

Dokumentation des Schulservers

Andreas Dangel

Dokumentation des Schulservers

von Andreas Dangel

Veröffentlicht 30.01.2004 17:18:38

Copyright © 2001, 2002, 2003 Andreas Dangel

Inhaltsverzeichnis

Vorwort	i
I. Netzwerk.....	i
1. Allgemeines.....	1
1.1. Topologie	1
1.2. Technik	1
1.3. Eingesetzte Hardware	1
2. IP-Adressen	2
II. Server	3
3. Installation.....	4
3.1. Debian installieren	4
3.1.1. Debian besorgen.....	4
3.1.2. Installation starten	4
3.1.3. Partitionierung.....	4
3.1.4. Weiterer Verlauf	4
3.2. Kernel machen	5
3.3. Software-RAID.....	7
3.3.1. Voraussetzungen.....	7
3.3.2. Schritt für Schritt-Anleitung	7
3.3.3. Fehlerfall	11
3.4. APC SmartUPS (USV).....	12
3.4.1. Zweck.....	12
3.4.2. Möglichkeiten ohne Treiber	12
3.4.3. Installation des Treibers	12
3.4.4. Anzeigetools.....	13
3.5. Netzwerkkarten konfigurieren	14
3.6. T-DSL einrichten	15
3.6.1. Logdatei	16
3.7. PPP-Filter.....	17
3.8. apt-get konfigurieren.....	17
3.8.1. apt-get verwenden	18
4. Dienste.....	19
4.1. Grundlegende Dienste	19
4.1.1. DNS-Server	19
4.1.2. dhcp-Server	21
4.1.3. Mail Transfer Agent (SMTP/UUCP)	21
4.1.3.1. uucp einrichten	22
4.1.3.2. exim (MTA) einrichten	24
4.1.3.3. Regelmäßig Mails abholen	25
4.2. Fernwartung.....	26
4.2.1. SSH-Server.....	26
4.2.2. dyndns	26
4.3. Dienste für Clients	27
4.3.1. Webserver.....	27
4.3.2. pop3-Server	30
4.3.3. Proxy und Filter	31
4.3.3.1. squid	31
4.3.3.2. squidGuard	31
4.3.4. Drucker.....	34
4.3.5. Samba.....	34

4.3.5.1. Logdateien	38
4.3.5.2. Tägliches Neustarten	39
4.4. Sicherheit	39
4.4.1. Firewall.....	39
4.4.2. inetd.....	40
4.4.3. netstat und nmap	40
4.4.3.1. netstat.....	40
4.4.3.2. nmap	41
4.4.4. Backup	41
4.4.5. sudoers	42
4.5. Zusätzliches	43
4.5.1. Homepage hochladen	44
4.5.2. Quota	44
4.5.3. ntpdate	45
5. Verwaltung und Wartung	46
5.1. Benutzerverwaltung	46
5.1.1. Benutzer	46
5.1.2. Gruppen.....	46
5.2. Benutzermanager	47
5.2.1. Benutzer löschen	47
5.2.2. Passwort ändern	47
5.2.3. Rechte eines Benutzers ändern	48
5.2.4. Benutzer sperren bzw. freigeben	48
5.2.5. Neuer Benutzer erstellen.....	49
5.2.6. Informationen über die Benutzer sammeln	49
5.3. proxymanager	50
5.3.1. Logdatei	51
5.4. Info-Tools.....	52
5.4.1. Festplattenplatz	52
5.4.2. uptime.....	52
5.5. Weitere CGI-Skripte	52
5.5.1. Passwort ändern	52
5.5.2. Mail-Weiterleitung konfigurieren.....	52
5.5.3. Benutzerverzeichnisse.....	53
III. Clients.....	54
6. Windows 98SE	55
6.1. Zusammenarbeit mit dem Netzwerk und Samba.....	55
6.1.1. Netzwerk	55
6.1.2. Computer-Name	55
6.1.3. Netz-Logon	55
6.2. Netzwerkinstallation von best. Programmen	55
6.3. Partitionsmanager	56
6.4. Windows beschränken	57
IV. Sonstiges	60
7. Linux-Grundlagen	61
7.1. Einloggen/Ausloggen	61
7.2. Linux herunterfahren (shutdown)	61
7.3. Dateiverwaltung.....	61
7.3.1. Dateien auflisten.....	61
7.3.2. Datei/Verzeichnis erstellen.....	63
7.3.3. Verzeichnis wechseln	63

7.3.4. Datei/Verzeichnis löschen	63
7.3.5. Dateien kopieren	63
7.3.6. Dateien verschieben	63
7.3.7. Textdateien anzeigen	63
7.4. Benutzerverwaltung	64
7.4.1. Benutzer erstellen.....	64
7.4.2. Benutzer bearbeiten.....	64
7.4.3. Benutzer löschen	64
7.4.4. Gruppe erstellen	64
7.4.5. Gruppe löschen	64
7.5. Paketverwaltung.....	65
7.6. Editor	65
7.6.1. Datei öffnen bzw. neue Datei erstellen.....	65
7.6.2. Dateien bearbeiten und speichern	65
7.7. Prozessmanagement.....	65
7.8. Linux Support	65

Tabellenverzeichnis

2-1. IP-Adressen	2
------------------------	---

Vorwort

Diese Dokumentation beschreibt den Aufbau des Netzwerkes am Gymnasium Münsingen. Der Schwerpunkt liegt beim Linux-Server. Um die Dokumentation umsetzen zu können, sind Erfahrungen mit Linux notwendig. Möglicherweise hilft auch das sehr, sehr knapp gehaltene Kapitel über Linux-Grundlagen weiter.

Auf der beiliegenden CD sind alle Konfigurationsdateien noch einmal enthalten. Außerdem ist diese Dokumentation in verschiedenen Formaten auch auf der CD. In der Druckversion fehlen die Bilder im Kapitel über die Einrichtung der Windows-Clients. Diese Bilder sind in der HTML-Version enthalten. Außerdem sind zu lange Zeilen, wie sie z.B. in Konfigurationsdateien vorkommen, in der Druckversion abgeschnitten. Die HTML-Version ist dagegen komplett.

I. Netzwerk

Kapitel 1. Allgemeines

1.1. Topologie

Sternförmig

1.2. Technik

Technik: Ethernet

1.3. Eingesetzte Hardware

Hub in R104; Netzdrucker in R104; Weitere Rechner in: R23 (Lehrerarbeitsraum), Sekretariat, R24 (mit Beamer), Biologie-, Physik- und Chemie-Sammlung; Weitere lokale (nicht im Netz verfügbare) Drucker; ...

Kapitel 2. IP-Adressen

Tabelle 2-1. IP-Adressen

Netzwerk	192.168.0.0
Netzmaske	255.255.255.0 (kurz: /24)
Broadcast	192.168.0.255
Server-IP	192.168.0.1
Netzdrucker R104	192.168.0.254
Oki-Page Netzwerkdruker R23	192.168.0.139
Farblaserdrucker Minolta R23	192.168.0.138
Servername	server
Domain	gm.rt.schule-bw.de

II. Server

Kapitel 3. Installation

3.1. Debian installieren

3.1.1. Debian besorgen

Auf dem Server soll Debian 3.0 ("Woody") installiert werden. Die Debian-Linux-Distribution wird komplett unter der GNU GPL 2 entwickelt. Daher lassen sich auch die CDs ganz legal aus dem Internet herunterladen. Wo und wie das geht, steht auf www.debian.de (<http://www.debian.de>). Insgesamt müssen 7 CDs heruntergeladen werden.

3.1.2. Installation starten

Um die Installation zu starten, muss von der ersten CD gebootet werden. Im Bootmenü der CD **bf24** eingeben und die Eingabetaste drücken. So wird der 2.4-Kernel gebootet. Dieser wird benötigt, damit wir das neue ReiserFS-Dateisystem verwenden können.

3.1.3. Partitionierung

Debian wird auf die erste Festplatte installiert. Diese muss allerdings noch partitioniert werden. Es wird von einer 80 GB großen Festplatte ausgegangen. Auf dieser werden drei primäre Partitionen benötigt:

1. `/dev/hda1`: die root-Partition (/); ca. 70 GB; Dateisystem: ReiserFS
2. `/dev/hda2`: die var-Partition (/var); ca. 10 GB; Dateisystem: ReiserFS; hier liegen verschiedene Daten, z.B. Emails.
3. `/dev/hda3`: die swap-Partition; ca. 500 MB

3.1.4. Weiterer Verlauf

Einfach den Anweisungen am Bildschirm folgen. Spezielle Module (= Treiber) brauchen nicht installiert werden. lilo kann in den MBR der ersten Festplatte installiert werden. Nach einem Neustart geht die Installation weiter. Wir verwenden keine MD5 verschlüsselte Passwörter, da wir die Benutzerdatenbank des alten Servers weiterverwenden wollen. Danach kommt die Frage nach dem root-Passwort; dieses Passwort unbedingt sehr gut merken. Einen normalen Benutzer legen wir nicht an. Die Programme tasksel und dselect verwenden wir nicht. Wir installieren die Programme, die benötigt werden, bei Bedarf. So haben wir eine minimalste Software-Auswahl und keine unnötige Programme. Wenn die Konfiguration des Mail-Servers exim kommt, die Option 5 wählen: Wir konfigurieren den Mail-Server später.

Nachdem die Installation vorüber ist, installieren wir zunächst noch einige wichtige Programme: less, vim, bzip2 und aptitude. less dient zum Anzeigen von Dateien, vim ist ein einfacher Editor, mit bzip2 lassen sich Dateien

komprimieren und aptitude ist ein einfach zu bedienendes Installationsprogramm für die Debian-Pakete. Die Installation dieser Pakete funktioniert als root mit **apt-get install less vim bzip2 aptitude**.

Um als root bei der Ausgabe von **ls** Farben zu erhalten, muss die Datei `/root/.bashrc` geändert werden. Es müssen lediglich die Kommentar-Zeichen vor einigen Zeilen entfernt werden, sodass die Datei ungefähr so aussieht:

```
#
# /root/.bashrc
#

[...]

# You may uncomment the following lines if you want 'ls' to be colorized:
export LS_OPTIONS='--color=auto'
eval `dircolors`
alias ls='ls $LS_OPTIONS'
alias ll='ls $LS_OPTIONS -l'
alias l='ls $LS_OPTIONS -lA'

[...]
```

3.2. Kernel machen

Hier wird beschrieben, wie ein Kernel aus den Quellcodes gemacht wird. Dazu werden mehrere Schritte benötigt.

1. Zunächst einmal muss dafür gesorgt werden, dass die benötigten Programme und Bibliotheken installiert sind. Wenn der Kernel zum ersten Mal nach der Installation von Debian gemacht wird, sind noch folgende Programme nachzuinstallieren: `binutils`, `cpp`, `g++`, `gcc`, `make`, `patch`, `libncurses5-dev`. Das geht mit folgendem Befehl in einem Rutsch (als root):

```
bash# apt-get install binutils cpp g++ gcc make patch libncurses5-dev
```
2. Jetzt muss der Kernel noch besorgt werden. Den aktuellen Quellcode gibt's unter www.kernel.org (<http://www.kernel.org>). Die Datei heißt z.B. `linux-2.4.21.tar.bz2` und ist ca. 30 MB groß.
3. Um den Hardware-Sensor (Temperaturanzeige) anzusteuern, werden noch extra Treiber benötigt. Der Kernel muss gepatcht werden. Die dafür notwendigen Dateien sind bei secure.netroedge.com/~lm78/ (<http://secure.netroedge.com/~lm78/>) zu finden. Es werden zwei Dateien benötigt. Sie heißen etwa `i2c-2.8.0.tar.gz` und `lm_sensors-2.8.0.tar.gz`.

Außerdem wird ein Patch benötigt, damit Quota unter dem ReiserFS-Dateisystem funktioniert. Mit Quota lässt sich die Festplattenbenutzung pro Benutzer begrenzen. Patches für aktuelle Kernel sind unter <ftp://ftp.suse.com/pub/people/mason/patches/data-logging/> zu finden. Im Verzeichnis `2.4.21/` sind verschiedene Patches für den Kernel der Version 2.4.21 vorhanden. Für Quota werden nur `07-quota-v2-2.4.21.diff.gz`, `08-reiserfs-quota-28.diff.gz` und `09-kinoded-8.diff.gz` benötigt. Näheres steht in der README-Datei.

4. Nachdem jetzt alle benötigten Quellcodes zusammen sind, muss der Kernel zuerst mal entpackt werden. Das geschieht am besten in einem Unterverzeichnis im Homeverzeichnis von root (`/root/kernel/`) mit diesen Befehlen:

```
bash# cd /root
bash# mkdir kernel
bash# cd kernel
bash# tar xfv /root/linux-2.4.21.tar.bz2
```

Dabei wird angenommen, dass alle Dateien, die heruntergeladen wurden, sich im Homeverzeichnis von root (`/root/`) befinden.

Nun kommen die Patches für den Hardware-Sensor dran. Folgende Schritte sind auszuführen, um die Treiber zu entpacken und den Kernel anschließend zu patchen:

```
bash# cd /root/kernel
bash# tar xfv /root/i2c-2.8.0.tar.gz
bash# cd i2c-2.8.0
bash# mkpatch/mkpatch.pl . ../linux > ../i2c-patch
bash# cd ../linux
bash# patch -p1 -E < ../i2c-patch

bash# cd /root/kernel
bash# tar xfv /root/lm_sensors-2.8.0.tar.gz
bash# cd lm_sensors-2.8.0
bash# mkpatch/mkpatch.pl . ../linux > ../lm_sensors-patch
bash# cd ../linux
bash# patch -p1 -E < ../lm_sensors-patch
```

Jetzt muss noch der ReiserFS-Quota-Patch verwendet werden. Das geht mit folgenden Befehlen:

```
bash# cd /root/kernel/linux
bash# zcat /root/07-quota-v2-2.4.21.diff.gz | patch -p1
bash# zcat /root/08-reiserfs-quota-28.diff.gz | patch -p1
bash# zcat /root/09-kinoded-8.diff.gz | patch -p1
```

5. Jetzt kann der Kernel endlich konfiguriert werden. Das Konfigurationsprogramm wird mit **make menuconfig** im Verzeichnis `/root/kernel/linux/` gestartet.

Software-RAID, e100 (Intel-Netzwerkkarte), 3c59x (3com-Netzwerkkarte)

6. Nach der Konfiguration kann der Kernel mit den Modulen compiliert werden. Einfach **make dep bzImage modules modules_install** eingeben.
7. Der neue Kernel muss jetzt noch installiert werden. Mit **cp /root/kernel/linux/arch/i386/boot/bzImage /boot/vmlinuz-2.4.21** das Kernel-Image an die richtige Stelle kopieren. Danach muss `/etc/lilo.conf` bearbeitet werden: Die erste Zeile, die mit "image=" anfängt, muss in "image=/boot/vmlinuz-2.4.21" abgeändert werden. Danach mit **lilo** lilo neu installieren und den Computer anschließend neu starten.
8. Nach dem Neustart können mit **modconf** die erstellten Module so konfiguriert werden, dass sie beim nächsten Neustart automatisch geladen werden. Wichtige Module sind hier die Module für den Hardware-Sensor (i2c-isa, w83..d), die Netzwerkkarten (e100, 3c59x) und das Modul für APM (Advanced

Power Management; apm). Damit die Treiber für den Hardware-Sensor einen Sinn ergeben, wird noch das Paket *lm-sensors* benötigt. Es wird mit **apt-get install lm-sensors** installiert. Wenn die Treiber geladen sind, kann mit **sensors** die aktuelle Temperatur angezeigt werden.

3.3. Software-RAID

Software-RAID dient zur besseren Sicherheit der Daten auf dem Server. RAID kann in verschiedenen Modi betrieben werden. Der Modus, der weiter unten eingerichtet wird, heißt RAID-1. Dabei befinden sich im Server zwei Festplatten annähernd gleicher Kapazität. Der Software-RAID-Treiber von Linux sorgt dann dafür, dass sich auf beiden Festplatten exakt der gleiche Inhalt befindet. Die Festplatten werden sozusagen gespiegelt. Fällt eine Festplatte aus, so ist die andere immer noch voll funktionsfähig.

Die Anleitung weiter unten beschreibt, wie ein vorhandenes Linux-System nachträglich in ein RAID-1-System umgewandelt werden kann. Bei modernen Linux-Distributionen kann man möglicherweise schon bei der Installation Software-RAID einrichten; dann kann dieser Abschnitt komplett übersprungen werden.

3.3.1. Voraussetzungen

Um Software-RAID einzurichten, werden zunächst einmal die *raidtools* in der Version 0.9 oder höher benötigt; ob die *raidtools* bereits installiert sind, lässt sich mit **mkraid -V** herausfinden. Das passende Debian-Paket zum Nachinstallieren heißt *raidtools2*. Dann muss sehr wahrscheinlich noch ein neuer Kernel der Version 2.4.x gemacht werden; dieser Vorgang ist genauer in Abschnitt 3.2 beschrieben.

Natürlich werden auch zwei Festplatten benötigt, die ungefähr die gleiche Größe haben sollten. Diese sind - sofern IDE-Festplatten - idealerweise jeweils als Master angeschlossen und als `/dev/hda` und `/dev/hdc` ansprechbar.

Die Ausgangssituation ist folgende: Auf der ersten Festplatte (`/dev/hda`) ist bereits ein Linux-System installiert. Dabei ist es nicht wichtig, dass es konfiguriert ist, denn das lässt sich später noch genau so gut konfigurieren. Es ist sogar eher noch besser, wenn Software-RAID vor der eigentlichen Konfiguration des Systems eingerichtet wird. Die Anleitung bezieht sich auf die Linux-Installation, die weiter oben beschrieben ist.

3.3.2. Schritt für Schritt-Anleitung

Alle folgenden Schritte werden als Superuser (root) durchgeführt.

1. Zuerst muss ein frischer Kernel gemacht werden. Dabei sind folgende Einstellungen zu machen:

```
"Multi-device support (RAID and LVM)"
[y] Multiple devices driver support (RAID and LVM)
[y] RAID support
[y] RAID-1 (mirroring) mode
```

Wichtig ist hier, dass der RAID-Treiber fest im Kernel ist und keine Module erstellt werden.

Eine genaue Anleitung für die Erstellung und Installation des neuen Kernels ist in Abschnitt 3.2 beschrieben.

2. Jetzt muss der neue Kernel getestet werden, d.h. es muss mit dem neuen Kernel gebootet werden. Wenn alles soweit funktioniert hat, kann mit dem nächsten Schritt weitergemacht werden. Übrigens: Der Kernel kann auch später nochmal aktualisiert werden; spezielle Hardware kann also später eingerichtet werden.
3. Nach dem Neustart kann mit **cat /proc/mdstat** überprüft werden, ob der neue Kernel auch tatsächlich RAID unterstützt. Dabei sollte in der ersten Zeile folgendes erscheinen:

```
Personalities : [raid1]
```

4. Als nächstes muss die zweite Festplatte so wie die erste Festplatte partitioniert werden. Dabei dürfen die Partitionen auf der zweiten Festplatte auf keinen Fall größer sein als die Partitionen auf der ersten Festplatte. Sie müssen entweder gleich groß oder kleiner sein.

Die Partitionierung kann z.B. mit dem Programm **cfdisk** vorgenommen werden. Am besten speichert man die Partitionstabelle der ersten Festplatte mit **cfdisk -P t /dev/hda > hda-table** in die Datei `hda-table`. Das sieht dann so aus:

Partition Table for /dev/hda

---Starting---					----Ending----				Start	Number of
#	Flags	Head	Sect	Cyl	ID	Head	Sect	Cyl	Sector	Sectors
1	0x80	1	1	0	0x83	15	63	1023	63	135742257
2	0x00	15	63	1023	0x83	15	63	1023	135742320	19531008
3	0x00	15	63	1023	0x82	15	63	1023	155273328	1028160
4	0x00	0	0	0	0x00	0	0	0	0	0

Wichtig sind die zwei letzten Spalten (Start Sector und Number of Sectors). Mit **cfdisk /dev/hdc** wird die zweite Festplatte partitioniert; es müssen genau die gleichen Partitionen erstellt werden. Das geht am einfachsten über die Sektor-Anzahl. Wenn das Programm nach der Größe fragt, einfach **12345S** eingeben, also die Anzahl der Sektoren mit angehängtem S. Zum Schluss müssen die Partitionstabellen der beiden Festplatten noch einmal verglichen werden, damit auch kein Fehler unterlief; also: Partitionstabelle der zweiten Festplatte wie oben in eine Datei speichern.

Hier sollte dann auch auf der zweiten Festplatte gleich das Bootflag gesetzt werden (auf `/dev/hda1`). Außerdem müssen alle Partitionen außer der swap-Partition den Partitionstyp *fd* (Linux raid autodetect) haben.

5. Jetzt muss die Konfigurationsdatei für den Software-RAID-Treiber geschrieben werden. Diese steht in `/etc/raidtab`. Im Moment ist dabei die erste Festplatte (`/dev/hda`) als fehlerhaft zu markieren.

```
#
# /etc/raidtab
#

# /dev/md0 ist die root-Partition
raiddev                /dev/md0
raid-level              1                # RAID-1 (mirroring)
nr-raid-disks          2
nr-spare-disks         0
persistent-superblock  1
chunk-size             32
```



```

device                /dev/hdc1
raid-disk              0
# dies ist die erste Festplatte, die hier zunächst als fehlerhaft
# markiert ist
device                /dev/hda1
failed-disk            1

# /dev/md1 ist die var-Partition
raiddev               /dev/md1
raid-level             1                # RAID-1 (mirroring)
nr-raid-disks         2
nr-spare-disks        0
persistent-superblock 1
chunk-size            32

device                /dev/hdc2
raid-disk              0
# die erste Festplatte ist hier genauso als fehlerhaft markiert
device                /dev/hda2
failed-disk            1

```

6. Danach können die raid-Geräte gemacht werden:

```

bash# mkraid /dev/md0
bash# mkraid /dev/md1

```

Mit **cat /proc/mdstat** kann nachgeschaut werden, ob die Geräte registriert sind. Die erste Festplatte ist hier als fehlerhaft eingetragen.

7. Anschließend können die raid-Geräte formatiert und gemountet werden. Da die erste Festplatte ja als fehlerhaft markiert ist, wird sie nicht formatiert, sondern nur die zweite Festplatte. Die Daten auf der ersten Festplatte bleiben also vorhanden.

Das Formatieren funktioniert so:

```

bash# mkreiserfs /dev/md0
bash# mkreiserfs /dev/md1

```

Für das Mounten der Dateisysteme müssen folgende Befehle ausgeführt werden:

```

bash# mount /dev/md0 /mnt
bash# mkdir /mnt/var
bash# mount /dev/md1 /mnt/var

```

8. Jetzt können die Dateien der ersten Festplatte auf die zweite Festplatte kopiert werden. Das geht ganz einfach mit folgenden Befehlen:

```

bash# cd /
bash# find . -xdev -name "*" -o -name ".*" | cpio -pmv /mnt
bash# cd /var
bash# find . -xdev -name "*" -o -name ".*" | cpio -pmv /mnt/var

```

9. Danach muss wieder getestet werden. Wir aktualisieren dafür zuerst folgende Datei:

```
#
# /mnt/etc/fstab
#

/dev/md0      /          reiserfs    defaults    0           0
/dev/md1      /var       reiserfs    defaults    0           0
[...]
```

Wir ersetzen also `/dev/hda1` mit `/dev/md0` und `/dev/hda2` mit `/dev/md1`.

Beim nächsten Start mit dem neu eingerichteten RAID werden so die raid-Geräte automatisch verwendet.

10. Nun muss noch eine Bootdiskette erstellt werden. Dazu wird eine leere, formatierte und fehlerfreie Diskette benötigt. Die Diskette in Laufwerk A: einlegen und folgende Befehle ausführen:

```
bash# dd if=/boot/vmlinuz-2.4.21 of=/dev/fd0 bs=18k
bash# rdev /dev/fd0 /dev/md0
bash# rdev -r /dev/fd0 0
bash# rdev -R /dev/fd0 1
```

Auf der so erstellten Bootdiskette ist der neue Kernel mit Software-RAID-Unterstützung (`/boot/vmlinuz-2.4.21` ist das Kernel-Image). Diese Bootdiskette kann später vielleicht mal sehr wichtig werden, z.B. wenn der Server aus irgendeinem Grund nicht mehr von der Festplatte booten kann. Deshalb sollte die Bootdiskette gut aufbewahrt werden und bei jedem Kernel-Update auch aktualisiert werden.

11. Jetzt kann der Server neu gestartet werden. Dazu müssen aber erst die raid-Geräte gestoppt werden:

```
bash# umount /mnt/var
bash# umount /mnt
bash# raidstop /dev/md0
bash# raidstop /dev/md1
```

Wenn die Bootdiskette eingelegt wurde, kann mit **shutdown -r now** der Server neu gestartet werden.

12. Nach dem Neustart muss geprüft werden, ob alle Daten auf die zweite Festplatte kopiert wurden. Wenn der Server ohne Probleme neugestartet ist, sollte dies der Fall sein. Wenn nicht, kann der Server immer noch ohne Diskette von der ersten Festplatte gestartet werden. Dann müssen wie in Punkt 7 beschrieben die raid-Geräte wieder gemountet werden, damit eventuell vergessene Dateien kopiert werden können.
13. Wir haben von der Diskette gebootet. Jetzt muss die erste Festplatte, die im Moment ja noch als fehlerhaft markiert ist, dem raid-Array hinzugefügt werden. Dabei gehen alle Daten auf dieser Festplatte verloren!

Als erstes müssen mit **cfdisk /dev/hda** die Partitionstypen der Partitionen auf der ersten Festplatte zu *fd* (Linux raid autodetect) geändert werden (die swap-Partition bleibt swap). Auch das Bootflag sollte nochmal frisch auf `/dev/hda1` gesetzt werden.

Dann muss in der Datei `/etc/raidtab` zweimal *failed-disk* durch *raid-disk* ersetzt werden.

Und schließlich wird mit folgenden Befehlen die erste Festplatte dem raid-Array hinzugefügt:

```
bash# raidhotadd /dev/md1 /dev/hda2
bash# raidhotadd /dev/md0 /dev/hda1
```

Mit **cat /proc/mdstat** kann der Fortschritt der Synchronisation angezeigt werden.

14. Während der Synchronisationsvorgang noch läuft, kann das System schon bootfähig gemacht werden. Dazu muss `/etc/lilo.conf` editiert werden. Die Datei sieht dann so aus:

```
#
# /etc/lilo.conf
#

boot=/dev/md0
raid-extra-boot=/dev/hda,/dev/hdc

read-only
prompt
timeout=50
root=/dev/md0

image=/boot/vmlinuz-2.4.21
    label=Linux
```

Jetzt muss noch lilo auf beiden Festplatten installiert werden. Dies geschieht einfach durch Aufruf von **lilo**.

15. Schließlich muss noch die swap-Partition auf der zweiten Festplatte eingerichtet werden. Dazu muss `/etc/fstab` verändert werden:

```
#
# /etc/fstab
#

[ ... ]
/dev/hda2      none      swap      sw,pri=1      0          0
/dev/hdc2      none      swap      sw,pri=1      0          0
```

Außerdem muss die swap-Partition noch mit **mkswap /dev/hdc2** formatiert werden. Beim nächsten Neustart sind dann beide swap-Partitionen aktiviert.

16. Wenn der Synchronisationsvorgang beendet ist, kann neu gestartet werden, um die Installation zu testen.

Falls das Booten von der Festplatte nicht funktioniert, kann immer noch auf die Bootdiskette zurückgegriffen werden.

3.3.3. Fehlerfall

Fällt einmal tatsächlich eine Festplatte aus, muss diese ersetzt werden. Im Folgenden wird angenommen, dass die erste Festplatte (`/dev/hda`) ausfällt. Die einzelnen Schritte sind dann wie folgt:

1. Erst muss der Server heruntergefahren werden, wenn er noch läuft.
2. Dann kann die defekte Festplatte ausgetauscht werden.
3. Jetzt muss gebootet werden. Wenn der Server nicht automatisch von der zweiten (intakten) Festplatte bootet, muss eine Bootdiskette verwendet werden (siehe voriger Abschnitt, Punkt 10).
4. Die neue Festplatte muss nun partitioniert werden.
5. Mit dem Befehl **raidhotadd** den Synchronisationsvorgang starten:

```
bash# raidhotadd /dev/md1 /dev/hda2
```

```
bash# raidhotadd /dev/md0 /dev/hda1
```

6. Mit **lilo** lilo neu installieren.

7. Nach dem Synchronisationsvorgang kann von der Festplatte neu gebootet werden.

Hinweis: Falls das BIOS nicht von der zweiten Festplatte booten kann, wird unbedingt eine Bootdiskette benötigt.

3.4. APC SmartUPS (USV)

3.4.1. Zweck

Eine USV (Unterbrechungsfreie Stromversorgung) soll, wie der Name schon sagt, die Stromversorgung sicherstellen. Dabei kann man sich die USV als einen sehr großen Akku vorstellen. Fällt der Strom aus, dann wird der Server weiterhin mit Strom versorgt - aus dem Akku. Außerdem schützt die USV den Server vor Überspannung.

3.4.2. Möglichkeiten ohne Treiber

Wenn die USV einfach so, ohne Treiber an den Server angeschlossen wird, dann springt sie automatisch ein, wenn der Strom ausfällt. Wenn der Akku leer ist, fällt der Strom komplett aus; der Server wird nicht heruntergefahren! Die USV ohne Treiber ist also nur für kurze Stromausfälle geeignet.

3.4.3. Installation des Treibers

Wenn der Treiber installiert ist, meldet die USV den Stromausfall. Dann kann der Server entsprechend reagieren. Reagiert er nicht, dann fährt die USV den Server herunter, sobald die Akkuladung einen kritischen Wert erreicht hat. Der Server kann aber auch reagieren und z.B. nach 60 Sekunden herunterfahren. Dadurch kann verhindert werden, dass der Akku beim möglicherweise etwas langen Herunterfahren versagt.

Die USV wird über die erste serielle Schnittstelle (COM1) an den Server angeschlossen. Zur Installation des Treibers wird das Programmpaket **nut** benötigt. Es kann mit **apt-get install nut** installiert werden. Konfiguriert wird der Treiber über die Datei `/etc/nut/upsd.conf`. Hier muss zunächst folgende Zeile hinzugefügt werden:

```
UPS myups /var/lib/nut/apcsmart-ttyS0 apcsmart -x cable=940-0095B /dev/ttyS0
```

Danach muss mit **chown root.nut /dev/ttyS0** dem Treiber den Zugriff auf die COM1-Schnittstelle erlaubt werden. Dann muss `/etc/nut/upsd.conf` weiter bearbeitet werden: Die Zeile mit `"#ACCESS grant master localhost [<upsmon-password>]"` suchen und ändern in:

```
ACCESS grant master localhost password
```

Damit wird dem Monitorprogramm erlaubt, den Treiber zu befragen. Dabei wird das Passwort "password" verwendet. Der Treiber und das Monitorprogramm müssen aber auf demselben Rechner laufen, sodass das Passwort nicht so sicherheitsrelevant ist.

In der Datei `/etc/nut/upsmon.conf` muss im ersten Abschnitt noch folgende Zeile ergänzt werden:

```
MONITOR myups@localhost 1 password master
```

Damit ist die einfache Konfiguration abgeschlossen. Mit `/etc/init.d/nut start` wird der Treiber geladen.

Damit der Server nach einer bestimmten Zeit nach dem Stromausfall automatisch herunterfährt, ist weitere Arbeit nötig. In der Datei `/etc/nut/upsmon.conf` müssen in den entsprechenden Abschnitten folgende Zeilen hinzugefügt werden:

```
NOTIFYCMD /sbin/upssched
NOTIFYFLAG ONLINE SYSLOG+EXEC
NOTIFYFLAG ONBATT SYSLOG+WALL+EXEC
```

Wenn der Strom ausfällt (ONBATT) oder wieder da ist (ONLINE) wird das Programm `/sbin/upssched` ausgeführt. Jetzt muss die Datei `/etc/nut/upssched.conf` mit folgendem Inhalt erstellt werden:

```
#
# /etc/nut/upssched.conf
#

CMDSCRIPT /usr/local/sbin/go-down
PIPEFN /var/run/upssched.pipe
AT ONBATT * START-TIMER onbattwarn 60
AT ONLINE * CANCEL-TIMER onbattwarn
```

Mit dieser Konfiguration wird bei Stromausfall (ONBATT) ein 60-Sekunden-Timer gestartet, der nach Ablauf das Script `/usr/local/sbin/go-down` ausführt. Wird die Stromversorgung vor Ablauf des Timers wiederhergestellt (ONLINE), wird der Timer abgebrochen.

Das Script `/usr/local/sbin/go-down` sieht so aus:

```
#!/bin/sh
#
# /usr/local/sbin/go-down
#
upsmon -c fsd
```

Die Zugriffsrechte für dieses Script werden mit **`chmod 700 /usr/local/sbin/go-down`** richtig gesetzt. Mit **`/etc/init.d/nut restart`** wird die neue Konfiguration verwendet. Damit ist die Konfiguration komplett abgeschlossen.

Jetzt sollte die USV noch getestet werden. Wird das Stromkabel an der USV herausgezogen, sollte in `/var/log/syslog` eine Meldung erscheinen. Nach 60 Sekunden sollte der Server herunterfahren.

3.4.4. Anzeigetools

Die Anzeigetools sind CGI-Programme für einen Webserver. Sie können also erst richtig genutzt werden, wenn ein Webserver läuft. Trotzdem können sie auch schon jetzt eingerichtet werden.

Mit **apt-get install nut-cgi** werden die Programme installiert. Zur Konfiguration muss zunächst die Datei `/etc/nut/hosts.conf` erstellt werden:

```
#
# /etc/nut/hosts.conf
#
MONITOR myups@localhost "Local UPS"
```

Weiter muss die Datei `/etc/nut/upsset.conf` erstellt werden. Diese Datei ist nötig, um dem `upsset.cgi`-Programm mitzuteilen, dass die CGI-Programme auf dem Webserver gesichert sind und nur vom internen Netzwerk zu erreichen sind. Das muss bei der späteren Konfiguration des Webserver beachtet werden! Die Datei hat folgenden Inhalt:

```
#
# /etc/nut/upsset.conf
#
I_HAVE_SECURED_MY_CGI_DIRECTORY
```

Die CGI-Skripte liegen in `/usr/lib/cgi-bin/nut/` und sollten, wenn der Webserver konfiguriert ist, in ein entsprechendes `cgi-bin`-Verzeichnis kopiert werden.

3.5. Netzwerkkarten konfigurieren

Im Server sind zwei Netzwerkkarten. Eine für das Netzwerk und die andere für den DSL-Anschluss. Damit die Netzwerkkarten konfiguriert werden können, müssen zunächst einmal die Treiber geladen sein. Der Befehl **ifconfig -a** gibt Aufschluss über die installierten Netzwerk-Geräte:

```
bash# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:10:DC:CD:61:59
          BROADCAST MULTICAST  MTU:1500  Metric:1
          [...]

eth1      Link encap:Ethernet  HWaddr 00:50:DA:90:8E:A9
          BROADCAST MULTICAST  MTU:1500  Metric:1
          [...]

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          [...]
```

Hier sind also zwei Ethernet-Netzwerkkarten (`eth0` und `eth1`) und ein loopback-Gerät installiert (`lo`). `eth0` ist die Netzwerkkarte onboard, also die mit dem Treiber `e100`. Dieser Treiber wurde vor dem der zweiten Netzwerkkarte `eth1` geladen (`3c59x`).

Zur Konfiguration: Die Einstellungen wie IP-Adresse stehen in der Datei `/etc/network/interfaces`. Diese muss so aussehen:

```
#
# /etc/network/interfaces
#
```

```

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    network 192.168.0.0
    broadcast 192.168.0.255

auto eth1
iface eth1 inet static
    address 192.168.22.1
    netmask 255.255.255.0
    network 192.168.22.0
    broadcast 192.168.22.255

```

Der Server erhält also die IP-Adresse 192.168.0.1. Die zweite Netzwerkkarte (eth1) wird mit einer fiktiven IP-Adresse konfiguriert, die nie wirklich benutzt wird. Sie ist vielmehr ein Platzhalter. Über diese zweite Netzwerkkarte läuft später der DSL-Verkehr.

Um die Konfiguration zu übernehmen muss noch der Befehl **/etc/init.d/networking restart** ausgeführt werden. Beim nächsten Neustart wird die Konfiguration automatisch übernommen.

3.6. T-DSL einrichten

Zunächst muss das DSL-Modem mit der zweiten Netzwerkkarte verbunden werden. Die zweite Netzwerkkarte ist bereits wie oben beschrieben konfiguriert (IP-Adresse: 192.168.22.1). Um den DSL-Anschluss zu konfigurieren, werden drei Pakete benötigt, die mit **apt-get install pppoe pppoeconf pppstatus** installiert werden.

Eine weitere Voraussetzung für den DSL-Zugang sind bestimmte Treiber, die als Kernel-Module geladen werden können. Es wird auf jeden Fall das Modul *ppp_async* benötigt. Das Modul *pppoe* wird nicht benötigt; die Aufgabe dieses Treibers wird von der DSL-Software übernommen.

Nach der Installation der Software-Pakete startet automatisch das Programm **pppoeconf**; falls nicht, muss es manuell gestartet werden: einfach **pppoeconf** eingeben. Im ersten Dialog sollten beide Netzwerkkarten (eth0 und eth1) zu sehen sein. Das Programm sucht dann an beiden Netzwerkkarten automatisch nach einem DSL-Modem. Einfach den Schritten auf dem Bildschirm folgen.

Der Benutzername für die DSL-Einwahl wird aus den Daten von T-Online aufgebaut:

```

Anschlusskennung: 000111111111
T-Online-Nummer:  2222222222
Mitbenutzer-Suffix: 0001

```

Daraus ergibt sich der Benutzername: **000111111111222222222220001@t-online.de**. Dann wird natürlich noch das Passwort benötigt. Die Frage, ob */etc/resolv.conf* beim Verbindungsaufbau mit DNS-Servern vom Provider gefüllt werden soll, ist mit nein zu beantworten. Wir verwenden später unseren

eigenen DNS-Server. Nach einigen weiteren Fragen, die wie empfohlen beantwortet werden sollten, ist die Konfiguration abgeschlossen.

Mit dem Programm **pppstatus** kann der Status der DSL-Verbindung abgefragt werden. Mit **pon dsl-provider** wird eine Verbindung aufgebaut, mit **poff** wieder abgebaut.

Um "dial on demand" einzurichten, also die Einwahl bei Bedarf, muss die Datei `/etc/ppp/peers/dsl-provider` bearbeitet werden. In "Section 2" müssen folgende zwei Zeilen stehen:

```
demand
idle 120
```

Dann wird die Verbindung bei Bedarf aufgebaut und wenn 120 Sekunden lang keine Daten gesendet oder empfangen werden, wird die Verbindung getrennt.

Damit der ppp-Dämon bei Systemstart geladen wird und somit die Internetverbindung verfügbar ist, muss im Verzeichnis `/etc/ppp/` mit dem Befehl **ln -s ppp_on_boot.dsl ppp_on_boot** ein symbolischer Link erstellt werden. Außerdem sollte in der Datei `/etc/ppp/ppp_on_boot.dsl` die Variable *INTERFACE* mit `eth1` belegt sein; dies ist gegebenenfalls anzupassen.

3.6.1. Logdatei

Soll mitgeloggt werden, wann der Server online geht und wann wieder offline, muss in `/etc/ppp/ip-up.d/` und `/etc/ppp/ip-down.d/` jeweils ein Script installiert werden. Das erste Script heißt `/etc/ppp/ip-up.d/log` und muss mit folgendem Inhalt erstellt werden:

```
#!/bin/sh
#
# /etc/ppp/ip-up.d/log
#

logfile="/var/log/dsl-einwahl"
date=`date`
echo $date up >> $logfile
```

Das zweite Script heißt `/etc/ppp/ip-down.d/log` und muss mit folgendem Inhalt erstellt werden:

```
#!/bin/sh
#
# /etc/ppp/ip-down.d/log
#

logfile="/var/log/dsl-einwahl"
date=`date`
echo $date down >> $logfile
```

Beide Scripte werden mit **chmod 755 /etc/ppp/ip-*.d/log** ausführbar gemacht. In der Datei `/var/log/dsl-einwahl` steht dann, wann sich der Server entweder eingewählt oder ausgewählt hat.

Damit die Logdatei nicht ins Unendliche anwächst, kann noch logrotate konfiguriert werden. Dafür wird die Datei `/etc/logrotate.d/dsl-einwahl` mit folgendem Inhalt erstellt:


```
/var/log/dsl-einwahl {
    rotate 7
    weekly
    compress
    missingok
    notifempty
}
```

Jetzt wird jede Woche die Logdatei "rotiert", d.h. die Datei wird umbenannt und komprimiert. Es werden maximal 7 Dateien behalten, also man kann die DSL-Einwahl 7 Wochen zurückverfolgen.

3.7. PPP-Filter

Der Internet-Zugang ist oben so konfiguriert worden, dass sich der Server automatisch (bei Bedarf) einwählt und wenn in 120 Sekunden keine Daten mehr gesendet oder empfangen wurden, beendet er die Verbindung wieder. Das Problem dabei sind heute die sogenannten Peer-to-Peer-Tauschbörsen wie eDonkey. So kommen relativ oft unerwünschte Pakete über den T-Online-Zugang am Server an. Sie stellen zwar kein Sicherheitsrisiko dar, zählen aber trotzdem zum Traffic. Linux registriert, dass Daten empfangen wurden und beendet die Verbindung nicht. Um dieses Szenario zu verhindern, muss Linux so eingerichtet werden, dass es nur noch ausgehende Datenpakete zählt. Da aber die Pakete trotzdem empfangen werden, reagiert Linux normalerweise mit einer Fehlerpaket als Antwort; die muss später noch mit einer Firewall verhindert werden.

Um den Filter einzurichten, muss am Ende der Datei `/etc/ppp/options` folgende Zeile hinzugefügt werden:

```
active-filter 'outbound'
```

Mit dem Befehl `/etc/init.d/ppp restart` wird diese Konfiguration übernommen. *Hinweis:* Dabei wird eine bestehende Internetverbindung getrennt.

3.8. apt-get konfigurieren

Das Programm **apt-get** wird immer benutzt, wenn neue Software installiert werden soll. Da wir über eine DSL-Verbindung ins Internet verfügen, kann apt-get so eingerichtet werden, dass es die neue Software automatisch aus dem Internet lädt. Somit entfällt das lästige CD-Einlegen. Außerdem lässt sich so das Debian-System einfach aktuell halten.

Zunächst löschen wir die alte Konfiguration: **rm /etc/apt/sources.list**. Dann rufen wir das Setup-Programm auf: **apt-setup**. Hier wählen wir bei der ersten Frage *http* aus. Die nächste Frage beantworten wir mit *No* (wir wollen keine Software verwenden, die nicht frei ist). Im nächsten Dialog wählen wir den Mirror-Server aus: Germany, ftp.de.debian.org. Die nächste Frage kann vorerst unbeantwortet bleiben. Wir verwenden im Moment noch keinen Proxy-Server. Später, wenn der Proxy-Server eingerichtet ist, sollte apt-setup nochmal aufgerufen werden und als Proxy-Server sollte *http://localhost:3128* eingegeben werden. Danach wollen wir keine weiteren Quellen mehr hinzufügen (*No*). Die letzte Frage bezieht sich auf Sicherheitsupdates. Sie sollte unbedingt mit *Yes* beantwortet werden.

Ein weiteres, sehr nützliches Programm ist **aptitude**. Mit diesem Programm kann man herausfinden, welche Software-Pakete zur Verfügung stehen und welche Software upgedatet werden kann. Falls es noch nicht installiert ist, kann dies mit **apt-get install aptitude** nachgeholt werden.

3.8.1. apt-get verwenden

Mit **apt-get update** werden die Paketlisten neu heruntergeladen. Mit **apt-get upgrade** werden alle auf dem System installierten Pakete auf die aktuelle Version gebracht. (Davor sollte immer **apt-get update** aufgerufen werden.) Mit **apt-get install paketname** wird das Programmpaket mit dem Namen "paketname" heruntergeladen und installiert. Mit **apt-get remove paketname** wird ein Programmpaket wieder entfernt (deinstalliert). Mit **apt-get check** wird überprüft, ob sich irgendwelche Unstimmigkeiten in den Paketen finden, z.B. Abhängigkeitsprobleme. Diese können dann wahrscheinlich mit **apt-get -f install** behoben werden.

Kapitel 4. Dienste

4.1. Grundlegende Dienste

4.1.1. DNS-Server

Der Domain-Name-Service (dns) ist für die Umwandlung von Internetadressen (z.B. `www.debian.org`) in numerische IP-Adressen (`198.186.203.20`) zuständig. Auch die umgekehrte Richtung ist möglich. Ohne diesen Dienst müsste man immer die IP-Adresse des Server kennen, auf den man z.B. beim Surfen zugreifen will. Zusätzlich zur IP-Adresse kann der dns-Server auch den Mailserver, der für eine Domain zuständig ist, speichern (sog. MX-Record). Weitere Infos wie Hardware-Ausstattung sind auch möglich, aber eher unüblich. Jede Domain ist in mindestens zwei dns-Servern eingetragen. Welche das sind, kann mit dem Programm **whois** herausgefunden werden.

Der gebräuchlichste dns-Server für Linux heißt `bind`. Mit **apt-get install bind9** wird Version 9 installiert. Der Server selber wird mit der Datei `/etc/bind/named.conf` konfiguriert. Im *options*-Abschnitt müssen folgende Zeilen hinzugefügt werden:

```
options {
    forwarders {
        217.5.100.129;
        194.25.2.129;
    };

    allow-query {
        127.0.0.0/8;
        192.168.0.0/24;
    };

    allow-transfer {
        none;
    };

    listen-on {
        127.0.0.1;
        192.168.0.1;
    };
};
```

Damit ist der dns-Server nur vom internen Netzwerk erreichbar. Schließlich ist er auch nur für das interne Netzwerk zuständig. Am Ende von `/etc/bind/named.conf` müssen noch folgende Zeilen hinzugefügt werden:

```
zone "gm.rt.schule-bw.de" IN {
    type master;
    file "gm.rt.schule-bw.de-zone";
};

zone "0.168.192.in-addr.arpa" IN {
    type master;
```

```
file "192.168.0-zone";
};
```

Jetzt müssen noch die sogenannten Zonen-Dateien erstellt werden. In diesen Dateien liegen die eigentlichen Informationen, die der dns-Server zur Umdanglung von Domains in IP-Adressen benötigt. Zuerst erstellen wir die Datei `/var/cache/bin/gm.rt.schule-bw.de-zone`:

```
;
; /var/cache/bind/gm.rt.schule-bw.de-zone
;
$TTL 2D
@      IN      SOA      dns      root.gm.rt.schule-bw.de. (
        2003081201      ; serial JJJJMMTTNN
        8H              ; refresh rate
        2H              ; retry rate
        1W              ; expiration date
        2D )            ; minimum ttl

        IN      NS      dns
        IN      MX      10 mail

dns     IN      A        192.168.0.1
mail    IN      A        192.168.0.1
pop     IN      A        192.168.0.1
news    IN      A        192.168.0.1
server  IN      A        192.168.0.1
proxy   IN      A        192.168.0.1

www     IN      NS      dnsl.belwue.de.
        ; www.gm.rt.schule-bw.de. wird von unserem
        ; dns-server nicht verwaltet, aber von belwue.
```

Diese Datei sorgt für die Auflösung eines Namens in seine IP-Adresse. Die andere Zonen-Datei ist für die umgekehrte Richtung da. `/var/cache/bin/192.168.0-zone` muss folgenden Inhalt haben:

```
;
; /var/cache/bind/192.168.0-zone
;
$TTL 2D
@      IN      SOA      dns.gm.rt.schule-bw.de. root.gm.rt.schule-bw.de. (
        2003081201      ; serial JJJJMMTTNN
        8H              ; refresh rate
        2H              ; retry rate
        1W              ; expiration date
        2D )            ; minimum ttl

        IN      NS      dns.gm.rt.schule-bw.de.

1      IN      PTR      server.gm.rt.schule-bw.de.
```

Mit `/etc/init.d/bind reload` wird die neue Konfiguration übernommen.

Damit unser Server seinen eigenen dns-Server auch benutzt, muss noch die Datei `/etc/resolv.conf` bearbeitet werden. Sie muss folgenden Inhalt haben:

```
#
# /etc/resolv.conf
```

```
#
search gm.rt.schule-bw.de
nameserver 192.168.0.1
```

Damit ist die Konfiguration des dns-Servers abgeschlossen.

4.1.2. dhcp-Server

Mit dem dhcp-Protokoll lassen sich die Clients automatisch konfigurieren. Da unser Netz die Adresse 192.168.0.0 hat und 192.168.0.1 und 192.168.0.254 schon belegt sind (Server und Netzdrucker), bleibt für die Clients im Prinzip der Adressbereich 2-253 übrig. Um aber für besondere Fälle (z.B. weitere Netzdrucker) noch einige Adressen frei zu haben, sollten die Clients nur folgenden Bereich nutzen: 140-253.

Zuerst muss der dhcp-Server wahrscheinlich installiert werden. Das geht mit dem Befehl **apt-get install dhcp**. Die Konfigurationsdatei heißt `/etc/dhcpd.conf`. Sie muss folgenden Inhalt haben:

```
#
# /etc/dhcpd.conf
#
default-lease-time 604800; # 7 days
max-lease-time 2592000; # 30 days

option domain-name "gm.rt.schule-bw.de";
option domain-name-servers dns.gm.rt.schule-bw.de;
option lpr-servers server.gm.rt.schule-bw.de;
option netbios-name-servers server.gm.rt.schule-bw.de;
option time-servers server.gm.rt.schule-bw.de;
option smtp-server mail.gm.rt.schule-bw.de;
option pop-server pop.gm.rt.schule-bw.de;
option www-server server.gm.rt.schule-bw.de;

option subnet-mask 255.255.255.0;
option broadcast-address 192.168.0.255;

subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.140 192.168.0.253;
}
```

In der Datei `/etc/default/dhcp` steht, auf welchen Netzwerkkarten der dhcp-Server aktiv ist. Hier muss folgende Zeile stehen:

```
INTERFACES="eth0"
```

Mit `/etc/init.d/dhcp restart` wird die Konfiguration übernommen.

4.1.3. Mail Transfer Agent (SMTP/UUCP)

In diesem Abschnitt wird die Konfiguration des Mailsystems beschrieben. Zunächst muss uucp eingerichtet werden. uucp wird zum Abholen der Mails vom Provider verwendet (in unserem Fall: belwue). Dann muss der eigentliche MTA (Mail Transfer Agent) eingerichtet werden, der SMTP-Server. Dieser Server erhält über uucp

die Mails der Domain gm.rt.schule-bw.de und verteilt diese an die Benutzer. Andererseits nimmt der SMTP-Server neue Mails von den Clients an und versendet sie über uucp.

4.1.3.1. uucp einrichten

Zunächst einmal muss die uucp-Software installiert werden: **apt-get install uucp**. Alle Konfigurationsdateien liegen in `/etc/uucp/`. Am besten werden in diesem Verzeichnis zuerst alle Konfigurationsdateien gelöscht.

Wir bearbeiten zuerst die Datei `call`. Hier werden drei Informationen benötigt: Der Systemname des Providers (hier: belwue), Login-Name und Passwort.

```
#
# /etc/uucp/call
#
# Format:
# system login passwd

belwue loginname password
```

"loginname" und "password" müssen natürlich noch ersetzt werden.

Die nächste Datei heißt `config`. Hier steht der UUCP-Name des lokalen Rechners.

```
#
# /etc/uucp/config
#
nodename loginname
```

Auch hier muss "loginname" ersetzt werden.

In der Datei `sys` werden die bekannten Systeme beschrieben, die verwendet werden sollen. Diese Datei sollte folgenden Inhalt haben:

```
#
# /etc/uucp/sys
#

###
# Globale Einstellung für alle Systeme

# Loginnamen und Passwort aus der Datei 'call' lesen
call-login      *
call-password   *

# Keine Einschränkung der Zugriffszeit
time            any

###
# Systemspezifische Einstellungen

system belwue
    called-login loginname
    commands     rnews rmail rsmtp rcsntp crsntp
```

```

myname      loginname
port type   tcp
address     news.belwue.de

```

Hier muss wieder "loginname" ersetzt werden.

In der Datei `/etc/hosts` muss noch eine Zeile ergänzt werden:

```
129.143.4.4 news.belwue.de belwue
```

Neue Mails werden dann letztendlich mit folgendem Befehl abgeholt: **uucico -r1 -sbelwue -f**.

Die Konfiguration kann natürlich auch getestet werden. Dazu die folgenden Befehle eingeben; die Ausgabe sollte ähnlich sein.

```

server:~# su - uucp
uucp@server:~$ /usr/lib/uucp/uuchk
Local node name loginname
Spool directory /var/spool/uucp
Public directory /var/spool/uucppublic
Lock directory /var/lock
Log file /var/log/uucp/Log
Statistics file /var/log/uucp/Stats
Debug file /var/log/uucp/Debug
Global debugging level
uucico -l will strip login names and passwords
uucico will strip UUCP protocol commands
Start uuxqt once per uucico invocation

System: belwue
Caller must log in as loginname
Will use loginname as name of local system
Call out using a specially defined port
The port is defined as:
  Port name system belwue port
  Port type tcp
  TCP service uucp
  Characteristics: eight-bit-clean reliable end-to-end fullduplex
Remote address news.belwue.de
Chat script "" \r\c ogin:-BREAK-ogin:-BREAK-ogin: \L word: \P
Chat script timeout 10
Chat script incoming bytes stripped to seven bits
Login name loginname
Password passwort
At any time may call if any work
May retry the call up to 26 times
May make local requests when calling
May make local requests when called
May send by local request: /
May send by remote request: ~
May accept by local request: ~
May receive by remote request: ~
May execute rnews rmail rsmtp rcsmtpt crsmtpt

```

```

Execution path /bin /usr/bin /usr/local/bin /usr/sbin
Will leave 50000 bytes available
Public directory is /var/spool/uucppublic
Will use any known protocol

```

Eine weitere Möglichkeit, die Konfiguration zu testen, ist, die Mails abzuholen und dabei Debug-Ausgaben erzeugen zu lassen. Das geht mit **uucico -r1 -x9 -sbelwue -f** recht einfach. In `/var/log/uucp/Debug` stehen die ausführlichen Debug-Ausgaben. Die Datei `/var/log/uucp/Log` wird immer erweitert, wenn uucp benutzt wird.

4.1.3.2. exim (MTA) einrichten

Exim ist wahrscheinlich schon installiert. Mit **eximconfig** wird das Konfigurationsprogramm aufgerufen. Bei der ersten Frage muss Option 1 ausgewählt werden (Internet site). Die Antwort auf die nächste Frage ("visible mail name of your system") lautet *gm.rt.schule-bw.de*. Dann noch vier mal "none" also Antwort verwenden und die grobe Konfiguration ist abgeschlossen.

Jetzt kommt die feinere Konfiguration in der Datei `/etc/exim/exim.conf`. Um uucp mit exim verwenden zu können, müssen im Abschnitt *Transports* folgende Zeilen hinzugefügt werden:

```

#####
#                                TRANSPORTS CONFIGURATION                                #
#####
#                                ORDER DOES NOT MATTER                                #
#      Only one appropriate transport is called for each delivery.      #
#####
uucp_pipe:
    driver = pipe
    command = "uux - -a$sender_address -r $host\!rmail ($local_part@$domain)"
    pipe_as_creator
    restrict_to_path
    path = "/usr/bin:/bin"
    return_output

```

Im Abschnitt *Router* müssen folgende Zeilen *am Anfang* des Abschnitts hinzugefügt werden (die Reihenfolge spielt eine Rolle):

```

#####
#                                ROUTERS CONFIGURATION                                #
#      Specifies how remote addresses are handled                                #
#####
#                                ORDER DOES MATTER                                #
#      A remote address is passed to each in turn until it is accepted.  #
#####
uucp:
    driver = domainlist
    transport = uucp_pipe
    route_list = "* belwue byname"

```

Damit wird die ausgehende Mail zuerst über uucp verschickt.

Im ersten Abschnitt (main configuration settings) muss die Variabel "trusted_users" gesucht werden. Folgendes muss geändert werden:


```
trusted_users = mail:uucp
trusted_groups = uucp
```

Im selben Abschnitt müssen noch zwei Optionen gesucht und geändert werden:

```
host_accept_relay = 127.0.0.1 : 192.168.0.0/24
never_users = root
```

Damit können alle Computer im Netzwerk über den Server als SMTP-Server Nachrichten versenden mit beliebigen Absender und Empfänger. Die zweite Option verhindert, dass dem Benutzer root (also dem Systemadministrator) direkt Mails zugestellt werden. Die Mails an root müssen an einen anderen Benutzer weitergeleitet werden. Das passiert im nächsten Absatz.

Zur Mailkonfiguration gibt es noch eine zweite, sehr wichtige Datei: `/etc/aliases`. Hier können z.B. Weiterleitungen definiert werden. **eximconfig** hat schon eine neue Datei erstellt. Am Ende dieser Datei sollten folgende Zeilen hinzugefügt werden:

```
info:                verwaltung, homepage
homepage:            thomas, alexander, jmueller, bohn, andreas
verwaltung:          root
sekretariat:         poststelle@xxxxxxxxx.schule.bwl.de, \sekretariat
proxymaster:         root
root:                andreas
```

```
# Anmerkung: Hier könnten noch Weiterleitungen von Mail an externe Adressen
# eingerichtet werden, zum Beispiel für Lehrer
```

```
andreas: a.dangel@gmx.de, \andreas
```

Die letzte Zeile ist ein Beispiel für eine externe Weiterleitung.

4.1.3.3. Regelmäßig Mails abholen

Soll der Server regelmäßig neue Mails abholen, muss der cron-Dienst entsprechend konfiguriert werden. In der Datei `/etc/cron.d/getEmail` steht eine neue cron-Konfiguration. Sie muss folgenden Inhalt haben:

```
#
# /etc/cron.d/getEmail
#

# min hour(s)      day month dow  user  command
0 7,10,13,15,17,19,21 * * * * root  /usr/local/sbin/getEmail
```

Die Datei `/usr/local/sbin/getEmail` sieht so aus:

```
#!/bin/sh
#
# /usr/local/sbin/getEmail
#

# Internetverbindung aufbauen: ping an news.belwue.de
ping -c 5 129.143.4.4 > /dev/null

# Mailqueue abarbeiten
/usr/sbin/exim -q
```

```
# neue Mails holen
/usr/sbin/uucico -rl -sbelwue -f
```

Mit **chmod 700 /usr/local/sbin/getEmail** wird das Script ausführbar gemacht.

Standardmäßig wird die Mail-Warteschlange von exim alle 15 Minuten ausgeführt. Dies ist unnötig und verursacht nur häufige Internet-Einwahlen. Die Mail-Queue wird im Script `/usr/local/sbin/getEmail` auch ausgeführt (**exim -q**). Wir deaktivieren deshalb die Standardeinstellung. Dazu wird der Eintrag in der vierten Zeile in `/etc/cron.d/exim` mit einem `"#"` auskommentiert.

4.2. Fernwartung

4.2.1. SSH-Server

Wenn ein ssh-Server installiert ist, kann man sich über das Netzwerk auf den Server einloggen. Dabei ersetzt ssh den alten telnet-Dienst; ssh verwendet im Gegensatz zu telnet eine Verschlüsselung. ssh-Clients gibt es für Linux und für Windows. Ein guter Client für Windows heißt "putty".

Mit **apt-get install ssh** wird der Server installiert. In `/etc/ssh/` liegen die Konfigurationsdateien. Die Datei `sshd_config` ist für den ssh-Server zuständig; die wichtigste Option lautet *PermitRootLogin*; sie sollte auf "no" gestellt sein. Damit kann sich zwar root nicht mehr direkt einloggen (was erstens ein Sicherheitsrisiko ist und zweitens nicht nötig ist), aber als normaler Benutzer wird man mit **su** zum root (natürlich nach Eingabe des root-Passwortes).

Um weiter an Sicherheit zu gewinnen (der ssh-Zugang ist, wenn der Server online ist, im ganzen Internet verfügbar...), sollte noch der Standard-Port geändert werden. Das geschieht in der Datei `/etc/ssh/sshd_config` mit der Option *Port*. Der Standard-Port ist 22, wir verwenden aber ab jetzt den Port 1222.

Die Konfiguration wird mit **/etc/init.d/ssh reload** übernommen. Mit **ssh -p 1222 user@server** kann man sich nun über das Netzwerk auf den Server einloggen.

4.2.2. dyndns

Bei jeder Einwahl über den T-DSL-Anschluss von T-Online bekommt der Server eine andere IP-Adresse, über die er im Internet erreichbar ist. Damit aber trotzdem Fernwartung über das Internet möglich ist, muss die IP-Adresse bekannt sein. Der kostenlose dyndns-Dienst stellt einen DNS-Eintrag zur Verfügung, der bei jeder Einwahl aktualisiert wird. Somit ist der Server dann immer über den gleichen Hostnamen erreichbar.

Mit **apt-get install ddclient** wird die dazu notwendige Software installiert. Die Konfigurationsdatei heißt `/etc/ddclient.conf` und sollte folgenden Inhalt haben:

```
#
# /etc/ddclient.conf
#
pid=/var/run/ddclient.pid
protocol=dyndns2
wildcard=yes
use=if, if=ppp0
server=members.dyndns.org
login=loginname
password=password
host.dyndns.org
```

"loginname", "password" und "host.dyndns.org" müssen natürlich noch angepasst werden.

Damit die IP-Adresse bei jeder Einwahl auch aktualisiert wird, muss noch `/etc/default/ddclient` angepasst werden. Die zwei entscheidenden Einträge lauten:

```
run_ipup="true"
run_daemon="false"
```

Jetzt muss noch dafür gesorgt werden, dass der Server regelmäßig online geht. Zum Beispiel kann der Server regelmäßig neue Mails herunterladen. Wie das einzurichten ist, steht in Abschnitt 4.1.3.3.

4.3. Dienste für Clients

4.3.1. Webserver

Der Webserver ist quasi die komfortable Bedienoberfläche für das Schulnetz. Über diesen Server werden einzelne CGI-Skripte ausgeführt, mit denen man zum Beispiel einen neuen Benutzer anlegen kann. Ansonsten lässt sich die Homepage der Schule anschauen und jeder Benutzer kann im Ordner `public_html/` in seinem Homeverzeichnis seine eigenen Seiten im Schulnetz veröffentlichen.

Als Webserver wird `apache` verwendet. Falls `apache` noch nicht installiert ist, lässt sich dies mit **`apt-get install apache`** nachholen. Die Konfigurationsdatei heißt `/etc/apache/httpd.conf`. Diese muss an einigen Stellen angepasst werden. Der folgende Ausschnitt aus der Konfigurationsdatei soll einen Anhaltspunkt geben, was gegenüber der Standardkonfigurationsdatei von Debian verändert werden muss:

```
#
# /etc/apache/httpd.conf
#

### Section 1: Global Environment
# Keine Änderungen nötig

### Section 2: 'Main' server configuration
ServerAdmin webmaster@gm.rt.schule-bw.de
ServerName server.gm.rt.schule-bw.de
DocumentRoot "/usr/local/httpd/htdocs"

<Directory />
```

```

    AuthUserFile      /etc/apache/passwd
    AuthGroupFile     /etc/apache/group
    Order allow,deny
    Options None
    AllowOverride None
</Directory>

#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory /usr/local/httpd/htdocs/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

# Einzelne Verzeichnisse
<Directory /usr/local/httpd/htdocs/admin/>
    AllowOverride None
    Options ExecCGI FollowSymLinks Indexes
    DirectoryIndex index.html index.cgi

    AuthType Basic
    AuthName "Systemverwaltung"
    AuthUserFile /etc/apache/passwd
    AuthGroupFile /etc/apache/group
    require group admin
</Directory>

<Directory /usr/local/httpd/htdocs/internet/>
    AllowOverride None
    Options ExecCGI FollowSymLinks
    DirectoryIndex index.html index.cgi

    AuthType Basic
    AuthName "Internet-Einwahl"
    AuthUserFile /etc/apache/passwd
    AuthGroupFile /etc/apache/group
    require group internet
</Directory>

<Directory /usr/local/httpd/htdocs/organisation/>
    AllowOverride None
    Options Indexes
    DirectoryIndex index.html index.htm

    AuthType Basic
    AuthName "Organisation"
    AuthUserFile /etc/apache/passwd
    AuthGroupFile /etc/apache/group
    require group lehrer
</Directory>

<Directory /usr/local/httpd/htdocs/passwd/>
    AllowOverride None
    Options ExecCGI FollowSymLinks

```

```

    DirectoryIndex index.html index.cgi

    AuthType Basic
    AuthName "Passwortaenderung"
    AuthUserFile /etc/apache/passwd
    AuthGroupFile /etc/apache/group
    require group users
</Directory>

<Directory /usr/local/httpd/htdocs/mail-config/>
    AllowOverride None
    Options ExecCGI FollowSymLinks
    DirectoryIndex index.html index.cgi

    AuthType Basic
    AuthName "Mail-Weiterleitung"
    AuthUserFile /etc/apache/passwd
    AuthGroupFile /etc/apache/group
    require group users
</Directory>

<Directory /usr/local/samba/homepage/>
    AllowOverride None
    Options Indexes FollowSymLinks
    Order allow,deny
    Allow from all
</Directory>

ScriptAlias /cgi-bin/ /usr/local/httpd/cgi-bin/

<Directory /usr/lib/cgi-bin/>
    AllowOverride None
    Options ExecCGI
    Order allow,deny
    Allow from all
</Directory>

<IfModule mod_mime.c>
    AddHandler cgi-script .cgi .sh .pl
</IfModule>

<IfModule mod_userdir.c>
    UserDir /home/public_html
</IfModule>

<Directory /home/public_html/*/>
    AllowOverride FileInfo AuthConfig Limit
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    <Limit GET POST OPTIONS PROPFIND>
        Order allow,deny
        Allow from all
    </Limit>
    <Limit PUT DELETE PATCH PROPPATCH MKCOL COPY MOVE LOCK UNLOCK>
        Order deny,allow
        Deny from all
    </Limit>
</Directory>

```

Es wurden für einige Verzeichnisse die Zugriffsrechte gesetzt. Oft muss man sich erst anmelden (AuthType Basic), wenn man so ein Verzeichnis betreten will.

Mit **/etc/init.d/apache restart** wird die neue Konfiguration übernommen. Jetzt wird noch ein Script benötigt, das die Dateien `/etc/apache/passwd` und `/etc/apache/group` erzeugt. Diese sind für eine Anmeldung notwendig. Das Script mit dem Dateinamen `/usr/local/sbin/generate_apache_auth` hat folgenden Inhalt:

```
#!/usr/bin/perl -w
#
# /usr/local/sbin/generate_apache_auth
#
# WARNING: will run as root!!!
# Copyright (C) 1999 Thomas Bleher <thomas@gm.rt.schule-bw.de>
# under the GNU GPL
#

umask(0027);
%valid_users = map {$_ => 1} split / /, (getgrnam('users'))[3];

open(DATEI, '/etc/shadow');
@passwords = <DATEI>;
close DATEI;

open DATEI, '>/etc/apache/passwd';
foreach (@passwords) {
    @user = split(/:/, $_);
    if (defined($valid_users{$user[0]})) { ## nur normale user (Sicherheit)
        print DATEI "$user[0]:$user[1]\n";
    }
}
close DATEI;

open DATEI, '/etc/group';
@groups = <DATEI>;
close DATEI;

open DATEI, '>/etc/apache/group';
foreach (@groups) {
    @user = split(/:/, $_);
    $user[3] =~ s/,/ /g;
    $user[3] =~ s/root//g;
    print DATEI "$user[0]: $user[3]";
}
close DATEI;

$suid = 0; # root
$gid = 33; # www-data
chown $suid, $gid, '/etc/apache/passwd', '/etc/apache/group';
```

Dieses Script wird mit **chmod 700 /usr/local/sbin/generate_apache_auth** ausführbar gemacht. Jedesmal, wenn ein neuer Benutzer angelegt wird, wenn ein Benutzer gelöscht wird oder wenn ein Passwort geändert wurde, muss dieses Script ausgeführt werden.

4.3.2. pop3-Server

Über einen POP3-Server können sich die Benutzer ihre Mail komfortabel über einen Mailclient (z.B. Mozilla-Mail, Outlook Express) abholen. Der POP3-Server verwendet die Mailboxen der lokalen Benutzer auf dem Server. Mit **apt-get install qpopper** wird ein einfacher POP3-Server installiert. Die Standardkonfiguration genügt; dieser Server muss nicht konfiguriert werden. Falls doch: Die Konfigurationsdatei heißt `/etc/qpopper.conf`.

4.3.3. Proxy und Filter

4.3.3.1. squid

Über den Proxy-Server squid erhalten die Clients Zugang zum Internet. Alle Rechner im Schulnetz greifen also nicht direkt auf das Internet zu, sondern nur auf den Proxy-Server, der dann die gewünschten Daten liefert. Zuerst muss die Software installiert werden: **apt-get install squid**. Die Konfigurationsdatei heißt `/etc/squid.conf`. In der Datei müssen einige Einstellungen angepasst werden. Hier sind die wichtigen Einstellungen:

```
#
# /etc/squid.conf
#

ftp_user webmaster@gm.rt.schule-bw.de

# ACCESS CONTROLS
# -----
[...]

# INSERT YOUR OWN RULE(S) HERE...
acl schulnetz src 192.168.0.0/255.255.255.0
acl server dst 192.168.0.1
http_access allow schulnetz

append_domain .gm.rt.schule-bw.de

always_direct allow server
```

Mit `/etc/init.d/squid restart` wird die Konfiguration übernommen.

4.3.3.2. squidGuard

Der Filter, der als Jugendschutzfilter eingerichtet werden kann, sperrt bestimmte Internetadressen. Aktuelle Listen erhält man unter www.squidguard.org/blacklist/ (<http://www.squidguard.org/blacklist/>).

Die Filtersoftware heißt squidguard und wird mit **apt-get install squidguard** installiert. Damit der Filter im Proxy integriert wird, muss `/etc/squid.conf` bearbeitet werden. Es müssen folgende zwei Einträge existieren:

```
#
# /etc/squid.conf
#
[...]
```

```
redirect_program /usr/bin/squidGuard
redirect_children 5
```

Dann müssen neue Blacklisten unter der oben genannten Adresse heruntergeladen werden. Mit **cd /var/lib/squidguard/db; tar xfvz ~/blacklists.tar.gz** wird das Archiv ins richtige Verzeichnis entpackt. Danach muss noch **chmod -R 777 /var/lib/squidguard/db/blacklists** ausgeführt werden. Die Konfigurationsdatei heißt `/etc/squid/squidGuard.conf`. Sie muss folgenden Inhalt haben:

```
#
# /etc/squid/squidGuard.conf
#
logdir /var/log/squid
dbhome /var/lib/squidguard/db

src localnet {
    ip 127.0.0.0/8
    ip 192.168.0.0/24
}

dest ads {
    domainlist    blacklists/ads/domains
    urllist       blacklists/ads/urls
}
dest aggressive {
    domainlist    blacklists/aggressive/domains
    urllist       blacklists/aggressive/urls
}
dest audio-video {
    domainlist    blacklists/audio-video/domains
    urllist       blacklists/audio-video/urls
}
dest drugs {
    domainlist    blacklists/drugs/domains
    urllist       blacklists/drugs/urls
}
dest gambling {
    domainlist    blacklists/gambling/domains
    urllist       blacklists/gambling/urls
}
dest hacking {
    domainlist    blacklists/hacking/domains
    urllist       blacklists/hacking/urls
}
dest mail {
    domainlist    blacklists/mail/domains
}
dest porn {
    domainlist    blacklists/porn/domains
    urllist       blacklists/porn/urls
    #expressionlist blacklists/porn/expressions
}
dest proxy {
    domainlist    blacklists/proxy/domains
    urllist       blacklists/proxy/urls
}
dest violence {
```



```

    domainlist      blacklists/violence/domains
    urllist         blacklists/violence/urls
    #expressionlist blacklists/violence/expressions
}
dest warez {
    domainlist      blacklists/warez/domains
    urllist         blacklists/warez/urls
}

acl {
    localnet {
        pass !aggressive !porn !violence all
        redirect http://server/cgi-bin/blocked.cgi?clientaddr=%a&url=%u
    }

    default {
        pass none
        redirect http://server/cgi-bin/blocked.cgi?clientaddr=%a&url=%u
    }
}

```

Damit werden die Adresslisten "aggressive", "porn" und "violence" abgeblockt. Wer eine solche Seite aufruft, bekommt das Script `blocked.cgi` zu sehen. Dieses Script gibt eine Fehlermeldung aus oder, wenn der Filter eine Bilddatei blockiert (z.B. Werbung), dann wird `/usr/local/httpd/htdocs/blocked.gif` angezeigt. Das Script `blocked.cgi` muss noch im Verzeichnis `/usr/local/httpd/cgi-bin/` erstellt werden:

```

#!/usr/bin/perl
#
# /usr/local/httpd/cgi-bin/blocked.cgi
#
use CGI;
$q = new CGI;
$clientaddr = "";
$url="";

$clientaddr = $q->param('clientaddr') if defined($q->param('clientaddr'));
$url = $q->param('url') if defined($q->param('url'));

if ($url =~ /\.(gif|jpg|jpeg|mpg|mpeg|avi|mov)$/i) {
    print "Content-type: image/gif\n";
    ($sec,$min,$hour,$mday,$mon,$year,$wday,$yday,$isdst) = gmtime($time);
    printf "Expires: %s, %02d-%s-%02d %02d:%02d:%02d GMT\n\n", $day[$wday],
        $mday,$month[$mon],$year,$hour,$min,$sec;
    open(GIF, "$ENV{'DOCUMENT_ROOT'}/blocked.gif");
    while (<GIF>) {
        print;
    }
    close(GIF);
} else {
    print "Content-type: text/html\n";
    printf "Expires: %s, %02d-%s-%02d %02d:%02d:%02d GMT\n\n", $day[$wday],
        $mday,$month[$mon],$year,$hour,$min,$sec;

    print <<EOF;
<html>
<head>
    <title>403 Forbidden - Zugriff verweigert</title>
</head>

```

```

<body>
<center>
<h1>403 Forbidden - Zugriff verweigert</h1>
Der Zugriff wurde Ihnen aus einem der folgenden Gründe verweigert:
<table border="0">
<tr><td>
<ul>
<li>Ihre IP-Adresse ist nicht aus dem Bereich des Schulnetzes</li>
<li>Der Jugendschutzfilter ist aktiviert. Die Filterkategorien sind:
<ul>
<li>aggressive</li>
<li>porn</li>
<li>violence</li>
</ul>
</li>
</ul>
</td></tr>
</table>
Wenn diese Seite zu Unrecht gesperrt wurde, schreiben Sie eine Mail an<br>
<a href="mailto:proxymaster@gm.rt.schule-bw.de">
proxymaster@gm.rt.schule-bw.de</a>
</center>
</body>
</html>
EOF
}
exit 0;

```

Mit **chmod 755 /usr/local/httpd/cgi-bin/blocked.cgi** wird das Script ausführbar gemacht. Damit squidGuard später schneller startet, wird **squidGuard -C all** einmal ausgeführt. Dabei werden die Adresslisten in ein Datenbankformat konvertiert. Mit **/etc/init.d/squid restart** wird der Filter letztendlich aktiviert.

4.3.4. Drucker

Damit der Netzwerkdrucker später über Samba den Windows-Clients zur Verfügung gestellt werden kann, muss der Netzwerkdrucker (der übrigens auch lokal am Druckerport des Servers angeschlossen sein könnte) zunächst auf dem Server eingerichtet werden. Dazu wird das "Common UNIX Printing System" benötigt, das mit **apt-get install cupsys cupsys-client** noch nachinstalliert werden muss. Konfiguriert wird das Druck-System über einen Webbrowser. Als Text-Browser eignet sich lynx. Möglicherweise muss dieser mit **apt-get install lynx** nachinstalliert werden. Mit **/etc/init.d/cupsys start** wird das Druck-System gestartet.

Mit **lynx http://localhost:631/** wird der Webbrowser zur Konfiguration gestartet. Hier muss zunächst *Manage Printers* ausgewählt werden, danach *Add Printer*. Der Browser fragt nun nach einem Benutzernamen. Einfach "root" und das root-Passwort verwenden. Auf der nächsten Seite muss der Freigabe-Name eingegeben werden, also z.B. "netzdrucker". *Location* und *Description* sind weniger wichtig. Mit *Continue* geht's weiter. Auf der folgenden Seite muss jetzt als *Device* "AppSocket/HP JetDirect" ausgewählt werden. Im Feld *Device URI* auf der nächsten Seite muss "socket://192.168.0.254:9100" eingetragen werden. *Model/Driver* ist "Raw" und "Raw Queue (en)". Das bedeutet, dass die Druckdaten genau so an den Drucker weitergeleitet werden, wie der Windows-Client sie verschickt. Unter Windows muss also noch ein passender Druckertreiber installiert werden. Das war's soweit. Der Rest muss unter Samba konfiguriert werden.

4.3.5. Samba

Samba stellt den Windows-Clients Speicherplatz zur Verfügung: Ein Homeverzeichnis, ein Verzeichnis für temporäre Daten, ein öffentliches Verzeichnis und ein Verzeichnis für Windows-Programme. Für bestimmte Benutzergruppen gibt es weitere Verzeichnisse: Homepage und Admin beispielsweise. Außerdem gibt es noch ein sogenannten "netlogon-script", das beim Anmelden bei Samba auf dem Client gestartet wird und die Verzeichnisse automatisch einbindet. Zusätzlich gibt Samba noch den Drucker frei, der im vorigen Abschnitt eingerichtet wurde.

Zunächst muss samba installiert werden; das geht mit **apt-get install samba**. Die Konfigurationsdatei heißt `/etc/samba/smb.conf`. Sie hat folgenden Inhalt:

```
#
# /etc/samba/smb.conf
#
[global]
    workgroup = WORKGROUP
    netbios name = server
    server string = Samba Server
    comment = Samba Server

    hosts allow = 192.168.0.

    load printers = yes
    printcap name = cups
    printing = CUPS

    guest account = nobody

    #log file = /var/log/samba/log.%m # für jede Maschine eine eigene Logfile
    log file = /var/log/samba/log.smb
    log level = 1
    max log size = 1000 # in KB
    syslog = 0

    security = user

    encrypt passwords = yes
    smb passwd file = /usr/local/samba/private/smbpasswd
    passwd program = /usr/bin/passwd %u
    unix password sync = yes

    socket options = TCP_NODELAY
    keepalive = 30

    interfaces = eth0

    local master = yes
    os level = 65
    domain master = yes
    preferred master = yes
    domain logons = yes

    logon script = logon.bat
    # für Roaming Profiles (WinNT)
```

```

#logon path = \\SERVER%\%U\profile
# für Roaming Profiles (Win9X)
#logon home = \\SERVER%\%U\profile

wins support = yes

# alles in eine Zeile!
invalid users = root daemon bin sys sync games man lp mail news uucp proxy
    postgres www-data backup operator list irc gnats identd sshd gdm telnetd
    ftp nut faxmaster partimag mysql homepage internet
valid users = +users nobody
admin users = +admin

[printers]
    comment = All Printers
    path = /var/spool/samba
    browseable = no
    public = yes
    guest ok = yes
    writeable = no
    printable = yes

[netlogon]
    path = /usr/local/samba/netlogon
    write list = +admin
    guest ok = yes

[homes]
    comment = Heimatverzeichnis von %U
    # wegen public_html: www-data
    force group = www-data
    read only = no
    create mask = 0755
    browseable = no

[tmp]
    comment = Temporaere Dateien
    path = /usr/local/samba/temp
    force group = users
    read only = no
    create mask = 0664
    directory mask = 0775
    guest ok = yes

[pub]
    comment = Oeffentliches Verzeichnis
    path = /usr/local/samba/pub
    write list = +lehrer
    force group = lehrer
    create mask = 0664
    force create mode = 0664
    directory mask = 0775
    force directory mode = 0775
    guest ok = yes

[homepage]

```

```

comment = Gymnasium-Muensingen Homepage
path = /usr/local/samba/homepage
write list = +homepage
force group = homepage
create mask = 0664
force create mode = 0664
directory mask = 0775
force directory mode = 0775

[programme]
comment = Programme fuer Windows
path = /usr/local/samba/programme
# jeder darf schreiben (Windows-Programme brauchen das ;-)
write list = +users nobody
force group = admin
create mask = 0664
force create mode = 0664
directory mask = 0775
force directory mode = 0775
guest ok = yes

[admin]
comment = Admin-Verzeichnis
path = /usr/local/samba/admin
valid users = +admin
force group = admin
read only = no
create mask = 0660
force create mode = 0660
directory mask = 0770
force directory mode = 0770

```

Die angegebenen Verzeichnisse müssen natürlich existieren (alles unter `/usr/local/samba/`). Außerdem müssen für diese Verzeichnisse die passenden Zugriffsrechte gesetzt sein. Mit **`/etc/init.d/samba restart`** werden die Einstellungen übernommen.

Für die Druckfunktion muss das Verzeichnis `/var/spool/samba` existieren. Außerdem muss es den Mitglieder der Gruppe "users" erlaubt sein, in dieses Verzeichnis zu schreiben (sonst dürfen diese Benutzer nicht drucken). Dies ist zu erreichen mit den folgenden zwei Befehlen:

```

chgrp users /var/spool/samba
chmod g+wx /var/spool/samba

```

Hier ist das Netlogon-Script. Es verbindet einige Netzlaufwerke, sodass diese im Explorer zur Verfügung stehen. Außerdem wird die Uhrzeit des Clients nach der Server-Uhr gestellt. Damit es keine Probleme mit Mozilla-Profilen gibt, wird eine Datei bei jedem Einloggen auf den Client kopiert; in dieser Datei steht, dass das Mozilla-Profil unter "H:\Mozilla" zu finden ist.

```

@echo off
net use h: \\arktur\homes /yes
net use i: \\arktur\homepage /yes
net use v: \\arktur\pub /yes
net use t: \\arktur\tmp /yes

```

```
net use p: \\arktur\progs /yes
net time \\arktur /set /yes

rem Mozilla-Profil-Konfiguration:
copy \\arktur\netlogon\registry.dat C:\WINDOWS\Anwendungsdaten\Mozilla /Y
```

4.3.5.1. Logdateien

Samba benützt drei Logdateien: In `/var/log/samba/log.nmbd` protokolliert der `nmbd`-Dämon seine Zugriffe. In `/var/log/samba/log.smbd` steht, wann der `smbd`-Dämon gestartet wurde. Und in `/var/log/samba/log.smb` steht, wer auf irgendwelche Freigaben zugegriffen hat. Diese Logdateien sollen auch mit `logrotate` rotiert werden. Damit dies zuverlässig klappt, wird vor dem Rotieren der Samba-Dämon gestoppt und danach wieder gestartet.

Die Datei `/etc/logrotate.d/samba` muss folgenden Inhalt haben:

```
/var/log/samba/log.smb {
    weekly
    missingok
    rotate 7

    prerotate
        /etc/init.d/samba stop || true
        killall -q smbd || true
    endscript

    compress
    notifempty
}

/var/log/samba/log.smbd {
    weekly
    missingok
    rotate 7

    prerotate
        /etc/init.d/samba stop || true
        killall -q smbd || true
    endscript

    compress
    notifempty
}

/var/log/samba/log.nmbd {
    weekly
    missingok
    rotate 7

    prerotate
        /etc/init.d/samba stop || true
        killall -q nmbd || true
    endscript
}
```

```

endscript

postrotate
    /etc/init.d/samba start
endscript

compress
notifempty
}

```

4.3.5.2. Tägliches Neustarten

Da es offenbar öfters vorkommt, dass der Samba-Server, nachdem er ein paar Tage ununterbrochen lief, nicht mehr reagierte, wird der Samba-Server nun mittels eines Cron-Jobs täglich neugestartet. Dazu wird im Verzeichnis `/etc/cron.daily/` die Datei `samba-restart` erstellt:

```

#!/bin/sh

/etc/init.d/samba stop
sleep 4
/etc/init.d/samba start
sleep 4

# überprüfen, ob nmbd läuft
ps `cat /var/run/samba/nmbd.pid` > /dev/null
if [ $? -eq 1 ]; then
    # nmbd läuft nicht...
    /etc/init.d/samba stop
    sleep 4
    /etc/init.d/samba start
fi

```

Die Datei muss noch mit **chmod 755 /etc/cron.daily/samba-restart** ausführbar gemacht werden.

4.4. Sicherheit

Ein Computer ist umso sicherer, je weniger Dienste auf ihm laufen. Dann hat er eine geringere Angriffsfläche. Am besten ist es, wenn die nicht benötigten Dienste ganz deaktiviert werden und man sich nicht komplett auf die Firewall verlässt. Außerdem nützt die Firewall gar nichts, wenn der verwendete Server eine Sicherheitslücke hat. Daher sollte in regelmäßigen Abständen die Software upgedatet werden.

4.4.1. Firewall

Die Firewall, die auf dem Server eingerichtet werden soll, schützt den Server zum einen vor dem Internet und zum anderen vor dem lokalen Netzwerk. Immer wenn der Server online ist, ist er auch im Internet sichtbar - und

daher auch angreifbar. Die Firewall soll so konfiguriert sein, dass sie nur die unbedingt benötigten Dienste erlaubt. Alles andere wird blockiert.

Das Firewall-Script wird hier nicht abgedruckt. Es ist aber auf der CD zu finden. Zunächst einmal wird das Script nach `/usr/local/sbin/firewall` kopiert. `/etc/init.d/firewall` ist ein symbolischer Link darauf. Und `/etc/rc[2345].d/S10firewall` sind ebenfalls symbolische Links, aber relative mit `../init.d/firewall` als Ziel.

Jetzt ist das Firewall-Script fertig installiert. Beim Systemstart wird es automatisch geladen. Manuell kann dies mit **`/etc/init.d/firewall start`** nachgeholt werden. Die Firewall kann mit **`/etc/init.d/firewall stop`** komplett deaktiviert werden.

Die Firewall besteht unter Linux aus mindestens zwei Teilen: INPUT und OUTPUT. Ein ankommendes Paket kommt immer nach INPUT, ein vom Server generiertes und ihn verlassendes Paket durchläuft immer die Firewallregeln in OUTPUT. Für beide Teile kann eine Standardreaktion eingestellt werden, wenn keine Firewallregel zutrifft. Bei INPUT werden alle Pakete verworfen und bei OUTPUT wird alles erlaubt.

4.4.2. inetd

Über den speziellen Server "inetd" werden bei Bedarf weitere Server gestartet. Da viele Dienste gar nicht benötigt werden, sollten sie ganz deaktiviert werden. Die Konfigurationsdatei zu "inetd" heißt `/etc/inetd.conf`. Mit einem "#" am Anfang der Zeile werden bestehende Einträge auskommentiert. Mindestens zwei Einträge sollten aber belassen werden: der smtp-Dienst und der pop3-Dienst. Ohne diese Dienste können die Clients keine Mails mehr verschicken und empfangen. Eine minimale Konfigurationsdatei sieht daher so aus:

```
#
# /etc/inetd.conf
#
smtp          stream  tcp      nowait  mail    /usr/sbin/exim exim -bs
pop-3         stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.qpopper -f /etc/qpop
```

Mit **`/etc/init.d/inetd restart`** wird die Konfiguration übernommen.

4.4.3. netstat und nmap

netstat und *nmap* sind zwei Tools, mit denen die Sicherheit ein wenig ausgelotet werden kann. *nmap* muss gegebenenfalls mit **`apt-get install nmap`** nachinstalliert werden.

4.4.3.1. netstat

netstat zeigt aktuelle Netzwerkverbindungen und geöffnete Ports an. Dazu muss **`netstat -tuanp`** aufgerufen werden. "-t" steht dabei für das Protokoll TCP, "-u" für UDP, "-a" zeigt alle Einträge an (nicht nur bestehende Verbindungen) und "-n" zeigt alle Ports und IP-Adressen numerisch an. "-p" zeigt zusätzlich - wenn vorhanden - das Server-Programm an.

```
Active Internet connections (servers and established)
```


Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State	PID/Program name
tcp	0	0	0.0.0.0:37	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:9	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:139	0.0.0.0:*	LISTEN	213/smbd
tcp	0	0	0.0.0.0:13	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:110	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:79	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:111	0.0.0.0:*	LISTEN	113/portmap
tcp	0	0	0.0.0.0:80	0.0.0.0:*	LISTEN	240/apache
tcp	0	0	0.0.0.0:113	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:21	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:631	0.0.0.0:*	LISTEN	206/cupsd
tcp	0	0	0.0.0.0:23	0.0.0.0:*	LISTEN	201/inetd
tcp	0	0	0.0.0.0:25	0.0.0.0:*	LISTEN	201/inetd
udp	0	0	0.0.0.0:517	0.0.0.0:*		201/inetd
udp	0	0	0.0.0.0:518	0.0.0.0:*		201/inetd
udp	0	0	192.168.0.3:137	0.0.0.0:*		208/nmbd
udp	0	0	0.0.0.0:137	0.0.0.0:*		208/nmbd
udp	0	0	0.0.0.0:9	0.0.0.0:*		201/inetd
udp	0	0	192.168.0.3:138	0.0.0.0:*		208/nmbd
udp	0	0	0.0.0.0:138	0.0.0.0:*		208/nmbd
udp	0	0	0.0.0.0:111	0.0.0.0:*		113/portmap

Bei TCP steht unter State "LISTEN", das heißt, dass hier ein Dienst auf neue Verbindungen wartet.

4.4.3.2. nmap

nmap ist ein sogenannter Portscanner. Er kann zum Beispiel dazu benutzt werden, eine Firewall zu testen. *nmap* zeigt dabei dann alle offene Ports an. Am besten wird *nmap* von verschiedenen Rechnern im Intranet und Internet ausgeführt. Ein Beispielaufwurf:

```
server:~# nmap 192.168.0.1
```

```
Starting nmap V. 2.54BETA31 ( www.insecure.org/nmap/ )
Interesting ports on server.adabolo.homelinux.net (192.168.0.1):
(The 1545 ports scanned but not shown below are in state: closed)
Port      State      Service
22/tcp    open       ssh
25/tcp    open       smtp
37/tcp    open       time
53/tcp    open       domain
80/tcp    open       http
139/tcp   open       netbios-ssn
444/tcp   open       snpp
4557/tcp  open       fax
4559/tcp  open       hylafax
```

```
Nmap run completed -- 1 IP address (1 host up) scanned in 1 second
```

Auf den angezeigten Ports ist eine Verbindung möglich. Das heißt, dass dieser Dienst erreichbar ist.

4.4.4. Backup

Um die Datensicherheit des Servers zu verbessern (wenn Software-RAID eingerichtet wurde, besteht schon eine Sicherheit), soll wöchentlich ein Backup der Home-Verzeichnisse (alles unterhalb von `/home/`) erstellt werden. Dieses Backup kann dann auf ein externes Medium gesichert werden (z.B. DVD).

Zuerst muss das Script `/usr/local/sbin/userdir-backup.sh` erstellt werden:

```
#!/bin/sh

tmpfile='mktemp /tmp/backup-filelist.XXXXXX'

echo "Führe Backup der Home-Verzeichnisse aus..."
tar cv /home 2> $tmpfile | bzip2 | split -b 2047m - \
    /usr/local/samba/admin/userdir-backup/userdir-backup-`date +%Y%m%d`.tar.bz2.

if [ $? -ne 0 ]; then
    cat $tmpfile
fi

echo `cat $tmpfile | wc -l` Dateien wurden gesichert.
rm $tmpfile

# Freier Festplattenplatz ausgeben
df -h
```

Zunächst muss dieses Script noch mit **chmod 700 /usr/local/sbin/userdir-backup.sh** ausführbar gemacht werden. Wenn dieses Script ausgeführt wird, dann werden unterhalb von `/usr/local/samba/admin/userdir-backup/` (also im Verzeichnis "userdir-backup" in der Samba-Freigabe "admin") mehrere Dateien erstellt, die eine maximale Größe von 2 GB haben (2047 MB). Die Namen der Dateien sind folgendermaßen aufgebaut: `userdir-backup-%datum.tar.bz2.%endung`. Datum kann z.B. "20031019", also zuerst das Jahr, der Monat und dann der Tag. Die Endung wird hochgezählt und beginnt mit "aa". Die erste Datei hat also die Endung "aa", die zweite "ab", die dritte "ac" usw.

Um das Backup zurückzusichern müssen diese Dateien zunächst wieder zu einer sehr großen einzelnen Datei zusammengeführt werden. Diese Datei ist aber möglicherweise zu groß, deshalb werden - wie beim Sichern - mehrere Befehle auf einmal angewandt: Mit **cat userdir-backup-%datum.tar.bz2.* | bunzip2 | tar xv** kann das Backup zurückgesichert werden. Diese Befehlskette muss aber im Root-Verzeichnis (`/`) ausgeführt werden oder man muss das entpackte Home-Verzeichnis entsprechend verschieben.

Damit das Backup-Script nun wöchentlich ausgeführt wird, muss ein sogenannter Crontab-Eintrag erstellt werden. Mit folgendem Crontab-Eintrag wird das Script jeden Sonntag (5. Wert: 0) um 04:00 Uhr (2. Wert: 4; 1. Wert: 0) als Benutzer "root" ausgeführt.

```
0 4 * * 0 root /usr/local/sbin/userdir-backup.sh
```

Dieser Eintrag wird der Datei `/etc/crontab` als letzte Zeile hinzugefügt. Damit ist das Backup-Script fertig installiert.

4.4.5. sudoers

Da Linux ein Multiuser-Betriebssystem ist, haben gewöhnliche Benutzer nur wenige Rechte. Im Gegensatz dazu hat der Superuser, der Administrator oder "root", alle Rechte. So darf beispielsweise nur der Administrator das Passwort eines x-beliebigen Benutzers ändern. Manche CGI-Programme für die Weboberfläche benötigen genau dieses Recht (Passwort ändern oder Passwort zurückstellen). Damit dies trotzdem realisierbar ist, gibt es ein Programm, mit dessen Hilfe man andere Programme als root ausführen kann. Dieses Programm heißt "sudo" und muss wahrscheinlich mit **apt-get install sudo** noch nachinstalliert werden.

Alle CGI-Programme laufen unter dem Benutzer *www-data*. Diesem Benutzer soll nun erlaubt werden, einige Programme als root auszuführen. Mit **visudo** wird die Konfigurationsdatei */etc/sudoers* bearbeitet. Sie muss folgenden Inhalt haben:

```
#
# /etc/sudoers
#

# Host alias specification
Host_Alias LOCAL = 127.0.0.1, 192.168.0.1

# User alias specification
User_Alias WWW = www-data

# Runas alias specification
Runas_Alias USERS = root, homepage

# Cmnd alias specification
Cmnd_Alias COMMANDS = /usr/sbin/usermod, /usr/local/sbin/generate_apache_auth,\
                        /usr/bin/smbpasswd, /usr/local/sbin/change-aliases,\
                        /usr/local/sbin/proxy-manager, /usr/local/sbin/firewall,\
                        /usr/local/sbin/mirror-hp.sh, /usr/sbin/groupadd,\
                        /usr/sbin/useradd, /bin/chown, /bin/ln, /bin/mkdir,\
                        /usr/sbin/setquota,\
                        /usr/local/sbin/make_mozilla_config.pl, /usr/bin/passwd,\
                        /usr/local/sbin/chpw.pl, /usr/local/sbin/rmsusr.pl,\
                        /usr/local/sbin/user-status

# User privilege specification
root ALL = (ALL) ALL
WWW LOCAL = (USERS) NOPASSWD: COMMANDS
#www-data ALL=(ALL) NOPASSWD: ALL
```

Der Benutzer "www-data" kann jetzt mit **sudo /usr/sbin/useradd neuer_benutzer** zum Beispiel einen neuen Benutzer hinzufügen, was sonst nur root durfte.

4.5. Zusätzliches

4.5.1. Homepage hochladen

Zunächst befindet sich die Homepage im Laufwerk "Homepage" unter Samba. Und dort genauer im Ordner "intranet". Alles was sich in diesem Ordner befindet, wird hochgeladen. Das Programm zum Hochladen heißt **weex** und wird mit **apt-get install weex** nachinstalliert. Damit der Server die Homepage halbautomatisch hochladen kann (nach dem Passwort wird noch gefragt), wird ein Benutzer mit dem Namen "homepage" erstellt. Sein Homeverzeichnis ist `/home/homepage/`. Die Konfigurationsdatei für weex heißt `/home/homepage/.weex/weexrc`. Sie muss mit folgendem Inhalt erstellt werden:

```
#
# /home/homepage/.weex/weexrc
#
[Belwue]
FtpPassive = true
HostName = www.belwue.de
LoginName = loginname
SrcDir = /usr/local/samba/homepage/intranet
DestDir = /htdocs
IgnoreLocalFile = FINDER.DAT
IgnoreLocalDir = RESOURCE.FRK
IgnoreRemoteDir = {
    /htdocs/wusage
    /htdocs/stat
}
IgnoreRemoteFile = /htdocs/homepage.count

[default]
Monochrome = True
```

"loginname" muss natürlich noch ersetzt werden.

Der Superuser (root) kann mit **su -c '/usr/bin/weex Belwue' - homepage** die Homepage updaten. Dabei wird er nach dem FTP-Passwort gefragt.

4.5.2. Quota

Wenn *Quota* aktiviert ist, dann kann man den freien Speicherplatz für jeden Benutzer einzeln begrenzen. Dabei protokolliert das Quota-System in einer Datei mit, wieviel Plattenplatz ein beliebiger Benutzer verbraucht.

Um Quota einzurichten, müssen zunächst zwei Programmpakete installiert werden: **apt-get install quota quotatool**. Außerdem muss der Kernel quota unterstützen. Im Standardkernel ist bisher nur Unterstützung für das Dateisystem *ext2fs* enthalten. Um mit *ReiserFS* Quota zu benutzen, muss der Kernel gepatcht werden. Näheres steht in Abschnitt 3.2.

Sind die Grundvoraussetzungen (Software und Kernel) vorhanden, muss die Datei `/etc/fstab` verändert werden. In der Zeile, die mit `"/dev/md0"` beginnt, muss noch die Mount-Option *usrquota* ergänzt werden. Die Zeile sieht dann so aus:

```
/dev/md0      /      reiserfs      defaults,usrquota      0      0
```

Nach einem Neustart muss mit **init S** in den Single-User-Mode gewechselt werden. Hier kann dann die Quota-Protokoll-Datei mit **quotacheck -avu -F vfsold** erstellt werden. Anschließend kann mit **init 2** wieder in den normalen Runlevel zurückgewechselt werden. Quota ist nun aktiviert.

Mit dem Programm **setquota** kann die Begrenzung des Festplattenplatzes für einen bestimmten Benutzer verändert werden. Die Aufruf-Syntax des Programms ist folgende:

```
setquota -u user <limit> <limit> 0 0 /dev/md0
```

Mit "limit" ist die Begrenzung in Kilobytes (1 KB = 1024 Bytes) gemeint.

4.5.3. ntpdate

Mit ntpdate lässt sich die Server-Uhr nach einer Atomuhr im Internet stellen. Zuerst muss mit **apt-get install ntpdate** das Programm installiert werden. Bei der Installation wird nach einem NTP-Server gefragt. Einfach nichts eingeben.

Am besten ist es, wenn bei jeder Internet-Einwahl die Uhr neu gestellt wird. Dazu wird die Datei `/etc/ppp/ip-up.d/ntpdate` erstellt:

```
#!/bin/sh
#
# /etc/ppp/ip-up.d/ntpdate
#

ntpdate -s ntp1.ptb.de
hwclock -w
```

Die Datei muss mit **chmod 755 /etc/ppp/ip-up.d/ntpdate** noch ausführbar gemacht werden.

5.1.2. Gruppen

Durch eine Mitgliedschaft in einer Gruppe bekommen die Benutzer mehr Rechte unter Linux. Wie bei den Benutzern gibt es zunächst eine Gruppe *root* (Gruppen-ID 0) und dann weitere Systemgruppen. Es existiert auch die Gruppe *nogroup* (GID 65534), was die Entsprechung zum Benutzer *nobody* darstellt. Zusätzlich gibt es jetzt eine Gruppe *users* (GID 100). In dieser Gruppe ist jeder reale Benutzer Mitglied. Alle Mitglieder der Gruppe *internet* (GID 101) dürfen das Internet aktivieren. Die Mitglieder der Gruppe *admin* (GID 102) haben unter Samba Administratoren-Rechte, d.h. sie dürfen jede Freigabe beschreiben. Jeder Schüler ist in der Gruppe *schueler* (GID 103) und jeder Lehrer in der Gruppe *lehrer* (GID 104). Jeder Benutzer, der die Homepage bearbeiten darf, muss Mitglied der Gruppe *homepage* (GID 105) sein.

Zusätzlich existiert für jeden einzelnen Benutzer eine eigene Gruppe, die denselben Namen wie der Benutzername trägt (GID und UID müssen nicht identisch sein).

5.2. Benutzermanager

Der Benutzermanager ist ein CGI-Programm für den Webserver. Es ist in Perl geschrieben und wird vom Browser aus angewendet. Die Bedienung ist selbsterklärend. Der Quellcode ist hier nicht abgedruckt, er ist auf der CD zu finden. Hier werden nur die Befehle genannt, die im Benutzermanager jeweils ausgeführt werden

5.2.1. Benutzer löschen

Jeder Benutzer existiert als Benutzer unter Linux und als Samba-Benutzer. Zunächst wird der Benutzer aus der Samba-Benutzerdatenbank gelöscht:

```
$ smbpasswd -x $login
```

Als nächstes wird das *public_html*-Verzeichnis des Benutzer gesichert werden. Es wird einfach in das Verzeichnis "oldhomes" auf dem Webserver verschoben. Außerdem müssen die richtigen Rechte gesetzt werden:

```
$ mv /home/public_html/$login /usr/local/httpd/htdocs/oldhomes/
$ chown -h -f -R root.www-data /usr/local/httpd/htdocs/oldhomes/$login
$ chmod -R 755 /usr/local/httpd/htdocs/oldhomes/$login
```

Danach kann der Benutzer, sein Homeverzeichnis und seine Gruppe gelöscht werden:

```
$ userdel -r $login
$ groupdel $login
```

Zum Schluss muss noch eine neue Passwort-Datei für apache erstellt werden, damit der Benutzer auch hier nicht mehr existiert.

```
$ /usr/local/sbin/generate_apache_auth
```

5.2.2. Passwort ändern

Das Passwort eines Benutzers muss an drei Stellen geändert werden: In der Linux-Benutzerdatenbank, bei Samba und dann in der apache-Passwort-Datei. Drei Befehle müssen dazu ausgeführt werden. Bei den ersten beiden muss das neue Passwort zweimal eingegeben werden.

```
$ passwd $login
$ smbpasswd -a $login
$ /usr/local/sbin/generate_apache_auth
```

5.2.3. Rechte eines Benutzers ändern

Zu den Rechten zählen zum einen die verschiedenen Gruppen, in denen der Benutzer Mitglied ist und zum andern die Quota-Einstellung, die angibt, wieviel Festplattenplatz der jeweilige Benutzer verbrauchen darf. Neben den Rechten kann auch noch die Login-Shell eingestellt werden: Ist die Shell */bin/bash*, dann darf sich der Benutzer direkt (bzw. über das Netzwerk) am Server einloggen. Soll das verhindert werden, dann muss */bin/false* eingestellt werden. Sinnvollerweise sollten sich nur Administratoren direkt einloggen dürfen. Die Gruppen können eine Kombination der folgenden Liste sein: *schueler*, *lehrer*, *internet*, *admin*, *homepage*.

Gruppenzugehörigkeit und Shell ändern:

```
$ usermod -s $shell -G users,$gruppen,$login $login
```

Quota ändern - \$limit muss in KB angegeben werden:

```
$ setquota $login /dev/md0 $limit $limit 0 0
```

Zum Schluss muss wieder die apache-Passwort-Datei neu erstellt werden:

```
$ /usr/local/sbin/generate_apache_auth
```

5.2.4. Benutzer sperren bzw. freigeben

Ein Benutzer kann gesperrt werden und wieder freigegeben werden. Wenn ein Benutzer gesperrt ist, kann er sich nicht mehr einloggen. Sein Zugang ist deaktiviert. Um einen Benutzer zu sperren, müssen drei Befehle ausgeführt werden:

```
$ passwd -l $login
$ smbpasswd -d $login
$ /usr/local/sbin/generate_apache_auth
```

Um einen Benutzer, der gesperrt ist, wieder freizugeben, müssen auch drei Befehle ausgeführt werden:

```
$ passwd -u $login
$ smbpasswd -e $login
$ /usr/local/sbin/generate_apache_auth
```


5.2.5. Neuer Benutzer erstellen

Als erster Schritt muss eine neue Gruppe erstellt werden, die den gleichen Namen trägt wie der Benutzer, den man einrichten will:

```
$ groupadd $login
```

Danach kann der Benutzer erstellt werden. Ist der Benutzer ein Schüler, so ist sein Homeverzeichnis `/home/schueler/$login`; bei Lehrern heißt das Homeverzeichnis `/home/lehrer/$login`. "`$name`" ist der vollständige Name, "`$login`" ist der Login-Name. "`$groups`" ist eine Kombination aus *schueler*, *lehrer*, *internet*, *admin*, *homepage*. Außerdem wird mit dem zweiten Befehl gleich das Passwort gesetzt.

```
$ useradd -c "$name" -d $home -g $login -G users,$groups,$login \
-s $shell -m $login
$ passwd $login
```

Danach wird das "public_home"-Verzeichnis eingerichtet:

```
$ mkdir -m 0755 /home/public_html/$login
$ chown $login.$login /home/public_html/$login
ln -s /home/public_html/$login $home/public_html
```

Anschließend wird der Benutzer für Samba erstellt. Dabei muss das Passwort gleich gesetzt werden.

```
$ smbpasswd -a $login
```

Jetzt muss noch der Quota-Wert gesetzt werden. Ein gewöhnlicher Benutzer darf maximal 50 MB benutzen:

```
$ setquota $login /dev/md0 51200 51200 0 0
```

Damit der Benutzer über den Webserver sein Passwort ändern kann, muss die apache-Passwortdatei neu generiert werden:

```
$ /usr/local/sbin/generate_apache_auth
```

Zum Schluss muss die Mozilla-Konfiguration angepasst werden. Hier werden der Name des Benutzer und seine EMail-Adresse eingerichtet.

```
$ /usr/local/sbin/make_mozilla_config.pl $home/Mozilla "$login" "$name"
```

5.2.6. Informationen über die Benutzer sammeln

Mit **passwd -S \$login** erfährt man, ob ein Benutzer gesperrt ist oder nicht und wann der das letzte Mal sein Passwort geändert hat:

```
$ passwd -S andreas
andreas P 03/02/2003 0 99999 7 -1
```

Hier ist der Benutzer nicht gesperrt. Bei einem gesperrten Benutzerkonto würde statt dem "P" ein "L" stehen (locked). Mit **quota -v \$login** erfährt man den benutzten Festplattenplatz des Benutzers. Mit **groups \$login** sieht man, in welchen Gruppen der Benutzer Mitglied ist.

Wer sich lieber die Informationen ohne Hilfe bestimmter Programme besorgen möchte, kann auch die Benutzerdatenbank von Linux direkt anschauen. Sie besteht aus drei Textdateien: In `/etc/passwd` stehen die verschiedenen Benutzer mit UID, vollständiger Name, Homeverzeichnis, Login-Shell. Das Passwort steht in der Datei `/etc/shadow`. Die Gruppenmitgliedschaft ist in der Datei `/etc/group` festgehalten.

5.3. proxymanager

Der Proxymanager regelt den Zugriff auf das Internet: Bevor ein Benutzer das Internet benutzen kann, muss er es zuerst freischalten. Das passiert über den Webserver mit dem Browser. Dabei wird das Internet nicht für diesen Benutzer speziell freigeschaltet, sondern für diesen Computer. Die Computer werden anhand ihrer IP-Adresse unterschieden.

Die Computer greifen nie direkt auf einen Webserver im Internet zu. Alle Anfragen gehen über den Server, genauer über den Proxyserver (Der Server ist also kein Router). So wie der Proxserver in Abschnitt 4.3.3 eingerichtet wurde, werden alle Anfragen gefiltert. Wir müssen nun diesen Filter erweitern. Dazu verwenden wir einen selbstgeschriebenen Filter. Da das Schulnetz nicht besonders groß ist, spielt Geschwindigkeit keine so große Rolle und wir können Perl als Programmiersprache verwenden.

Das Filterprogramm ist auf der CD zu finden und heißt `/usr/local/sbin/squid-redirector`. Es überprüft mit Hilfe des *proxymanager*-Programms, das weiter unten beschrieben wird, ob das Internet für die IP-Adresse des anfragenden Rechners freigeschaltet ist. Wenn das Internet nicht freigeschaltet ist, wird im Browser eine Fehlermeldung angezeigt ("Internet ist nicht aktiviert"). Diese Fehlermeldung ist in Wirklichkeit ein CGI-Programm. Wenn das Internet aktiviert ist, wird die Anfrage an den Inhaltsfilter *squidGuard* weitergeleitet.

Das *proxymanager*-Programm ist in `/usr/local/sbin/proxy-manager` zu finden. Es ist auch ein Perl-Programm. Das Programm kann mit verschiedenen Parametern aufgerufen werden:

```
$ /usr/local/sbin/proxy-manager -h
Syntax: proxy-manager [-hdocak] [login] [ip-adresse] [zeitsekunden]

-h   Dieser Hilfetext
-d   der eigentliche proxy-manager wird im Hintergrund gestartet
-o   öffnet eine Verbindung für "ip-adresse" und schließt sie
      automatisch, wenn "zeitsekunden" erreicht sind
-c   schließt die Verbindung für "ip-adresse" manuell
-a   prüft, ob für "ip-adresse" eine Verbindung besteht
-k   beendet den proxy-manager und löscht alle Verbindungen
-t   tested, ob der proxy-manager im Hintergrund läuft
      (siehe /var/run/proxy-manager.pid).

[login]
      Der Benutzer, der eine Verbindung öffnet, etc. wird geloggt.

[ip-adresse]
      Jede beliebige Form ist erlaubt:
```

Beispiel: 192.168.0.168
Es kann auch 'all' verwendet werden.

```
[zeitsekunden]
    Anzahl der vergangenen Sekunden seit seit 00:00:00, Jan 1, 1970.
    Zu ermitteln z.B. durch "date +%s".
```

Hinweise:

Falls Fehler beim Start des proxy-managers auftreten
sollten: die Zugriffsrechte überprüfen:
/var/log/internet-freigabe
/var/state/internet-freigabe

```
proxy-manager 1.2 (2003-08-20)
(c) 2001 Thomas Bleher <ThomasBleher@gmx.de>
(c) 2002,2003 Andreas Dangel <adabolo@adabolo.de>
```

proxy-manager comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions; see the GNU GPL.

Im Firewall-Script wird der Proxymanager mit **proxy-manager -d** als Hintergrundprozess gestartet. Das ist nötig, denn das Internet kann immer nur für eine bestimmte Zeit freigeschaltet werden. Der Hintergrundprozess prüft jede Minute, ob für einen freigeschalteten Computer die Zeit abgelaufen ist und deaktiviert gegebenenfalls das Internet. Alle weiteren Parameter sind ja im Hilfetext erklärt. Das CGI-Programm, mit dem man das Internet letztendlich aktivieren kann, ruft ebenfalls den Proxymanager auf, z.B. mit **proxy-manager -o andreas 192.168.0.140 1067188761**. Daraufhin schreibt der Proxymanager in die Datei /var/state/internet-freigabe folgende Zeile:

```
192.168.0.140:1067188761
```

Der Hintergrundprozess überprüft dann, ob der angegebene Zeitpunkt vorbei ist und entfernt die Zeile gegebenenfalls wieder.

Damit der Proxyserver (squid, siehe Abschnitt 4.3.3) unseren neuen Filter auch verwendet, muss die Datei /etc/squid.conf bearbeitet werden. Folgender Eintrag muss abgeändert werden:

```
redirect_program /usr/local/sbin/squid-redirector
```

Mit **/etc/init.d/squid restart** wird unser neuer Filter verwendet.

Alle hier erwähnten Programme sind auf der CD zu finden.

5.3.1. Logdatei

Die Logdatei für den Proxymanager heißt /var/log/internet-freigabe. Hier wird mitgeloggt, wer das Internet jeweils freischaltet. Damit diese Datei nicht ins Unendliche anwächst, wird wieder logrotate konfiguriert: Die Datei /etc/logrotate.d/internet-freigabe muss mit folgendem Inhalt existieren:

```
/var/log/internet-freigabe {
    rotate 7
    weekly
```

```

compress
missingok
notifempty
}

```

5.4. Info-Tools

5.4.1. Festplattenplatz

Um den verbrauchten und den noch freien Festplattenplatz zu erfahren, wird **df** verwendet. **df** steht für "Disk free". Mit der Option "-h" werden die Angaben besser verständlich formatiert; es werden Angaben in MB und GB gemacht. Hier ein Beispielaufwurf von **df -h**:

```

$ df -h

```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/md0	65G	24G	41G	37%	/
/dev/md1	9.3G	237M	9.0G	3%	/var

5.4.2. uptime

Wie lange der Server seit dem letzten Neustart schon läuft, erfährt man mit dem Befehl **uptime**:

```

$ uptime
12:46am up 3 days, 4:31, 0 users, load average: 0.00, 0.00, 0.00

```

Der Server läuft hier also seit 3 Tagen, 4 Stunden und 31 Minuten. Außerdem zu sehen ist die durchschnittliche Auslastung (load average). Ein Wert von 0 bedeutet, dass der Server überhaupt nicht ausgelastet ist.

5.5. Weitere CGI-Skripte

Alle CGI-Skripte sind auf der CD zu finden. Beim Installieren ist zu beachten, dass die CGI-Skripte ausführbar sein müssen. Es müssen also gegebenenfalls die Attribute mit **chmod 755 script.cgi** angepasst werden.

5.5.1. Passwort ändern

Damit jeder Benutzer sein Passwort ändern kann, gibt es ein CGI-Skript. Es ist mit dem Browser über den Webserver erreichbar. Das Skript wird unter `/usr/local/httpd/htdocs/passwd/index.cgi` installiert.

5.5.2. Mail-Weiterleitung konfigurieren

Jeder Benutzer hat ja eine Mailadresse der Form: *login@gm.rt.schule-bw.de*. Alle Mails, die an diese Adresse geschickt werden, können auch an eine externe Mailadresse (z.B. gmx.de oder web.de) weitergeleitet werden. Mit dem CGI-Script, das unter `/usr/local/httpd/htdocs/mail-config/index.cgi` installiert wird, kann dies jeder Benutzer selber einrichten.

5.5.3. Benutzerverzeichnisse

Mit diesem CGI-Script lassen sich die verschiedenen public-html-Verzeichnisse der Benutzer anschauen. Diese Verzeichnisse sind über die Adresse `http://server/~login/` erreichbar. Das Script wird unter `/usr/local/httpd/cgi-bin/benutzer` installiert.

III. Clients

Kapitel 6. Windows 98SE

In der Schule verwenden wir Windows 98 SE als Client-Betriebssystem. Windows 98 besitzt noch einen DOS-Modus, sodass ältere DOS-Programme auch weiterhin benutzt werden können. Außerdem kann so der Linux-basierte Partitionsmanager verwendet werden, der die Partition sichert und bei Bedarf wieder zurückkopiert.

6.1. Zusammenarbeit mit dem Netzwerk und Samba

6.1.1. Netzwerk

Zunächst muss der Windows-Rechner ins Netzwerk eingebunden werden, d.h. er muss eine passende IP-Adresse besitzen. Diese erhält er über DHCP ("IP-Adresse automatisch beziehen"). Zu dieser Option kommt man über: Rechtsklick auf "Netzwerkumgebung" und "Eigenschaften" wählen. Dann im Register "Konfiguration" Doppelklick auf "TCP/IP". Hier ist die Option im Register "IP-Adresse" zu finden.

6.1.2. Computer-Name

Um die Samba-Logdateien etwas übersichtlicher zu gestalten erhält jeder Computer einen eindeutigen Namen im Windows-Netzwerk. Die Benennung erfolgt so: RXXX-PC#. "XXX" steht für die Raum-Nummer, sie ist immer dreistellig (z.B. R023). "#" ist die Nummer des PCs im Raum. Die Rechner werden im Raum einfach durchnummeriert (Bsp. R023-PC2 oder R104-PC15). Zu dieser Option kommt man über: Rechtsklick auf "Netzwerkumgebung" und "Eigenschaften" wählen. Dann zum Register "Identifikation" wechseln. Hier den Computernamen eintragen. Die Arbeitsgruppe heißt "workgroup".

6.1.3. Netz-Logon

Damit sich ein Benutzer beim Start von Windows einloggen kann und damit das Passwort vom Linux-Server bestätigt wird, ist auch eine spezielle Einstellung nötig. Zur dieser Option kommt man über: Rechtsklick auf "Netzwerkumgebung" und "Eigenschaften" wählen. In diesem Dialog muss die "Primäre Netzwerkanmeldung" auf "Client für Microsoft-Netzwerke" stehen. Außerdem muss dieser "Client für Microsoft-Netzwerke" mit einem Doppelklick konfiguriert werden. Im sich neu geöffneten Dialog wird "An Windows NT-Domäne anmelden" aktiviert. Die Windows NT-Domäne heißt "workgroup".

6.2. Netzwerkinstallation von best. Programmen

Bestimme Programme, wie z.B. StarOffice, bieten von sich aus eine Netzwerkinstallation an. Hier muss einfach als Installationspfad `P:\Programmname` gewählt werden. Meistens geht dies auch mit jedem anderen Programm, bei dem man den Installationspfad selber wählen kann. Eine etwas flexiblere Lösung ist die, dass man als Installationspfad `//server/programme/Programmname` wählt. Dann werden die Symbole (Icons) im

Startmenü bzw. auf dem Desktop entsprechend erstellt und wenn sich der Benutzer unter Windows nicht einloggt, sondern "Abbrechen" drückt, dann kann er die Programme trotzdem mit Doppelklick starten.

Die Programme, die im Netzwerk verfügbar sein sollen, werden also unter `//server/programme` installiert. In dieser Freigabe hat jeder Benutzer alle Rechte, d.h. ein x-beliebiger Schüler könnte hier alles löschen. Es ist also dringend ein Backup dieser Freigabe zu empfehlen.

6.3. Partitionsmanager

Der Partitionsmanager ist in Wirklichkeit ein Mini-Linux-System. Es wird mit dem Linuxloader LoadLin im DOS-Modus gestartet. Alle Programme sind auf einer Startbaren RAMDisk. Mit speziellen Argumenten, die man schon LoadLin mit auf den Weg gibt, kann die Partition z.B. vollautomatisch zurückgesichert werden. Das Programm, das die eigentliche Arbeit leistet, heißt *partimage*. Hier ein paar Aufruf-Beispiele:

```
C:\PARTIMG > loadlin bzimage root=/dev/ram initrd=initrd.gz rw manual
```

Hier wird ein Interaktivmodus gestartet ("manual"). Man kann hier z.B. manuell eine Netzwerkverbindung aufbauen, etc.

```
C:\PARTIMG > loadlin bzimage root=/dev/ram initrd=initrd.gz rw restore sicher.img
```

Hier wird das Image mit dem Dateinamen `sicher.img` zurückkopiert. Das Image liegt auf dem Laufwerk D: (unter Linux: `/dev/hda5`); das ist das erste logische Laufwerk. Zurückkopiert wird das Image auf das Laufwerk C: (`/dev/hda1`). Laufwerk C: wird also wiederhergestellt. Nach Abschluss des Vorgangs wird der Rechner automatisch neugestartet.

```
C:\PARTIMG > loadlin bzimage root=/dev/ram initrd=initrd.gz rw store sicher.img
```

Mit diesem Befehl wird ein Image von Laufwerk C: erstellt. Es wird unter `D:\SICHER.IMG` gespeichert. Der Rechner wird danach ebenfalls neugestartet.

```
C:\PARTIMG > loadlin bzimage root=/dev/ram initrd=initrd.gz rw update
//server/pub/sicher.img sicher.img
```

Dies ist ein Befehl und er muss in einer Zeile geschrieben werden. Der Pfadname `//server/pub/sicher.img` gibt an, wo sich das Image im Netzwerk befindet. Es wird dann auf den Rechner kopiert und als `sicher.img` auf dem Laufwerk D: gespeichert. Diese Funktion arbeitet leider nicht ganz ordnungsgemäß, sodass der "update"-Befehl leider nicht verwendet werden kann. Wenn ein Image schnell verteilt werden muss, dann kann im Netlogon-Script ein Befehl wie

```
copy \\server\pub\sicher.img D:\SICHER.IMG
```

eingefügt werden. Anschließend muss man sich nur noch unter Windows anmelden und das Image wird auf den jeweiligen Rechner kopiert.

Um das Zurückspielen des Images, das auf Laufwerk D: liegt, kann der entsprechende Befehl in eine Stapel-Datei (Batch-Datei) geschrieben werden. Danach muss `C:\CONFIG.SYS` bearbeitet werden. (Die Datei ist möglicherweise schreibgeschützt und/oder versteckt.) Die Datei muss ungefähr folgenden Inhalt haben:

```
[menu]
menuitem=W98, Windows 98 starten
menuitem=PART, Image zurückspielen
```



```

menudefault=W98,5

[W98]
DEVICE=C:\WINDOWS\setver.exe
device=C:\WINDOWS\COMMAND\display.sys con=(ega,,1)
Country=049,850,C:\WINDOWS\COMMAND\country.sys

[PART]
shell=c:\command.com /C C:\PARTIMG\RESTORE.BAT

[COMMON]

```

Damit erscheint bei jedem Windows-Start ein Menü ("Windows 98 Startmenü"). Wenn der Benutzer nach 5 Sekunden nichts ausgewählt hat, wird automatisch Windows gestartet. Wählt der Benutzer den Menüpunkt "Image zurückspielen" wird das Mini-Linux gestartet und der Computer wiederhergestellt.

Das Programm ist auf der CD zu finden.

6.4. Windows beschränken

Damit nicht jeder Benutzer in der Systemsteuerung herumspielt und etwas umstellt, wird diese deaktiviert. Das geschieht einfach mit einer Registrierungsdatei. So wird das Startmenü ein wenig kleiner und die Anzeige-Eigenschaften sind nicht mehr erreichbar. Es kann also kein Benutzer ein Bildschirmschonerpasswort setzen. Mit einer zweiten Registrierungsdatei können die Restriktionen wieder rückgängig gemacht werden. Die Registrierungsdateien können ganz bequem über einen Doppelklick eingelesen werden und sind nach einer Neuansmeldung aktiv.

Hier die Datei `restrict.reg`.

```

REGEDIT4

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer]
"NoSetFolders"=dword:00000001
"NoSetTaskbar"=dword:00000001
"NoFolderOptions"=dword:00000001
"NoSetActiveDesktop"=dword:00000001
"NoWindowsUpdate"=dword:00000001
"NoRecentDocsMenu"=dword:00000001
"NoRecentDocsHistory"=dword:00000001
"ClearRecentDocsOnExit"=dword:00000001
"NoFavoritesMenu"=dword:00000001
"NoPrinterTabs"=dword:00000001
"NoDeletePrinter"=dword:00000001
"NoAddPrinter"=dword:00000001
"NoSaveSettings"=dword:00000001

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Winlogon]
"DontDisplayLastUserName"=dword:00000001

[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Network]
"DisablePwdCaching"=dword:00000001

```

```
[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\Network]
"HideSharePwds"=dword:00000001
"DisablePwdCaching"=dword:00000001
"NoNetSetup"=dword:00000001
"NoNetSetupIDPage"=dword:00000001
"NoNetSetupSecurityPage"=dword:00000001

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\System]
"DisableRegistryTools"=dword:00000001
"NoDispCPL"=dword:00000001
"NoDispBackgroundPage"=dword:00000001
"NoDispScrSavPage"=dword:00000001
"NoDispAppearancePage"=dword:00000001
"NoDispSettingsPage"=dword:00000001
"NoProfilePage"=dword:00000001
"NoSecCPL"=dword:00000001
"NoPwdPage"=dword:00000001
"NoAdminPage"=dword:00000001
"NoProfilePage"=dword:00000001
"NoDevMgrPage"=dword:00000001
"NoConfigPage"=dword:00000001
"NoFileSysPage"=dword:00000001
"NoVirtMemPage"=dword:00000001

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders]
"Personal"="H:\\"
```

Hier die Datei `unrestrict.reg`.

REGEDIT4

```
[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer]
"NoSetFolders"=dword:00000000
"NoSetTaskbar"=dword:00000000
"NoFolderOptions"=dword:00000000
"NoSetActiveDesktop"=dword:00000000
"NoWindowsUpdate"=dword:00000000
"NoRecentDocsMenu"=dword:00000000
"NoRecentDocsHistory"=dword:00000001
"ClearRecentDocsOnExit"=dword:00000001
"NoFavoritesMenu"=dword:00000000
"NoPrinterTabs"=dword:00000000
"NoDeletePrinter"=dword:00000000
"NoAddPrinter"=dword:00000000
"NoSaveSettings"=dword:00000001

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Winlogon]
"DontDisplayLastUserName"=dword:00000001

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\Network]
"HideSharePwds"=dword:00000001
"DisablePwdCaching"=dword:00000001
"NoNetSetup"=dword:00000000
"NoNetSetupIDPage"=dword:00000000
"NoNetSetupSecurityPage"=dword:00000000

[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\System]
```

```
"DisableRegistryTools"=dword:00000000
"NoDispCPL"=dword:00000000
"NoDispBackgroundPage"=dword:00000000
"NoDispScrSavPage"=dword:00000000
"NoDispAppearancePage"=dword:00000000
"NoDispSettingsPage"=dword:00000000
"NoProfilePage"=dword:00000000
"NoSecCPL"=dword:00000000
"NoPwdPage"=dword:00000000
"NoAdminPage"=dword:00000000
"NoProfilePage"=dword:00000000
"NoDevMgrPage"=dword:00000000
"NoConfigPage"=dword:00000000
"NoFileSysPage"=dword:00000000
"NoVirtMemPage"=dword:00000000
```

```
[HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders]
"Personal"="H:\\"
```

Außerdem werden hier kleinere Feineinstellungen an Windows vorgenommen. So wird z.B. das Verzeichnis "Eigene Dateien" so konfiguriert, dass es auf das Homeverzeichnis (H:\) auf Samba zeigt. Da wird Windows so eingestellt, dass es beim Login-Dialog den Namen des vorigen Benutzers löscht. Außerdem wird der Passwort-Cache deaktiviert. Ist diese Funktion aktiv, dann speichert Windows in C:\WINDOWS für jeden Benutzer eine PWL-Datei, in der das Passwort gespeichert ist. Mit ein paar Hilfsprogrammen aus dem Internet kann mit dieser Datei das Passwort rekonstruiert werden.

IV. Sonstiges

Kapitel 7. Linux-Grundlagen

7.1. Einloggen/Ausloggen

Nachdem linux hochgefahren ist, erscheint ein Text-basierter Login-Bildschirm:

```
Debian GNU/Linux 3.0 server tty1
```

```
server login:
```

Hier kann man dann seinen Benutzernamen eingeben. Danach wird nach dem Passwort verlangt. Ist man eingeloggt, sieht der Bildschirm ungefähr so aus:

```
Last login: Sun Nov  9 09:35:49 2003 on tty1
Linux server 2.4.22 #1 Sun Aug 17 16:51:06 CEST 2003 i686 unknown
```

```
Most of the programs included with the Debian GNU/Linux system are
freely redistributable; the exact distribution terms for each program
are described in the individual files in /usr/share/doc/*/copyright
```

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have mail.
andreas@server:~$
```

Die letzte Zeile ist die Eingabeaufforderung. Hier können jetzt Befehle eingegeben werden.

Will man sich wieder ausloggen, so muss man den Befehl **logout** ausführen. Es erscheint dann wieder der Login-Bildschirm.

7.2. Linux herunterfahren (shutdown)

Linux herunterfahren kann nur der Superuser. Man muss also als Benutzer root eingeloggt sein. Jetzt kann man mit **init 0** oder **halt** Linux sofort herunterfahren. Mit **shutdown -h +15** fährt Linux in 15 Minuten herunter.

Will man Linux neustarten, so muss man **init 6** bzw. **reboot** ausführen. Auch mit Shutdown geht das Neustarten: **shutdown -r +15** startet Linux in 15 Minuten neu.

7.3. Dateiverwaltung

7.3.1. Dateien auflisten

Mit dem Befehl **ls** werden Dateien aufgelistet (ls = LiSt). Dieser Befehl entspricht dem DOS-Befehl *dir*. Mit bestimmten Parametern kann die Ausgabe variiert werden: **ls -l** gibt eine detailreichere Liste aus (-l für long).

Dabei gibt der Befehl zunächst mal das aktuelle Verzeichnis aus. Will man ein anderes Verzeichnis sehen, muss man z.B. **ls /home** aufrufen. Hier ein paar Beispiele:

```
$ ls
CD-ROOT                                source
CD-ROOT-Inhalt.txt                    source-20031015-214838.tar.bz2
Makefile                              source-20031016-214533.tar.bz2
archiv1.tar.bz2                       source-20031018-120615.tar.bz2
html                                   source-20031018-150309.tar.bz2
passw?rter.out                        source-20031018-184430.tar.bz2
passw?rter.pdf                        source-20031020-191948.tar.bz2
server-docu-sicherung.pdf              source-20031021-183556.tar.bz2
server-docu-sicherung.ps              source-20031023-191929.tar.bz2
server-docu-sicherung2.ps             source-20031025-222636.tar.bz2
server-docu-sicherung4.pdf            source-20031026-190024.tar.bz2
server-docu-sicherung4.ps             source-20031027-190336.tar.bz2
server-docu.out                       source-20031108-193713.tar.bz2
server-docu.pdf                       source-20031108-203958.tar.bz2
smb.conf                              template
smb.conf.neu

$ ls -l
total 7944
drwxr-xr-x  3 andreas andreas    4096 Aug 18 19:48 CD-ROOT
-rw-r--r--  1 andreas andreas    3081 Aug 19 16:03 CD-ROOT-Inhalt.txt
-rw-r--r--  1 andreas andreas     411 Oct 21 18:05 Makefile
-rw-r--r--  1 andreas andreas   7079 Aug 19 14:55 archiv1.tar.bz2
drwxr-xr-x  3 andreas andreas    8192 Nov  8 20:25 html
-rw-r--r--  1 andreas andreas      0 Sep  3 22:14 passw?rter.out
-rw-r--r--  1 andreas andreas   7940 Sep  3 22:14 passw?rter.pdf
-rw-r--r--  1 andreas andreas  214374 Oct 15 21:02 server-docu-sicherung.pdf
-rw-r--r--  1 andreas andreas 1924997 Oct 15 21:04 server-docu-sicherung.ps
-rw-r--r--  1 andreas andreas 1948182 Oct 15 21:05 server-docu-sicherung2.ps
-rw-r--r--  1 andreas andreas 1255649 Oct 15 21:06 server-docu-sicherung4.pdf
-rw-r--r--  1 andreas andreas 1947974 Oct 15 21:06 server-docu-sicherung4.ps
-rw-r--r--  1 andreas andreas      0 Nov  9 10:39 server-docu.out
-rw-r--r--  1 andreas andreas 264492 Nov  9 10:39 server-docu.pdf
-rw-r--r--  1 andreas andreas   3116 Nov  8 21:01 smb.conf
-rw-r--r--  1 andreas andreas   3083 Nov  9 10:34 smb.conf.neu
drwxr-xr-x  5 andreas andreas    4096 Nov  9 11:05 source
-rw-r--r--  1 andreas andreas   9991 Oct 15 21:48 source-20031015-214838.tar.bz2
-rw-r--r--  1 andreas andreas  26668 Oct 18 15:03 source-20031018-150309.tar.bz2
-rw-r--r--  1 andreas andreas  27297 Oct 18 18:44 source-20031018-184430.tar.bz2
-rw-r--r--  1 andreas andreas  28145 Oct 20 19:19 source-20031020-191948.tar.bz2
-rw-r--r--  1 andreas andreas  28841 Oct 21 18:35 source-20031021-183556.tar.bz2
-rw-r--r--  1 andreas andreas  29333 Oct 23 19:19 source-20031023-191929.tar.bz2
-rw-r--r--  1 andreas andreas  30701 Oct 25 22:26 source-20031025-222636.tar.bz2
-rw-r--r--  1 andreas andreas  32937 Oct 26 19:00 source-20031026-190024.tar.bz2
-rw-r--r--  1 andreas andreas  33429 Oct 27 19:03 source-20031027-190336.tar.bz2
-rw-r--r--  1 andreas andreas   35187 Nov  8 19:37 source-20031108-193713.tar.bz2
-rw-r--r--  1 andreas andreas 139364 Nov  8 20:39 source-20031108-203958.tar.bz2
drwxr-xr-x  2 andreas andreas    4096 Jul 10 18:39 template
$ ls /home
andreas ftp samba
```

Die erste Spalte in der Ausgabe von **ls -l** zeigt die Rechte an: Das erste Zeichen stellt den Typ dar ("-" ist eine normale Datei, "d" ein Verzeichnis). Die nächsten drei Zeichen stellen die Rechte für den Besitzer der Datei/des Verzeichnisses dar. "r" heißt Leserecht, "w" Schreibrecht und "x" Ausführrecht. Die folgenden drei Zeichen

stellen die Rechte für die Gruppe dar, der diese Datei gehört. Die letzten drei Zeichen sind die Rechte, die jeder andere Benutzer des Systems an der Datei hat.

In der dritten Spalte steht der Besitzer und in der vierten Spalte die Gruppe. In der fünften Spalte steht die Größe der Datei in Bytes. Danach kommt das Datum der letzten Änderung und schließlich der Dateiname.

7.3.2. Datei/Verzeichnis erstellen

Mit **touch dateiname** kann eine leere, 0-Bytes-große Datei erstellt werden. Ein Verzeichnis erstellt man mit **mkdir verzeichnisname**.

7.3.3. Verzeichnis wechseln

Das aktuelle Verzeichnis erfährt man mit **pwd** (Print Working Directory). Mit **cd neues_verzeichnis** wechselt man das aktuelle Arbeitsverzeichnis.

7.3.4. Datei/Verzeichnis löschen

Mit **rm dateiname** wird eine Datei gelöscht. Ein leeres Verzeichnis kann man mit **rmdir verzeichnis** löschen. Ist das Verzeichnis noch nicht leer, kann das Verzeichnis mit dem ganzen Inhalt gelöscht werden: **rm -r verzeichnis**.

Hinweis: Gelöschte Dateien sind in der Regel nicht mehr wiederherstellbar!

7.3.5. Dateien kopieren

Mit **cp quelle ziel** kann man Dateien kopieren. Der Befehl funktioniert wie der DOS-Befehl "copy". Wird als Quelle eine Datei angegeben, so kann das Ziel ebenfalls eine Datei sein oder ein Verzeichnis, in das die Datei hineinkopiert werden soll. Ist die Quelle ein Verzeichnis, so kann das Ziel auch nur ein Verzeichnis sein, in welches dann das Quellverzeichnis hineinkopiert wird.

Hinweis: Vorhandene Dateien werden ohne Nachfrage überschrieben!

7.3.6. Dateien verschieben

Mit **mv quelle ziel** kann man Dateien verschieben. Der Befehl funktioniert wie der DOS-Befehl "move". Quelle und Ziel können wie beim "cp"-Befehl entweder Dateien sein oder Verzeichnisse.

Auch hier werden vorhandene Dateien ohne Nachfrage überschrieben.

7.3.7. Textdateien anzeigen

Mit **less textdatei** kann man eine Textdatei (z.B. ein Konfigurationsdatei) bequem anschauen. Mit den Cursor-Tasten scrollt man durch den Text. Mit einem Tastendruck auf "q" verlässt man den Datei-Anzeiger.

7.4. Benutzerverwaltung

Die Benutzer unter Linux werden in einer Datenbank verwaltet. Diese Datenbank besteht aus mehreren Textdateien: In `/etc/passwd` stehen die Benutzer, deren Homeverzeichnisse, die jeweilige Shell. Die dazugehörigen Passwörter stehen in der Datei `/etc/shadow`. Sie sind verschlüsselt. Mit dem Befehl **mkpasswd** lassen sich solche verschlüsselten Passwörter von Hand herstellen. In `/etc/group` stehen die Gruppen, die es gibt. In `/etc/gshadow` stehen Passwörter für die einzelnen Gruppen; diese Datei wird in der Regel nicht benötigt. Mit dem Befehl **vipw** lässt sich `/etc/passwd` bearbeiten. Mit **vigr** lässt sich `/etc/group` bearbeiten. Allerdings muss man dazu das Dateiformat kennen.

7.4.1. Benutzer erstellen

Um einen neuen Benutzer anzulegen, verwendet man den Befehl **useradd**. **useradd neuer_user** legt beispielsweise den Benutzer `neuer_user` an. **useradd --help** gibt Auskunft über die verschiedenen Argumente, die man dem Befehl geben kann. Nachdem der Benutzer mit diesem Befehl angelegt wurde, muss noch das Homeverzeichnis unterhalb von `/home/` erstellt werden. Außerdem sollte mit **passwd neuer_user** noch ein Passwort gesetzt werden.

7.4.2. Benutzer bearbeiten

Ein vorhandener Benutzer wird mit **usermod** bearbeitet. Mit **usermod -s /bin/bash user** erhält der Benutzer `user` als Shell `/bin/bash`. **usermod --help** gibt auch hier wieder Auskunft über die verschiedenen Argumente.

7.4.3. Benutzer löschen

Mit **userdel user** wird der Benutzer `user` gelöscht. Sein Homeverzeichnis bleibt erhalten. Mit **userdel -r user** wird auch sein Homeverzeichnis gelöscht. Es gibt keine Möglichkeit, die Daten wiederherzustellen!

7.4.4. Gruppe erstellen

Um eine neue Gruppe anzulegen, wird der Befehl **groupadd gruppe** verwendet.

7.4.5. Gruppe löschen

Mit **groupdel gruppe** wird die Gruppe `gruppe` gelöscht.

7.5. Paketverwaltung

Hierzu steht schon einiges unter Abschnitt 3.8.

7.6. Editor

Da unter Linux fast alle Konfigurationsdateien Textdateien sind, sollte man wenigstens einen Texteditor beherrschen. In diesem Abschnitt wird der Texteditor *vim* vorgestellt. Er ist ganz anders als ein Editor aus der Windows- bzw. DOS-Welt zu bedienen, aber *vim* ist sehr verbreitet.

7.6.1. Datei öffnen bzw. neue Datei erstellen

Um die Datei `/etc/hosts` zu bearbeiten, wird einfach **vim /etc/hosts** ausgeführt. Die Datei wird dann geöffnet. Existiert die angegebene Datei nicht, dann wird sie erstellt. So lassen sich also auch neue Dateien erstellen.

7.6.2. Dateien bearbeiten und speichern

Nach dem Öffnen muss man *vim* erstmal in den Editiermodus umschalten. Das geht einfach, indem man die Taste "i" drückt. In der untersten Zeile steht dann "-- INSERT --". Jetzt kann man wie gewohnt den Text eingeben. Die Tasten "Pos1", "Ende", "Einf" und "Entf" funktionieren wie erwartet. Ist man mit dem Editieren fertig, verlässt man den Editiermodus mit der "Esc"-Taste. Die letzte Zeile wird wieder leer. Jetzt muss man einfach ":w" eingeben und mit der Eingabetaste bestätigen. In der letzten Zeile wird der Befehl übrigens angezeigt. Will man den Editor beenden, gibt man ":q" und Eingabetaste ein. Speichern und gleichzeitiges Beenden geht mit ":x" und Eingabetaste.

7.7. Prozessmanagement

Mit **ps awx** erfährt man, welche Prozesse im Moment laufen. Außerdem wird die Prozess-ID (PID) angezeigt. Mit **kill \$PID** kann man den Prozess mit der Nummer \$PID beenden. Funktioniert das nicht, hilft vielleicht **kill -9 \$PID** weiter.

7.8. Linux Support

Zu fast jedem Befehl gibt es unter Linux eine Handbuchseite (Manual Page). Diese Seite lässt sich mit **man befehl** ansehen. Manchmal ist auch **info befehl** aufschlussreicher.

Außerdem bietet fast jeder Befehl selber eine kleine Hilfestellung an. Diese erhält man mit dem Argument "--help" oder "-h": **befehl --help**.

Unter `/usr/share/doc/` gibt es für jeden Programmpaket ein Unterverzeichnis. In diesem Verzeichnis ist manchmal auch eine Dokumentation enthalten.

Im Internet unter www.tldp.org (<http://www.tldp.org>) gibt es u.a. auch sogenannte HOWTOs. Das sind Kurzanleitungen, die einen ganz bestimmten Themenbereich erklären. Meistens sind es praxisnahe Anleitungen.