# Tipps und Tricks

In diesem Bereich des Wikis werden sporadisch Kurzanleitungen, Workarounds usw. zu verschiedenen Themenbereichen auftauchen.

## Rettungsumgebung (Chroot)

Wie ich selbst leidvoll erkennen musste, taugt die openSUSE Rettungsumgebung nicht dazu eine vollständige „Chroot“ Umgebung aufzubauen um an einem nicht mehr startenden System zu arbeiten. Daher hier eine kurze Anleitung zur Realisation einer Chroot-Umgebung für openSUSE.

Schritt 1 ist statt des Rettungssystems ein openSUSE-Live-System zu starten.

Danach können Schritt für Schritt alle Teilverzeichnissysteme zu einer vollständigen Umgebungzusammen gesetzt werden. Im Folgenden wird dies basierend auf unserer Partitionierungsempfehlung mit Software-RAID und LVM vorgenommen:

```
linux:~ # mount /dev/system/root /mnt
linux:~ # mount /dev/system/var /mnt/var
linux:~ # mount /dev/mdXXX /mnt/boot
```

**mdXXX** steht für das Software-RAID Device auf dem das /boot-Verzeichnis liegt. Leider nummeriert die Live-Umgebung SW-RAID Devices anders durch, als das installierte System. Kann also sein, das aus /dev/md0 im realen System /dev/md127 in der Live-Umgebung wird. Einfach ausprobieren.

Dann müssen noch die speziellen Verzeichnisse hinzugefügt werden:

```
linux:~ # mount -t proc none /mnt/proc
linux:~ # mount -t sysfs non /mnt/sys
linux:~ # mount -o bind /dev /mnt/dev
```

(Genau der Teil funktioniert mit dem einfachen Rettungssystem nicht!)

Damit ist die Chroot-Umgebung fertig und kann betreten werden:

```
linux:~ # chroot /mnt
```

Jetzt kann im installierten System gearbeitet werden. Sie können hier beispielsweise Grub reparieren oder eine neue inird erzeugen.

Verlassen wird es einfach mit ***exit***

## Zarafa License-Daemon im Eigenbau

Offiziell wird openSUSE als Basis einer Zarafa-Version nicht unterstützt, was unter anderem dazu führt, dass weder Zarafa-Backup noch der License-Daemon dafür zur Verfügung stehen. Beide Tools

enthalten Closed-Source-Komponenten und sind damit nicht Bestandteil der Open-Source-Pakete die wir in unserem Build-Service-Repository vorhalten.

Sie sind aber Bestandteil der Free-Edition und können somit kostenlos genutzt werden, allerdings nur auf unterstützten Plattformen, wie etwa Ubuntu LTS.

Problematisch ist grundsätzlich, dass Zarafa seine eigene Software meist statisch gegen vorhandene System-Libraries linkt. Da sich diese von Distribution zu Distribution in Ihren Versionen unterscheiden, ist es nicht möglich SLES oder RedHat-Pakete einfach unter openSUSE zu nutzen.

Möglich ist aber den License-Daemon in einer Art Sandbox laufen zu lassen, in der er alle Libraries hat, die er benötigt. Die folgende Beschreibung erläutert den Aufbau der Sandbox auf einem Ubuntu-LTS System

## VM einrichten

Zunächst muss dafür ein Ubuntu-LTS System installiert werden, hier empfiehlt sich die Arbeit mit Virtualbox. Entgegen der Angaben im Zarafa-Wiki spielt die Architektur der Sandbox sehr wohl eine Rolle. Da wir unsere invis-Server immer als 64Bit Systeme installieren, nutzen wir entsprechend auch ein 64Bit Ubuntu als Basis.

Zusätzlich zur einfachen Standard-Installation wird das Software-Paket **mklibs** benötigt:

```
heinz@ubuntu:~$ sudo apt-get install mklibs
```

Wenn Sie Ihre Sandbox unter SLES bauen möchten, werden Sie feststellen, dass es für SUSE-Linuxe das Paket **mklibs** nicht gibt. Abhilfe schaft ein gleichnamiges Shellscript: https://github.com/blunderer/mklibs

## Zarafa-Pakete herunterladen und entpacken

Benötigt wird die zur installierten VM passende Zarafa Free-Edition, zu finden auf dem Zarafa-Download-Server: http://download.zarafa.com/community

```
heinz@ubuntu:~$ wget
http://download.zarafa.com/community/final/7.0/7.0.8-35178/zcp-7.0.8-35178-u
buntu-10.04-x86_64-free.tar.gz
....
heinz@ubuntu:~$ tar -xzvf zcp-7.0.8-35178-ubuntu-10.04-x86_64-free.tar.gz
```

## Zarafa-Pakete installieren

In diesem Schritt müssen vorbereitend einige Zarafa-Pakete installiert werden. Benötigt werden:

- zarafa-licensed
- zarafa-common
- zarafa-client
- zarafa-server

```
heinz@ubuntu:~$ cd zcp-7.0.8-35178-ubuntu-10.04-x86_64/
heinz@ubuntu:~/zcp-7.0.8-35178-ubuntu-10.04-x86_64$ sudo dpkg -i zarafa-licensed_7.0.8-35178_amd64.deb zarafa-common_7.0.8-35178_amd64.deb zarafa-client_7.0.8-35178_amd64.deb zarafa-server_7.0.8-35178_amd64.deb
```

Es ist nicht notwendig vorherige Versionen zu deinstallieren, das erledigt ***dpkg*** automatisch.

# Speicherüberlauf Cyrus Index-DB

Cyrus IMAP speichert die den Mailboxen zugehörigen Index-Datenbanken unter /var/lib/imap im Berkeley-DB (sleepycat) Format ab. Die für dieses Format übliche Konfigurationsdatei „DB_CONFIG“ fehlt im Verzeichnis /var/lib/imap/db. Das bedeutet, dass die Datenbank mit Standardwerten betrieben wird. Es kann vorkommen, das der per Default vorgegebene für die Datenbanken zur Verfügung stehende maximale Speicherplatz überschritten wird. Die Folge ist, dass Cyrus die weitere Annahme in die von ihm verwalteten Postfächer verweigert.

Dieser Fehler äußert sich in /var/log/messages mit folgenden Zeilen:

```
Feb 21 11:01:47 invis5bio lmtpunix[1802]: DBERROR db4: Logging region out of memory; you may need to increase its size
Feb 21 11:01:47 invis5bio lmtpunix[1802]: DBERROR: opening /var/lib/imap/deliver.db: Cannot allocate memory
Feb 21 11:01:47 invis5bio lmtpunix[1802]: DBERROR: opening /var/lib/imap/deliver.db: cyrusdb error
Feb 21 11:01:47 invis5bio lmtpunix[1802]: FATAL: lmtpd: unable to init duplicate delivery database
Feb 21 11:01:47 invis5bio master[17315]: service lmtpunix pid 1802 in READY state: terminated abnormally
```

Behoben werden kann dieser Fehler durch erzeugen einer Konfigurationsdatei für die Berkeley-DB unter dem Namen „DB_CONFIG“ in /var/lib/imap/db:

```
set_cachesize 0 2097152 1

# Data Directory
#set_data_dir db

# Transaction Log settings
set_lg_regionmax 2097152
set_lg_bsize 2097152
set_lg_max 4194304
set_tx_max 200
set_tas_spins 1
#set_lg_dir logs
```

Daran anschließend ist Cyrus neuzustarten:

```
linux:~ #/etc/init.d/cyrus restart
```

# Group-e -- Passwwort des Benutzers "config" zurück setzen

Das Passwort des Benutzers „config“ wird unabhängig von der generellen Benutzerverwaltung in Group-e immer in der MySQL-Datenbank verwaltet. Um es zurückzusetzen muss eine SQL-Anweisung über das MySQL-Komandozeilen Frontend abgesetzt werden:

```
linux:~ # mysql -u root -p

...

mysql> use groupe;
Database changed
mysql> UPDATE `ModulCfg` SET `CfgVal`=MD5( 'config' ) WHERE
`ModulCfg`.`FKModul`='global' AND `ModulCfg`.`FKCfgKey`='cfg/pwd' AND
`ModulCfg`.`FKObjID`=0;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> quit
Bye
linux:~ #
```

Danach kann man sich wieder als Benutzer „config“ mit dem Passwort „config“ anmelden.

# Nein Danke Zensursula

**Achtung, der folgende Text ist nicht ganz neutral. Er enthält persönliche Meinugsäußerungen. Ich bitte dies zu entschuldigen.**

Ab Oktober 2009 soll das Web-Sperren-Gesetz gegen die Verbreitung von Kinderpornographie in Kraft treten. Da dieses Gesetzt, trotz des sicherlich aufrichtigen Ansatzes, am Thema vorbei geht und allenfalls eine Zensurinfrastruktur im Internet etabliert, zeigen wir hier eine Möglichkeit auf, es zu umgehen. Möglicherweise hilft das ja dabei die Wirkungslosigkeit dieses Gesetzes zu verdeutlichen.

Wer einen invis-Server im eigenen Netz betreibt und sich (zumindest vorerst) sicher sein möchte, dass seine Nameserverabfragen nicht von „manipulierten“ Nameservern beantwortet werden, kann dies durch Ändern der „forwarders“ in der bind-Konfiguration umgehen.

Ändern Sie einfach in der Datei „/etc/named.conf“ die Einträge hinter „forwarders“:

```
        # The forwarders record contains a list of servers to which queries
        # should be forwarded.  Enable this line and modify the IP address
to
        # your provider's name server.  Up to three servers may be listed.
        # Die folgende Zeile ist um die IP-Adresse des fuer Sie zustaendigen
DNS
        # zu erweitern.
```

```
        forwarders { 194.25.2.129; 192.168.178.1; };

        # Enable the next entry to prefer usage of the name server declared
in
        # the forwarders section.

        forward first;
```

Tragen Sie statt des T-Online Nameservers und etwa dem einer Fritzbox (wie hier gezeigt) bis zu drei Nameserver aus der unter http://www.ungefiltert-surfen.de zu findenden Liste ein und starten Sie ***bind*** mit:

```
Kommandozeile: /etc/init.d/named restart
```

neu.

## Faxgate-Client unter Linux nutzen

Um den Faxgate-Client unter Linux nutzen zu können sind ein paar Kleinigkeiten zu beachen:

1. Die Firewall des Linux-Clients muss Port 50000/TCP geöffnet haben.
2. Die „.jar“ Datei ist mittels der Befehlszeile „java -jar FaxgateClient.jar“ zu starten. Das lässt sich ja unter KDE oder Gnome als fertiges Icon auf den Desktop legen.
3. In der hosts-Datei von Client und Server sollten trotz funktionierendem DNS in Einträgen mit der IP „127.0.0.2“ stattdessen die realen IP-Adressen verwendet werden.
4. Da unter Linux nicht unter fremder Benutzerkennung gedruckt werden kann, muss für jeden Fax-Benutzer Serverseitig eine Fax-Konfiguration via YaST erzeugt werden. Wird dort die MSN für eingehende Faxe nicht eingetragen, kann dieser Eintrag nur für den Faxversand genutzt werden. Der Faxempfang kann so weiterhin über die Benutzerkennung „fax“ erfolgen. D.h. alle Faxe landen im speziellen Faxpostfach und sind für alle Nutzer lesbar.

From:
https://wiki.invis-server.org/ - **invis-server.org**

Permanent link:
**https://wiki.invis-server.org/doku.php?id=tipps_und_tricks&rev=1544880033**

Last update: **2018/12/15 13:20**