```

Nach dem Paketnamen suchen wo eine bestimmte Datei enthalten ist.

```
dpkg -S kf5-config
libkf5kdelibs4support5-bin: /usr/share/man/uk/man1/kf5-config.1.gz
libkf5kdelibs4support5-bin: /usr/share/man/de/man1/kf5-config.1.gz
libkf5kdelibs4support5-bin: /usr/share/man/nl/man1/kf5-config.1.gz
libkf5kdelibs4support5-bin: /usr/share/man/sv/man1/kf5-config.1.gz
libkf5kdelibs4support5-bin: /usr/share/man/man1/kf5-config.1.gz
libkf5kdelibs4support5-bin: /usr/share/man/pt_BR/man1/kf5-config.1.gz
libkf5kdelibs4support5-bin: /usr/bin/kf5-config
libkf5kdelibs4support5-bin: /usr/share/man/ca/man1/kf5-config.1.gz
```

Das Paket heist also „libkf5kdelibs4support5-bin“.

Schlüssel von einem Schlüsselserver importieren:

```
apt-key adv --recv-keys --keyserver keyserver.ubuntu.com 0C5A2783
```

Eigenes Debianrepository signieren:

```
gpg --import geheimer-Schlüssel.asc
cd Repository
apt-ftparchive packages . > Packages
apt-ftparchive release . > Release
gpg --output Release.gpg -ba Release
```

Datenbanken und deren Befehle

PostgreSQL

Um sich mit einer Datenbank auf einem entfernten Server zu verbinden und die vorhandenen Datenbanken anzuzeigen, gibt man folgendes ein:

```
psql -h entfernter_Server -U Benutzer -W -l
```

MYSQL

Erstellen einer MYSQL-Datenbank inkl. Benutzer

Zuerst muss man sich per root (oder einem User mit Rootrechten für MYSQL) auf dem Server verbinden. Läuft der MYSQL-Server auf dem gleichen host ist der Befehl relativ simpel:

```
mysql -u root -p
```

Jetzt kann man sich eine neue Datenbank anlegen und die entsprechenden Rechte vergeben.

```
CREATE DATABASE movies;
GRANT ALL ON movies.* TO movies_user@'localhost' IDENTIFIED BY
'geheimes_Passwort';
GRANT ALL ON movies.* TO movies_user@'supertux.bla.com' IDENTIFIED BY
'geheimes_Passwort';
```

Die Befehle erstellen eine Datenbank mit dem Benutzer „movie_user“, wobei die dieser sich von „localhost“ und „supertux.bla.com“ verbinden darf. Um sich alle vorhandenen Datenbanken anzeigen zu lassen kann man in der mysqlCLI folgendes Kommando eingeben:

```
mysql> show databases;
```

Backup und Recovery von einer MYSQL-Datenbank

Mit dem Befehl

```
mysqldump --opt -u root -p --all-databases > sicherung.sql

oder bei Problemen

mysqldump --single-transaction -u username -p db > db.sql
```

sichert mit den Rechten des (SQL-Benutzers) „root“ alle Datenbanken in die Datei sicherung.sql. Die Sicherung kann natürlich auch mit einem anderen Benutzer durchgeführt werden, sofern dieser die notwendigen Rechte in den zu sichernden Datenbanken hat. Sehr wichtig ist die angegebene Option –opt, da diese alle notwendigen Sperren für die Dauer der Sicherung setzt.

Anstatt allen Datenbanken kann man auch einzelen Datenbanken sichern. Möchte man z.B. nur die Datenbank „movies“ sichern, so lautet der Befehl

```
mysqldump --opt -u root -p movies > moviesDB-backup.sql
```

Zurückspielen einer MYSQL-Datenbank:

```
mysql -u root -p movies < moviesDB-backup.sql
mysqladmin -u root -p flush-privileges
```

Wichtig hierbei ist das die Datenbank die man zurückspielen möchte bereits im System existiert.

Mit entfernter Datenbank verbinden

```
mysql -u amarok -D amarok -h <HOSTNAME> -p
```

Bash History unendlich mit sofortigen schreiben

First, you must comment out or remove this section of your .bashrc (default for Ubuntu). If you don't, then certain environments (like running screen sessions) will still truncate your history:

```
# for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
# HISTSIZE=1000
# HISTFILESIZE=2000
```

Second, add this to the bottom of your .bashrc:

```
# Eternal bash history.
# ---------------------
# Undocumented feature which sets the size to "unlimited".
# http://stackoverflow.com/questions/9457233/unlimited-bash-history
export HISTFILESIZE=
export HISTSIZE=
export HISTTIMEFORMAT="[%F %T] "
# Change the file location because certain bash sessions truncate
.bash_history file upon close.
#
http://superuser.com/questions/575479/bash-history-truncated-to-500-lines-on
-each-login
export HISTFILE=~/.bash_eternal_history
# Force prompt to write history after every command.
# http://superuser.com/questions/20900/bash-history-loss
PROMPT_COMMAND="history -a; $PROMPT_COMMAND"
```

Note: every command is written immediately after it's run, so if you accidentally paste a password you cannot just „kill -9 %%“ to avoid the history write, you'll need to remove it manually.

Also note that each bash session will load the full history file in memory, but even if your history file grows to 10MB (which will take a long, long time) you won't notice much of an effect on your bash startup time.

Grub2 Defaulteintrag

Das ganze geht mit

```
grub-set-default
```

oder direkt in die Datei

```
/etc/default/grub
```

```
GRUB_DEFAULT=XX
```

Danach natürlich ein „update-grub2“ nicht vergessen.

From:
https://deepdoc.at/dokuwiki/ - **DEEPDOC.AT - enjoy your brain**

Permanent link:
https://deepdoc.at/dokuwiki/doku.php?id=dinge_die_einem_das_leben_erleichtern&rev=1536567249

Last update: **2018/09/10 10:14**