

# LINUX TUX GRUNDLAGEN





- 1 ZEITPLAN:..... 4**
  
- 2 GESCHICHTE: ..... 5**
  - 2.1 DIE GEBURT VON LINUX:..... 5
  - 2.2 UNIX: ..... 5
  
- 3 INSTALLATION: ..... 6**
  - 3.1 STANDARDINSTALLATION:..... 6
  - 3.2 INSTALLATION VIA NFS: ..... 7
  - 3.3 TIPPS ZUR INSTALLATION ..... 7
  
- 4 GRUNDBEGRIFFE (TCP/IP): ..... 8**
  - 4.1 IP-ADRESSE: (VERS. 4) ..... 8
  - 4.2 NETZWERKMASKE: ..... 8
  - 4.3 ADRESSBEREICHE: ..... 8
  - 4.4 GATEWAY:..... 8
  - 4.5 DNS-SERVER: ..... 8
  
- 5 NACHINSTALLATION:..... 8**
  
- 6 LINUX ALS WORKSTATION: ..... 9**
  - 6.1 DRUCKER UNTER LINUX ..... 9
    - 6.1.1 LPD:..... 9
    - 6.1.2 CUPS: ..... 9
  - 6.2 DER BOOTMANAGER: ..... 10
    - 6.2.1 LILO – DER LINUX-LOADER: ..... 10
    - 6.2.2 GRUB ..... 10
  - 6.3 WÄHLVERBINDUNGEN UNTER LINUX:..... 11
  
- 7 SYSTEMGRUNDLAGEN:..... 11**
  - 7.1 TERMINALS: ..... 11
  - 7.2 RUNLEVEL:..... 12
  - 7.3 BENUTZER / GRUPPEN: ..... 12
  - 7.4 DAS DATEISYSTEM:..... 12
  - 7.5 DIE ZUGRIFFSRECHTE: ..... 15
  - 7.6 STARTEN VON PROGRAMMEN: ..... 15
  - 7.7 X-WINDOWS: ..... 16



7.8	HILFESYSTEM: .....	17
7.9	WICHTIGE TOOLS / BETRIEBSSYSTEMBEFEHLE:.....	17
7.10	WICHTIGE VERZEICHNISSE:.....	18
7.11	WICHTIGE DATEIEN IM VERZEICHNIS /ETC UND DEREN BEDEUTUNG.....	18
7.12	DER STARTVORGANG IM DETAIL:.....	20
<b>8</b>	<b><u>ZUGRIFF AUF ANDERE DATEISYSTEME:.....</u></b>	<b>20</b>
8.1	LINUX-DATEIZUGRIFF AUF NOVELLSERVER .....	20
8.2	WINDOWSDRUCKER VON LINUX AUS ANSPRECHEN .....	21
8.3	ZUGRIFF AUF FREIGELEGEBENE ORDNER VON WINDOWSRECHNERN .....	21
<b>9</b>	<b><u>KONFIGURATION VON NETZWERKDIENTEN:.....</u></b>	<b>21</b>
9.1	INSTALLATION VON WEITEREN NETZWERKKARTEN:.....	21
9.1.1	AKTIVIERUNG DES PASSENDEN TREIBERS (MODUL):.....	21
9.1.2	EINSTELLEN DER ENTSPRECHENDEN ADRESSE.....	21
9.1.3	PRÜFEN DER EINSTELLUNGEN: .....	22
9.2	ROUTING: .....	22
9.3	TESTEN DER NETZWERKVERBINDUNGEN: .....	23
9.3.1	PING .....	23
9.3.2	TRACEROUTE.....	23
9.4	ZUWEISEN VON MEHREREN IP-ADRESSEN ZU EINER NETZWERKKARTE: .....	23
9.5	MASQUERADE: .....	24
9.5.1	Bis SUSE 6.4:.....	24
9.5.2	Bis SUSE 7.1: .....	25
9.5.3	AB SUSE 7.2: .....	25
9.5.4	AB SUSE 8.0: .....	25
9.6	WEB-SERVER: .....	26
9.6.1	VERWENDUNG VON PHP: .....	26
9.6.2	ZUGRIFF AUF BESTIMMTE WEBSEITEN MITTELS AUTHENTIFIZIERUNG: .....	27
9.7	FERNWARTUNG: .....	28
9.7.1	TELNET: .....	28
9.7.2	SSH (SECURE SHELL):.....	28
9.7.3	FISH:.....	28
9.7.4	WARTUNG MIT WEBMIN: .....	28
9.8	FTP-SERVER:.....	29
9.9	BENUTZERPLATZBESCHRÄNKUNG: .....	29
9.10	WARTUNG VON BENUTZERDATEN .....	30
9.10.1	WEBMIN-USERINTERFACE .....	30
9.11	ABSICHERN DES SYSTEMS.....	30
<b>10</b>	<b><u>PROZESSMANAGEMENT.....</u></b>	<b>31</b>
10.1	PROZESSE ANZEIGEN.....	32
10.2	PROZESSE ABBRECHEN .....	33
10.3	GRAPHISCHE PROZESS-TOOLS .....	34



**1 ZEITPLAN:**

Montag 05.07	
09.30 – 10.15	Begrüßung, Organisation, Aufbau und Anschluss der Rechner
10.30 – 12.00	Installation via NFS, IP Grundlagen, Dateisystem-Grundlagen
13.30 – 15.00	Grundkonfiguration (X-Windows, Netzwerk, Sound), Einrichten und Anpassen der grafischen Oberfläche, Tools, Browser, Software nachinstallieren
15.15 – 16.45	Drucker installieren, Wählverbindungen unter Linux, Updates einspielen, Email, Office, Zugriff auf lokale Dateisysteme
Dienstag 06.07.	
08.45 – 10.15	Konsolen, Arbeiten im Textmodus Verzeichnisstruktur, Konfigurationsfiles,
10.30 – 12.00	Hilfesystem, Manpages, Benutzerstruktur, Zugriffsrechte, Prozeßmanagement
13.30 – 15.00	Benutzerstruktur, Zugriffsrechte, Netzzugriff auf Novell- und Windowssysteme
15.45 – 16.45	Fernwartungsmöglichkeiten (telnet, ssh, Webmin) X-Windows Ausgabe umleiten
Mittwoch 07.07.	
08.45 – 10.15	Starten von Systemdiensten, Webserver – Grundkonfiguration FTP-Zugriff einrichten
10.30 – 12.00	ev.: Unix-Netze: Grundlagen (hosts, nfs, nis) Unix-Netze: NFS-Server/Client NIS-Server/Client



Die Hinweise in diesem Skriptum beziehen sich auf die Installation und Konfiguration der SuSE Distribution 9.1

## 2 Geschichte:

Die Informationen aus diesem Kapitel stammen überwiegend aus SELFLINUX (<http://selflinux.sourceforge.net>)

### 2.1 Die Geburt von Linux:

**1991**

Der 21jährige finnische Student *Linus Benedict Torvalds* beginnt damit, ein auf Minix angelehntes Betriebssystem für AT-386-Computer zu schreiben.

**1992**

*Linus Torvalds* verteilte die Version 0.12 per anonymous FTP im Internet, was zu einem sprunghaften Anstieg der Testerzahl führte. Da dieser Anstieg so groß wurde, dass die nötige Kommunikation nicht mehr per Email zu bewältigen war, wurde in den Usenet News die Gruppe `alt.os.linux` ins Leben gerufen. Dies hatte zur Folge, dass eine explosionsartige Weiterentwicklung des Systems in ganzen Internet stattfand und von *Linus Torvalds* koordiniert wurde.

### 2.2 UNIX:

Linux wurde wie oben bereits erwähnt aus einer UNIX Version entwickelt. Aus diesem Grund ist das nächste Kapitel einer Übersicht über die Entwicklung von Unix gewidmet.

Die Geschichte von UNIX liest sich stellenweise wie ein Heldenepos oder ein Kriminalroman, manchmal auch wie eine Grotteske. Schon die Geburt von UNIX vollzog sich unter eigenartigen Umständen, denn es entsprang einem gescheiterten Projekt:

Anfang 1969 gab es ein Gemeinschaftsprojekt des *MIT*, *General Electric* und den *Bell Labs* von *AT&T*, das Ideen für eine neue Generation von Betriebssystemen gesammelt hatte und daran ging, unter dem Namen *Multics* diese Ideen umzusetzen. Da weder Zeitplan noch Budget eingehalten werden konnten, zog sich *Bell Labs* sehr schnell aus dem Projekt zurück. *Ken Thompson* und *Dennis Ritchie*, zwei Mitarbeiter von *Bell Labs*, die an *Multics* mitgearbeitet hatten, waren von den Einfällen und Erfahrungen, die sie mit *Multics* gesammelt hatten, so beeindruckt, dass sie kurzerhand eine abgespeckte Version des ursprünglichen *Multics* selbst schrieben und unter dem Namen **Unics**, später **UNIX**, in die Welt setzten. (hauptsächlich um das Spiel „Space Travel“ auf eine PDP-7 zu portieren).

*AT&T* verkaufte die Lizenzen zu Unix zu nominellen Gebühren an Universitäten (u.a. auch an die Universität von Berkley die eine eigene Distribution unter der Bezeichnung BSD (Berkley System Distribution) hervorbrachten). Zur Verwendung von BSD waren jedoch auch Lizenzen von *AT&T* notwendig. Auch kommerzielle Anbieter übernahmen UNIX und entwickelten daraus ihre eigenen Systeme. Nachdem im Jahr 1984 *AT&T* die Lizenzgebühren deutlich an hob, entwickelte jeder Hersteller seine eigene Version weiter und die Systeme wie HP-UP, SunOS, AIX (IBM), ULTRIX (Digital), SINIX (Siemens) und XENIX (MS) wurden immer Inkompatibler. Verschiedene Standardisierungsbestrebungen wie System V (*AT&T*) und POSIX scheiterten.

Erfolg hatte dagegen ein Projekt, gestartet Anfang der 80er-Jahre am *MIT* von *Richard Matthew Stallman*, dem "letzten Hacker der Altvorderenzeit", das GNU-Projekt: GNU's Not UNIX. Sein Ziel war es, von Grund auf ein neues, UNIX-ähnliches Betriebssystem zu schreiben, das frei verfügbar sein sollte. *Stallman* wollte damit einen Gegenpol zur zunehmenden Proprietärisierung der Softwarewelt schaffen. Durch seinen intensiven Einsatz und Beiträge anderer Programmierer entstand bis Ende der 90er eine beachtliche und leistungsstarke Sammlung an UNIX-Tools. Auch wenn das System bislang nicht vollständig



ist, konnten sich die GNU-Tools dennoch auf vielen UNIXen etablieren, unter anderem auch deshalb, da einzelne UNIX-Anbieter ihre Einnahmequellen noch etwas auszubauen gedachten. Mit dem Grundpaket wurde z. B. kein C-Compiler mitgeliefert, worauf viele Systembetreuer, um Geld zu sparen, auf *Stallmans* GNU C-Compiler zurückgriffen, der ohnehin qualitativ besser war. So wurden die GNU-Tools ein systemübergreifender Quasi-Standard. Die freie Entwicklungsmethode hatte erreicht, woran die proprietären Standardisierungsversuche bislang gescheitert waren.

Auch von akademischer Seite wurde der immer zugeknöpfteren Haltung der UNIX-Vertreiber begegnet: zu Anfang wurde der Quellcode von AT&T den Universitäten offen zur Verfügung gestellt und so vielerorts als Tutorial für die Arbeitsweise eines Betriebssystems verwendet. Als AT&T den Quellcode unter Verschluss brachte, fiel diese Möglichkeit weg. Andy Tanenbaum, Informatik-Professor an der Freien Universität Amsterdam, entschloss sich daher, für seine Studenten eine eigene Version von UNIX zu schreiben, die nichts mit dem urheberrechtlich geschützten Code von AT&T zu tun hatte. Nach zwei Jahren harter Arbeit brachte er sein System unter dem Namen Minix heraus. Es war weniger für die praktische Arbeit, sondern in erster Linie als Lehrobjekt gedacht. Dennoch wurde es von sehr vielen Studenten auch praktisch auf dem heimischen PC eingesetzt, da es im Gegensatz zu den kommerziellen UNIXen für einen moderaten Preis zu haben war. Allerdings stieß Minix in diesem Einsatzgebiet sehr schnell an seine Grenzen. Viele seiner Anwender machten Tanenbaum Vorschläge und schickten Patches für Erweiterungen und Verbesserungen. Tanenbaum allerdings war damit sehr zurückhaltend, da er Minix in erster Linie als Tutorial sah, kam es ihm mehr auf eine knappe und klare Struktur als auf eine möglichst umfassende Funktionalität an.

Ein Minix-Anwender mit Namen Linus Torvalds gab sich damit nicht zufrieden. Das GNU-System war bis auf den Kernel vollständig, aber das Release des GNU-Kernels mit Namen HURD schien noch auf sich zu warten. Um die zeitliche Lücke bis dahin zu füllen, begann er selbst einen Kernel zu schreiben, der sehr rasch unter dem Namen Linux Verbreitung fand und eine große Entwickler- und Benutzergemeinde zusammenbrachte. Da die meisten Entwickler auf UNIXen arbeiteten, auf denen die GNU-Tools liefen, lag es nahe, den Linux-Kernel so einzurichten, dass er zusammen mit den GNU-Tools verwendet werden konnte: GNU/Linux. Der Kernel HURD ist über "akademische" Anfänge bislang nicht hinausgekommen, so dass das anfänglich als "Provisorium" gedachte Linux sich an seiner Stelle etabliert hat.

Zur gleichen Zeit löste sich BSD aus seiner ursprünglichen Abhängigkeit von AT&T: eine Gruppe von BSD-Leuten ersetzte alle Anweisungen im Quellcode, die noch von AT&T beigesteuert waren, durch neue und erstritt in einem langwierigen Gerichtsverfahren für BSD die Freiheit. Daraus gingen die Projekte FreeBSD, NetBSD und OpenBSD hervor, die auch eine beachtliche Verbreitung gefunden haben und manchmal als Linux-Vettern bezeichnet werden (und so manche Linux-Distribution enthält das ein oder andere "Schmankerl" aus einem der drei Projekte).

### **3 Installation:**

#### **3.1 Standardinstallation:**

Standardmäßig wird Linux via CD-Rom Laufwerk installiert. Dazu sind die ersten beiden CD's als Boot-CD's ausgeführt. Falls man über einen Rechner mit älterem BIOS (der nicht von CD's booten kann) verfügt, kann man auch eine Boot-Diskette erstellen, von der aus der CD-Zugriff erfolgen kann (unter DOS/Windows das Setup von der CD starten).



Seit 6.3 hat man die Wahl zwischen einer grafischen Oberfläche während der Installation (yast2, mind. 64MB Ram erforderlich!; Booten von CD1) und der bisherigen Installationsoberfläche yast1 (Textmodus, Booten von CD2)

Ab Linux 7.1 wird die Distribution auch auf einer DVD ausgeliefert. Falls man über eine DVD - Laufwerk verfügt ist diese Variante sicherlich zu bevorzugen, da kein Wechsel des Installationsmediums notwendig ist.

WICHTIG: Vor dem Beginn der Installation sollte man sich in der Hardware-Datenbank am Suse-Server ([www.suse.de](http://www.suse.de)) über die notwendige Treiberunterstützung informieren.

Da für das Seminar nicht so viele CD-Sätze wie notwendig vorhanden sind, wird hier ein anderer Weg gewählt. Für die Installation wird ein NFS (Network File System) – Server eingerichtet, auf dessen Festplatte der Inhalt der 5 Installations – CD's (in das Verzeichnis /suse) kopiert wurde. Über das Setup-Programm wurden außerdem eine Boot- und eine Moduldiskette erzeugt. Wenn die Rechner nun mit Diskette gestartet werden, gelangt man in das Installationsprogramm. Über den Menüpunkt << **Laden von Modulen** >> muss vor der eigentlichen Installation noch das Modul für die Unterstützung der eingebauten Netzwerkkarte geladen werden.

### 3.2 Installation via NFS:

Voraussetzung: funktionierende Netzwerkverbindung zum NFS-Server (Hardwareseitig und richtig konfiguriertes TCP/IP-Protokoll; Infos zu TCP/IP z.B.(bei installiertem Linuxrechner) [http://localhost/sdb/de/html/cep\\_ip\\_base.html](http://localhost/sdb/de/html/cep_ip_base.html) ; diese Seite ist auch am SuSE Webserver zu finden; weitere Infos auch am PI-Server: <http://www.pinoe-hl.ac.at> ). Die zur Installation notwendigen Disketten (Bootdisk, Moldules1 und Modules3) befinden sich als Images auf der ersten CD im Verzeichnis /boot. Falls der Rechner ein SCSI System, besitzt ist auch die Modules2 Diskette erforderlich. Zur Erzeugung der Disketten kann das ebenfalls auf der CD enthaltene Programm Raw-Write (im Ordner /dosutils) verwendet werden.

### 3.3 Tipps zur Installation

Bei der Installation sind einige Entscheidungen notwendig. Eine wichtige Entscheidung ist die Partitionierung der Festplatte. Eine Standardinstallation verwendet i.a. mindestens 3 Partitionen:

- /boot            kleine Partition für die Startdateien des Systems (sollten vor dem 1024. Zylinder liegen)
- swap            Partition für Auslagerungsdaten (hier verwendet Linux ein eigenes Dateisystem)
- /                Die Root- (Wurzel) Partition des Dateisystems

Für die Datenpartitionen stehen 3 unterschiedliche Dateisysteme zur Verfügung:

- Ext2:            Das Extended-2- Filesystem (bisher das Standarddateisystem unter Linux). Es unterstützt Benutzerplatzbeschränkung (Quotas). Ein Nachteil ist, dass bei einem Systemabsturz bei einem neuerlichen Systemstart erst ein e2fsck (FileSystemCheck) durchgeführt wird, der bei größeren Platten einige Minuten dauern kann.
- ReiserFS:       Ein Journaling-Filesystem. Erlaubt bei einem Systemabsturz einen schnellen Systemstart. Nachteil: Quotas sind nicht möglich.
- Ext3:            Vereinigt die Vorzüge beider Systeme.



## 4 Grundbegriffe (TCP/IP):

(Siehe auch PI-Weberserver → Grundlagen von IP und Netzwerken)

### 4.1 IP-Adresse: (Vers. 4)

Vergleichbar mit Telefonnummer (weltweit eindeutig!!), besteht aus 4 Byte (durch Punkte getrennt)

### 4.2 Netzwerkmaske:

Bitmaske, die mit der IP-Adresse mit „UND“ verknüpft wird, um Netzwerkanteil (~Vorwahl) und Hostanteil (~lokale Nummer) zu ermitteln.

255.255.255.0	C-Klasse
255.255.0.0	B-Klasse
255.0.0.0	A-Klasse

### 4.3 Adressbereiche:

- Klasse A:
  - 1. Bit = 0 → 0 – 126 .x.x.x
- Klasse B:
  - 1. Bit = 1, 2. Bit = 0 → 128 – 191.x.x.x
- Klasse C:
  - 1.Bit=1, 2.Bit=1, 3.Bit=0 → 192 – 223.x.x.x

#### private Adressbereiche:

- 1 Class A
  - 10.0.0.0 – 10.255.255.255
- 16 Class B
  - 172.16.0.0 – 172.31.255.255
- 256 Class C
  - 192.168.0.0 – 192.168.255.255

#### Loopback:

- 1 Class A
  - 127.0.0.0 – 127.255.255.255

### 4.4 Gateway:

Adresse des nächsten Routers am Weg ins Internet (wohin sollen alle Pakete geschickt werden, die nicht ins lokale Netz gehören)

### 4.5 DNS-Server:

Adresse des (der) Domain-Name-Servers (liefert die Übersetzung der Internetnamen in IP-Adressen)

## 5 NACHINSTALLATION:

Ab SuSE 8.0 (auch teilweise bei Version 7.x) empfiehlt sich sowohl Updates als auch Nachinstallationen über das entsprechende Modul des Konfigurationstools yast2 durchzuführen. Vor allem für Updates bietet das YAST YOU (Yast Online Update) komfortable Funktionen (eine schnelle Internetverbindung vorausgesetzt.)



## 6 LINUX als WORKSTATION:

Nach einer einfachen Installation von SuSE Linux (z.B. Standard mit Office) kann man aus dem Linuxrechner relativ einfach einen vollwertigen Arbeitsplatz machen. Auf der grafischen Oberfläche findet man Tools zur Konfiguration von Netzwerkdiensten (Anschluss über Netzwerkkarte, Einwahl via PPP) und zur Konfiguration von Druckern. Zu den Druckern wäre zu erwähnen, dass Unix-Systeme grundsätzlich PostScript-Drucker erwarten. Falls man einen Drucker verwendet, der kein PostScript versteht, muss über einen passenden Filter (aps-Filter) der Druckbefehl aufbereitet werden. Für die gängigsten Druckertypen findet man passende Druckerfilter als Teil der Distribution.

Das Paket OpenOffice wurde bei der Standardinstallation bereits installiert. Es ist pro Benutzer nur noch das Abarbeiten eines kleinen Installationsskripts notwendig. Für alle anderen Tätigkeiten findet man in der Distribution, oder im Internet passende Anwendungen zur Erledigung der Aufgaben.

### 6.1 Drucker unter Linux

Wie bereits erwähnt erwartet Linux einen postscriptfähigen Drucker. Um auch nicht Postscriptdrucker verwenden zu können verwendet man einen geeigneten Filter (vgl. Druckertreiber), der die Postscriptbefehle in ein für den Drucker verständliches Format umwandelt. Falls der Drucker lokal am Rechner angeschlossen ist kann Linux gängige Drucker automatisch erkennen. In der Schulumgebung sind die Drucker an Printserver angeschlossen, die man unter Linux direkt ansprechen kann. Dafür gibt es in der Druckerkonfiguration auch bereits passende Eingabemasken, um etwa die IP - Adresse des Servers und die Bezeichnung des Druckers am Server angeben zu können. Wie Drucker anderer Systeme (Windows) angesprochen werden können folgt etwas später im Skriptum.

Im Prinzip hat man nun die Auswahl zwischen 2 verschiedenen Drucksystemen: LPD und CUPS. Bei einer Standardinstallation wird nun CUPS installiert. Eine Ausnahmesituation, in der es notwendig ist auf das ältere LPD-System umzusteigen ist das Ansprechen einer Novell-PRINTQUEUE aus LINUX.

#### 6.1.1 LPD:

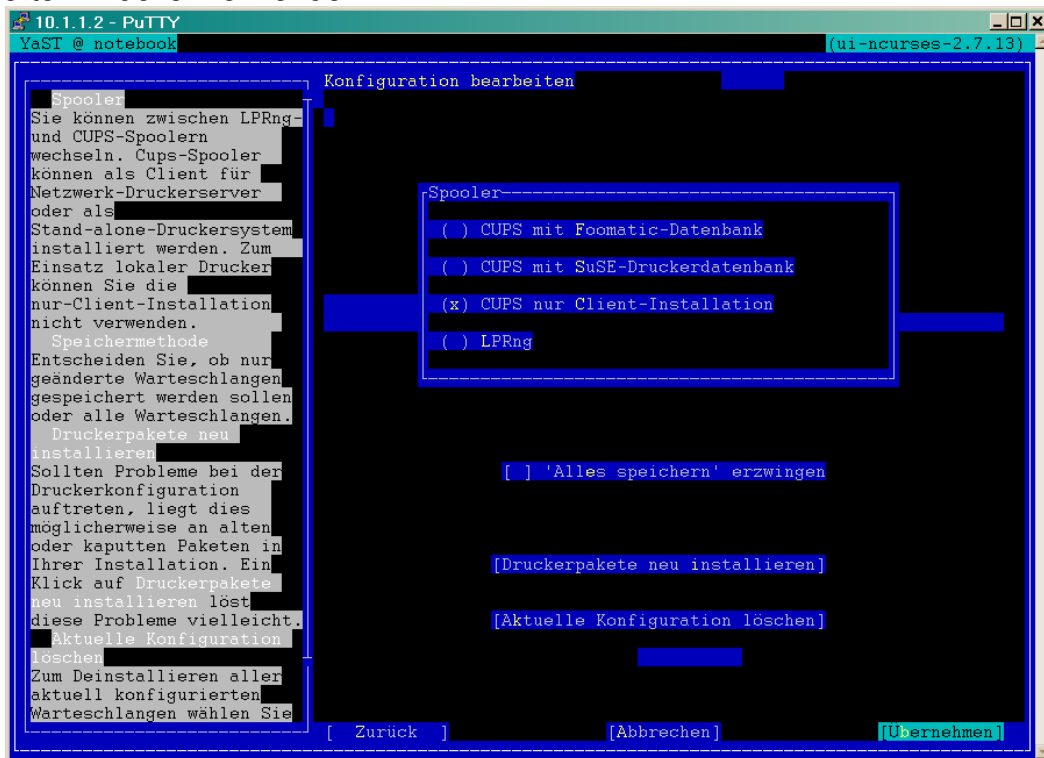
Das LPD (LinePrinterDämon) System erzeugt im allgemeinen Filterwarteschlangen, in denen die Postscriptbefehle des Systems in für den jeweiligen Drucker verständliche Befehle übersetzt werden. Wenn nun die Druckerwarteschlange (z.B lp ) am Rechner mit der IP-Adresse 10.0.2.10 freigegeben ist, können alle anderen Rechner des Netzwerkes in diese Warteschlange drucken. Dabei ist zu achten, das auf den Clientrechnern keine weitere Aufbereitung (Umwandlung) der Druckdaten erfolgen darf. (RAW-Warteschlange). Bei einer Umleitung in eine Novellwarteschlange ist zu beachten, dass von yast aus die Datei /etc/lpfilter/xxxxx/redirect (xxxxx steht für den Namen der lokalen Druckerwarteschlange) möglicherweise fehlerhaft erzeugt wird (wenn z.B. der Novellbenutzer kein Passwort besitzt) und händisch korrigiert werden muss.

#### 6.1.2 CUPS:

Bei Cups unterscheidet man zwischen Client und Serverkonfigurationen. Der Rechner, an den der Drucker direkt angeschlossen ist dient als Druckserver und muss daher als CUPS-Server konfiguriert werden. (dies ist die Standardinstallation!!). Die Konfiguration dieses Druckers erfolgt über das entsprechende YAST Modul. (Auch die Freigabe dieses Druckers für das Netzwerk).



Auf allen anderen Rechnern des Netzwerkes, die diesen Drucker verwenden wollen wird über YAST eingestellt, dass sie CUPS-Clients sind. In die entsprechende Maske trägt man nur noch die Adresse des CUPS-Servers ein. Damit können sie alle am CUPS Server definierten Drucker verwenden.



## 6.2 DER BOOTMANAGER:

### 6.2.1 LILO – Der Linux-Loader:

Lilo eignet sich auch als Bootmanager des Rechners bei verschiedenen Betriebssystemen. Er wird über die Datei /etc/lilo.conf konfiguriert. Die erste Konfiguration wird als Standardbetriebssystem verwendet. Nach einer Änderung in lilo.conf muss noch der Befehl lilo ausgeführt werden.

### 6.2.2 GRUB

Als Alternative zu LILO wird in den neueren Distributionen GRUB (GRand Unified Bootloader) installiert.

GRUB verwendet als Konfigurationverzeichnis den Ordner /boot/grub und wird in den MBR installiert. Grub verfügt über ein Kommandozeileninterface sowie eine Menüauswahl. Die Konfiguration kann über einen Editor oder über das Konfigurationstool YAST --> Konfiguration des Bootloaders erfolgen. Eine detaillierte Information erhält man über "info grub" in einem Shellfenster.

Beispieldatei /boot/grub/menu.1st

```
# Modified by YaST2. Last modification on Sun Jun 8 18:34:36 2003
```

```
color white/blue black/light-gray
default 1
gfxmenu (hd0,2)/boot/message
timeout 8
```



```

title linux
  kernel (hd0,2)/boot/vmlinuz root=/dev/hda3 vga=0x314 splash=silent showopts
  initrd (hd0,2)/boot/initrd

title windows
  root (hd0,0)
  chainloader +1

title floppy
  root (fd0)
  chainloader +1

title failsafe
  kernel (hd0,2)/boot/vmlinuz.shipped root=/dev/hda3 showopts ide=nodma apm=off
    acpi=off vga=normal nosmp noapic maxcpus=0 3
  initrd (hd0,2)/boot/initrd.shipped
    
```

Zu beachten ist, dass in der Grub-Konfiguration die Partitionen der Festplatte anders als üblich angesprochen werden: (hd0,0) bedeutet erste Festplatte (hd0) und darauf die erste Partition. Mit dem Kernelparameter wird auf eine bestimmte Datei verwiesen, die den Kernel enthalten sollte, der gebootet wird. Optional können hier gewisse Startparameter übergeben werden. Auch Betriebssysteme wie Windows oder OS/2, die eigentlich von GRUB nicht direkt unterstützt werden können mit diesem Bootsystem gestartet werden. Dazu dient der Parameter „chainloader +1“, der GRUB anweist in der jeweiligen Partition jeweils den nächsten Sektor anzusteuern und zu booten.

### 6.3 Wählverbindungen unter Linux:

In yast bzw. yast2 gibt es passende Module um PPP-Verbindungen und ISDN Verbindungen zu konfigurieren. Gängige Modems werden wie die Drucker automatisch erkannt. Die Verbindung kann auf Wunsch so konfiguriert werden, dass sie bei Bedarf automatisch aufgebaut wird. (→ NETZWERKSERVER!!) Dazu wird in der Datei /etc/rc.config die Variable PPPD\_DOD\_START auf „yes“ gesetzt.

Analog kann man auch bei ISDN –Verbindungen vorgehen.

Ab SuSE 8.0 empfiehlt sich eine Konfiguration über yast2 (auch Dial-on-Demand) kann hier entsprechend gesetzt werden.

## 7 SYSTEMGRUNDLAGEN:

### 7.1 Terminals:

Der Rechner bietet bei Standardkonfiguration an der lokalen Konsole mindestens 6 Terminals an. Ein Umschalten zwischen den einzelnen Konsolen erfolgt mit der Tastenkombination [Strg]+[Alt]+[Fx], wobei mit [Fx] eine der Funktionstasten gemeint ist.

Konsolen 1-6	Textkonsole
Konsole 7	Grafische Oberfläche
Konsole 10	Systemmeldungen (kein Login-Prompt)



### 7.2 Runlevel:

Das installierte Standardsystem kennt mehrere Betriebsstufen (RUNLEVELS)

RUNLEVEL	AUFGABE
0	„halt“ Anhalten des Systems
1	Betrieb ohne Netzwerkfunktionen
2	Betrieb mit allen Netzwerkfunktionen
3	Betrieb mit allen Netzwerkfunktionen und grafischer Oberfläche
6	„reboot“ Neustarten des Systems

Ein Wechsel zwischen den Betriebsmodi erfolgt mit dem Befehl „init x“ wobei x für einen gültigen Runlevelwert steht.

Ab Version 7.1 sind die Runlevel anders organisiert:

RUNLEVEL	AUFGABE
0	„halt“ Anhalten des Systems
1	Betrieb ohne Netzwerkfunktionen
2	Lokaler Betrieb mit remote-Netzwerk
3	Betrieb mit allen Netzwerkfunktionen
5	Betrieb mit allen Netzwerkfunktionen und grafischer Oberfläche
6	„reboot“ Neustarten des Systems

### 7.3 Benutzer / Gruppen:

Systembenutzer werden am einfachsten über yast → Administration des Systems → Benutzerverwaltung angelegt. Jedem Benutzer wird eine eindeutige numerische ID zugewiesen, mit der er im System identifiziert wird. Weiters kann dem Benutzer eine Login-Shell (STANDARD: /bin/bash), eine primäre Gruppenzugehörigkeit (STANDARD: users) und ein Home - Verzeichnis zugewiesen werden. Diese Informationen werden in die Datei „/etc/passwd“ eingetragen. Das Passwort wird aus Sicherheitsgründen verschlüsselt in der Datei „/etc/shadow“ gespeichert.

Analog kann man auch über den Menüpunkt Gruppenverwaltung eine Benutzergruppe anlegen. (Diese Definition wird in die Datei „/etc/group“ eingetragen).

Eine Sonderstellung hat der Benutzer „root“. Er ist vergleichbar mit dem Administrator unter NT bzw. dem Supervisor / Admin unter Novell. Für diesen Benutzer (übrigens mit der ID 0) gelten keine Einschränkungen. Daher muss diese Kennung besonders abgesichert werden (Passwort; darf sich nur von lokalen Konsolen einloggen (Ausnahme ssh); ftp-Zugriff ist nicht erlaubt).

Für die Verwaltung der Benutzer und Gruppen steht auf der KDE-Oberfläche auch der Usermanager zur Verfügung.

### 7.4 Das Dateisystem:

Unter Linux werden die verschiedensten Dateisysteme (auch der Speicher) in eine gemeinsame Wurzel („/“ root) hineingemountet. Als Trennzeichen zwischen den Verzeichnissen dient der Slash „/“. (Achtung: Linux unterscheidet zwischen Groß- und Kleinbuchstaben). Welche Dateisysteme beim Start unter welcher Bezeichnung („Mountpoint“) in das Dateisystem eingegliedert werden, steht in der Datei „/etc/fstab“.

Device	Bedeutung
/dev/hda	1. IDE –Platte (Master auf 1. Contr.)
/dev/hda1	1. Partition auf 1. Platte



/dev/hda2	2. Partition auf 1. Platte
.....	
/dev/hdb	Slave auf 1. IDE-Contr.
/dev/hdc	Master auf 2. IDE-Contr.
/dev/sda	1.SCSI-Platte
/dev/sda1	1. Partition auf 1. SCSI-Platte
/dev/sdb	2. SCSI-Platte

Bei einer Standardinstallation werden 3 Partitionen auf der Platte angelegt:

Mount-Point	Funktion
/boot	Enthält die Boot-Dateien des Systems (müssen vor dem 1024. Zylinder liegen)
	Swap-Partition
/	Root-Partition (enthält die Wurzel des Dateisystems)

Händisch den Zugriff auf Dateisysteme ermöglichen:

Diskette a: `mount /dev/fd0 /media/floppy` → Diskette ist unter `/media/floppy` im System erreichbar

CD: `mount /dev/hdc /media/cdrom` → CD-Rom ist unter `/media/cdrom` im System erreichbar

(Wenn das CD-RomLaufwerk als Master am 2. Controller angeschlossen ist)

**ACHTUNG: Veränderungen werden zunächst nur im Arbeitsspeicher abgelegt (schnellerer Zugriff) und erst später mit dem Inhalt des Datenträgers synchronisiert. Diese Synchronisation erfolgt spätestens beim Deaktivieren (umount) des Datenträgers. Dies ist vor allem für das Kopieren auf Disketten wichtig.**

**Hinweis zur aktuellen Version (SuSE9.1):**

**Mit der neuen Kernelversion (2.6.x) ist da händische Mouneten von CD's, DVD's, .. nicht mehr notwendig.**

**Weitere Informationen:**

Mountpoint: Ein (leerer) Verzeichniseintrag, über den via Mount-Befehl die Verbindung zu einem Dateisystem herstellt.

Wer darf Mouneten? Das Einhängen (mounten) von Dateisystemen ist üblicherweise dem root-Benutzer vorbehalten. Eine Ausnahme bilden jene Devices, die in der Datei `/etc/fstab` definiert werden und den Parameter `user` enthalten.

Was kann gemountet werden:

Lokale Systeme: `/dev/fd0` (Diskette), `/dev/hdc` (z.B. CD), `/dev/hda2` (Partition)

NFS-Platten: `10.0.1.1:/suse` (freigebener Ordner `/suse` auf `10.0.1.1`)

Unter Linux werden alle Geräte über Devices angesprochen. Devices sind definierte Schnittstellen zu der Hardware, wobei der direkte Zugriff auf die Hardware nur dem Kernel vorbehalten ist. Im Grunde sind Devices Dateien mit einer Inode, jedoch ohne Datenteil und sind im Verzeichnis `/dev` gespeichert.

Sie werden durch drei Informationen beschrieben: der Major Nummer, der Minor Nummer und dem Zugriffstyp. Die Major Nummer bestimmt den Linuxkerneltreiber, der für dieses Device verantwortlich ist. Hiervon gibt es zur Zeit 25 Treiber. Die Minor Nummer spezifiziert die Geräteart, z. B. 720 kB Floppy oder 1,44 MB Floppy, bei Festplatten gibt sie die Partition an, die verwaltet



werden soll. Beim Zugriffstyp gibt es zwei Arten, blockorientiert für gepufferte Geräte wie Festplatten und zeichenorientiert für ungepufferte Geräte wie serielle Schnittstellen. Zum Teil finden Sie im /dev-Verzeichnis auch Links. Diese werden mit einem beschreibenden Namen für die Vereinfachung auf Standardschnittstellen gelegt, z. B. für das Modem oder die Maus. Eine Übersicht über die häufig benötigten Schnittstellen ist in nachfolgender Tabelle aufgeführt.

**Tabelle:** Devicenamen und ihre Bedeutung (Auswahl)

Devicename	Gerät
*bm	Bus Mäuse
console	aktive Konsole
fd*	Diskettenlaufwerk
ftape*	Floppystreamer (ohne Rückspulfunktion) (Link)
hd*	IDE / ATA Festplatten oder CDROM
lp*	Parallele Schnittstellen
mem	Arbeitsspeicher
modem	Standardschnittstelle für das Modem (Link)
mouse	Standardschnittstelle für die Maus (Link)
null	unendlich großes Loch ("schwarzes Loch")
psaux	PS/2 Maus
nst*	SCSI Bandlaufwerk ohne Rückspulen
port	E/A-Schnittstellen
ptyp*	Terminalschnittstellen unter X (Master)
rmt	Bandlaufwerk (ohne SCSI)
sd*	SCSI Laufwerk
scd*	SCSI CD-ROM
st*	SCSI Bandlaufwerk mit automatischer Rückspulfunktion
tape*	Standardstreamer (Link)
tty*	Virtuelle Textterminals
ttyp*	Terminalslaves unter X
ttyS*	Serielle Schnittstellen

Die Sternchen in der Tabelle sind Platzhalter für eine nähere Bezeichnung, welches Gerät angesprochen werden soll. Hierfür gibt es zwei Notationen. Bei zeichenorientierten Geräten wird numerisch von ``0" an begonnen, die Geräte durchnummerieren. Z. B. ist die erste serielle Schnittstelle /dev/ttyS0 (unter DOS COM1). Bei blockorientierten Geräten gibt es zwei Methoden. Zum einen wird z. B. beim Diskettenlaufwerk oder SCSI CD-ROM die Spezifizierung wie bei zeichenorientierten Geräten vorgenommen, /dev/fd0 zeigt auf das erste Floppy. Bei Festplatten und CD-ROMs wird die Spezifizierung mittels Buchstaben vorgenommen, /dev/sda zeigt auf das erste SCSI Laufwerk, /dev/hda auf die Master-Festplatte am ersten IDE-Controller. Die einzelnen



Partitionen einer Festplatte werden über eine weitere Zahl beschrieben, hier jedoch wird mit der Nummerierung mit ``1" begonnen! Z. B. kennzeichnet /dev/hdc5 die erste logische Partition der Festplatte, die als Master am zweiten IDE-Port hängt (siehe auch , Seite ).

Der Systemverwalter kann weitere Devices anlegen, wenn die bereits durch die Installation vorhandenen Devices nicht ausreichen. Hierfür gibt es den Befehl mknod.

Syntax von mknod:

mknod [Optionen] Name Typ [Major Minor]

Typ:        b            legt blockorientiertes Device an  
           c, u        legt zeichenorientiertes Device an  
           p            legt eine FIFO an

### 7.5 Die Zugriffsrechte:

RECHT	Bezeichnung	Numerischer Wert
Lesen	r	4
Schreiben	w	2
Ausführen	x	1

Diese drei Zugriffsrechte werden nun 3mal vergeben:

Für den Besitzer der Datei:            owner

Für die Besitzergruppe der Datei:    group

Und für alle anderen:                others

Der Befehl zum Ändern der Zugriffsrechte lautet: „chmod“ . Der Befehl chmod 644 /etc/passwd weist der Datei /etc/passwd den Modus 644 zu. Dies bedeutet, dass der Besitzer die Datei lesen und schreiben (verändern) darf, die Gruppe und auch alle anderen dürfen diese Datei nur lesen. Eine ausführbare Datei wird durch das Setzen des Execute-Bits gekennzeichnet.

Auf der grafischen Oberfläche können die Zugriffsrechte wie unter Windows üblich über das Kontextmenü gesetzt werden.

Für eine feinere Steuerung von Zugriffsrechten kann mit so genannten Access Control Lists (ACL) durchgeführt werden. Diese werden vom neuen Kernel unterstützt (vorzugsweise auf EXT3 Dateisystemen) und bieten vor allem im Zusammenhang mit der Verwendung von Samba den Vorteil die Rechtesteuerung von Windows auf Linux zu übertragen. Unter Linux selbst fehlen bei vielen Tools noch die Unterstützungen für ACLs (z.B bei Konqueror).

### 7.6 Starten von Programmen:

Wenn eine Datei als „Ausführbar“ gekennzeichnet ist, kann sie aus einer Shell heraus gestartet werden. Entweder starten sie das Programm mit vollständigem Pfad, oder, wenn das Programmverzeichnis im Suchpfad enthalten ist, genügt ein Aufruf ohne Pfadangabe (z.B.: yast). Selbst wenn man mittels „cd Verzeichnis“ ins Programmverzeichnis gewechselt hat muss der Programmaufruf mit ./PROG\_NAME erfolgen. Wenn sie mehrere Kommandos nacheinander ausführen wollen können sie sie in einen Aufruf (durch ; getrennt) schreiben.

Sonderformen:

Ping 131.130.1.11&	Startet ein Ping zum angegebenen Host als Hintergrundprozess
Ping 131.130.1.11& >/dev/tty10	Startet den Hintergrundprozess und leitet die Ausgabe auf die Konsole 10 um
Nohup ping 131.130.1.11&	Dieser Prozess wird auch beim Ausloggen des Benutzers nicht beendet



**7.7 X-WINDOWS:**

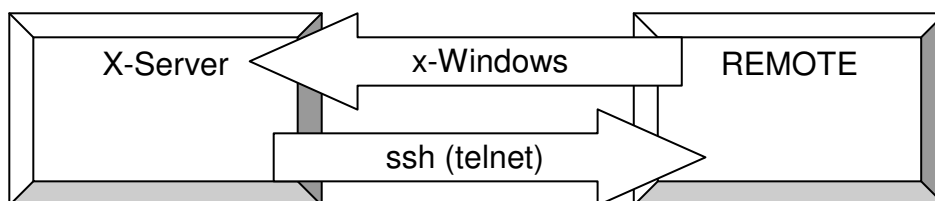
Bei einem laufenden X-Server (grafischer Oberfläche) ist es auch möglich anderen Rechnern zu gestatten X-Window-Programme auf dem (fremden) Rechner zu starten. Dies kann z.B. verwendet werden um mittels telnet oder ssh einen fremden Rechner zu übernehmen, ein Programm für die grafische Oberfläche zu starten und die Ausgabe dieses Programmes (wie kwrite oder kfmclient (KONQUERER)) auf den eigenen Rechner umzuleiten.

**HINWEIS:** Der Novellserver bietet die Möglichkeit (konfigurierbar über inetcfg an der Serverkonsole) einen Remotezugriff einzurichten. Neben der RCONSOLE, die nur das IPX-Protokoll verwendet kann auch eine X-Konsole eingerichtet werden. Damit ist von außerhalb eine telnet-Verbindung zum Server möglich, die Rückverbindung vom Server zum eigenen Rechner startet dann ein Fenster auf der X-Window-Oberfläche, von dem aus die Serverkonsole dann administrierbar wird. Da die Novellserver im allgemeinen jedoch nicht direkt im Internet sichtbar sind. Wäre für die Einrichtung einer solchen Verbindung noch eine spezielle Anpassung der Firewallregeln am Linuxrechner (wird im Teil 3 behandelt) notwendig.

**ERGÄNZUNG:** Eine einfachere Fernwartung des Novellserver erhält man jedoch, wenn man ein vt100 kompatibles Terminal verwendet. Dazu kann man unter Linux mit dem Befehl „export TERM=vt100“ in diesen Modus umsteigen. (Kontrolle mit „echo \$TERM“). Danach kann man mit einer Telnet-Verbindung zum Novellserver die Serverkonsole auf einen Linuxrechner umleiten (z.B.: Linux mit ssh übernehmen und dann vom Linuxrechner mit telnet die Serverkonsole übernehmen). Wenn ein direkter Zugriff auf den Novellserver möglich ist kann man z.B. auch mit dem Telnet von Windows 2000 die Konsole übernehmen:

- telnet (ohne Zielangabe)     startet die telnet-Sitzung
- set TERM vt100               schaltet in den VT100 Modus
- open 10.x.y.z                 verbindet zum Server
- quit                           schließt Telnet

**Siehe Grafik:**



**BEFEHLE:**

Starten eines Programmes mit Ausgabe auf einem anderen Rechner:

```
kwrite -diplay 10.0.1.1:0.0 &
```

wenn vorher der Zugriff auf den Rechner 10.0.1.1 mit xhost MY\_IP erlaubt wurde.

Xhost + schaltet die Zugriffskontrolle aus (alle dürfen)

xhost - schaltet die Zugriffskontrolle wieder ein

```
Oder export DISPLAY=10.0.1.1:0.0
```

```
kwrite
```

.....

- Xn.hosts        anlegen (steuert den Zugriff auf den Server n)
- xlsfonts        zeigt die inst. Fonts
- xfontsel        zeigt einen Probetext in der gew. Schriftart



```
xterm -geometry 70x35 -fn lucidasanstyewriter-bold-24 &
xterm -geometry +100+50 70x35 -fn lucidasanstyewriter-bold-24 // BS Position und Größe
```

.Xresources im jeweiligen Home Verzeichnis des Users. Dort können Sie Zeilen wie die folgenden eintragen:

```
XTerm*font: -misc-fixed-bold-r-normal--13-100-100-100-c-70-iso8859-1
XTerm*Background: bisque2
XTerm*Foreground: blue
XTerm*geometry: 90x40
```

**Starten einer weiteren Oberfläche**

```
/usr/X11R6/bin/X :1
BEFEHL -display localhost:1.0
bzw.: einen Fenstermanager auch auf der neuen Oberfläche starten:
export DISPLAY=localhost:1.0
/usr/X11R6/bin/windowmaker
```

Für die Verbindung zu einem X-Server wird der Port 6000+n verwendet, wobei n die Nummer des X-Servers ist. D.h. die erste X-Windows Oberfläche ist über Port 6001 erreichbar, die 2. über 6002 u.s.w.

**7.8 Hilfesystem:**

Am Linuxrechner gibt es mehrere Möglichkeiten Informationen zu installierten Diensten und Tools zu erhalten. Bei installierter grafischer Oberfläche (und gestartetem Webserver) kann man die Hilfedateien von Suse mit einem Webbrowser betrachten. (Diese Informationen kann man natürlich auch auf der SuSE-Homepage [www.suse.de](http://www.suse.de) betrachten.) Auf der Textoberfläche kann man mit dem Befehl „man DIENST“ die MAN-Page (manual) zu einem bestimmten Dienst ansehen. (z.B. man chmod). Falls man in diesen beiden Hilfesystemen nicht die passenden Informationen findet und man z.B. zum Dienst wuftp (ein FTP-Server unter Linux) Konfigurationsmöglichkeiten nachlesen will bietet sich (bei installiertem Dienst) noch ein Nachsehen im Verzeichnis /usr/share/doc/packages/wuftp an. In dieses Verzeichnis (/usr/share/doc/packages) werden zu jedem installierten Paket einige Informationsdateien abgelegt. Falls auch damit nicht die gewünschte Information zur Verfügung steht bleibt nur noch der Weg ins Internet.

**7.9 Wichtige Tools / Betriebssystembefehle:**

BEFEHL	BESCHREIBUNG
mc	Ruft den MidnightCommander (ein NortonCommander – Klon) auf. Damit kann man Dateien editieren, kopieren, verschieben,... , die Zugriffsrechte verändern, den Besitzer ändern,....
ls	List; zeigt den Inhalt eines Verzeichnisses
Befehl --help	Zeigt die Hilfe zu einem Befehl an (meistens)
man Befehl	Zeigt die Man(ual) – Page zu einem Befehl an. Wenn die grafische Oberfläche installiert ist (und einige Pakete aus der Serie doc) sind diese Manpages auch in HTML-Form verfügbar
ps	Zeigt eine Übersicht über die laufenden Prozesse
kill PID	Stoppt den Prozess mit der Nummer PID
cp A B	(„copy“) Kopiert A nach B
mv A B	(„move“) Verschiebt A nach B



## 7.10 Wichtige Verzeichnisse:

Verzeichnisse und Dateien, die mit einem „.“ beginnen, sind versteckte Dateien.

VERZEICHNIS	BEDEUTUNG
/boot	Bootdateien, Kernelimage
/media/cdrom	Vorbereitet für das Mounten von CD´s
/etc	Konfigurationsdateien
/etc/httpd	Konfigurationen für Apache
/etc/mail	Konfigurationen für Sendmail
/etc/sysconfig	Weitere Konfigurationsdateien (Firewall,...)
/media/floppy	Vorbereitet für das Mounten von Disketten
/home	Arbeitsverzeichnisse der Benutzer
/lib/modules/2.xxxx	Ladbare Module, wobei xxxx für die Kernelversion steht.
/root	Arbeitsverzeichnis des Root-Benutzers
/etc/init.d	Startskripte; Darunter die Links für die einzelnen Runlevels. Vor 7.1 findet man diese Dateien unter /sbin/init.d
/usr/bin	Betriebssystembefehle (für alle Benutzer)
/usr/share/doc/packages	Dokumentationen zu installierten Paketen
/srv/ftp	ftp-Server (anonymous-ftp)
/srv/www	WWW-Server
/usr/local/nwe	Root-Verzeichnis für Novellserver Emulation
/usr/src/linux	Quellen des Kernels
/var/lib/mysql	Datenbanken des SQL-Servers
/var/lib/named	Konfigurationsdateien des Domain Name Servers
/var/spool/mail	Mailverzeichnis (Eingehend)
/var/spool/mqueue	Mailverzeichnis (Ausgehend)
/var/squid/cache	Cache-Verzeichnisse des Proxyservers

## 7.11 Wichtige Dateien im Verzeichnis /etc und deren Bedeutung

DATEI	BEDEUTUNG
modules.conf	Optionseinstellung für das Laden von Modulen: z.B: Netzwerkkarten.
crontab	' cron.... Konfigurationsdateien für automatisches Starten von Diensten Zugeordnete Verzeichnisse: Cron.daily Cron.weekly Cron.monthly
default	Verzeichnis für Default Einstellungen: z.B Anlegen neuer Benutzer
dhcpcd.conf	Konfigurationsdatei für den DHCP-Server
exports	Dateisysteme die exportiert werden (via NFS – UNIX Netzwerk)
ftphosts	Einschränkungen (Hostseitig) von wo aus FTP-Zugriff erfolgen kann.
fstab	Wie werden die Dateisysteme gemountet.....
ftpaccess	Einstellungen für den FTP – Server
ftpusers	Diese Benutzer dürfen nicht mittels FTP auf den Server zugreifen
group	Benutzergruppen
host.conf	Reihenfolge der Dienste für Namensauflösung (hosts, bind)
HOSTNAME	Definiert den primären Namen des Rechners
hosts	Liste mit einer Übersetzung zwischen Name und IP – Adr.



hosts.allow	hosts... Dateien in denen der Zugriff von außen auf diesen Rechner eingeschränkt werden kann
hosts.deny	
hosts.equiv	
hosts.lpd	
httpd	in diesem Verzeichnis befinden sich die Konfigurationsdateien für den Web – Server
inetd.conf	Steuert einige wichtige Dienste: z.B ftp, telnet, Pop3 (Mail),
inittab	steuert z.B. den RunLevel des Systems
isapnp.conf	Konfiguration von ISA – PNP Karten (wird z.B mit dem Befehl: pnpdump >/etc/isapnp.conf erzeugt)
issue	Begrüßungstext für Login
issue.net	Begrüßungstext für Login aus dem Netz (Telnet, )
lilo.conf	Konfigurationsdatei für LiLo (LinuxLoader) (Bearbeiten über YAST)
login.defs	Login – Einstellungen
<b>mail/aliases</b>	Mailaliases (Textdatei)
mail/aliases.db	Mailaliases (Datenbank wird mit »newaliases« aus der Textdatei erzeugt )
motd	Tagestext (Anzeige nach LOGIN)
mtab	gemountete Filesysteme
named.conf	Konfigurationsdatei für Nameserver
networks	Netzwerkdefinitionen (Subnetze)
passwd	Definifion der Benutzer des Rechners (Textdatei)
permissions	Sicherheitseinstellungen für den Rechner
permissions.easy	(kann über Yast konfiguriert werden)
permissions.local	
permissions.paranoid	
permissions.secure	
printcap	Druckerdefinitionen (über Yast erstellen)
profile	Systemweite Einstellungsdatei (Suchpfade, Zeitzone,..), kann durch zusätzliche Benutzereinstellungsdateien (\$home/.bashrc) ergänzt werden
rc.config	Globale Einstellungen des Rechners (kann über yast od. einen Texteditor bearbeitet werden)
rc.config.d	Verzeichnis mit weiteren Ergänzungen zur Systemdatei rc.config (z.B.: firewall.rc.config, sendmail.rc.config)
route.conf	Einstellungen für Routing (Weiterleiten von Paketen)
sendmail.cf	Konfigurationsdatei des Mailservers (Sendmail) Auswahl der passenden Konfiguration über Yast
services	Übersicht Portnummer – Protokoll
shadow	Speichert die Paßwörter der einzelnen Benutzer (verschlüsselt)
shells	Auflistung der gültigen Kommandoshells
skel	Verzeichnis mit Dateien, die beim Anlegen des Benutzers automatisch in dessen Homeverzeichnis kopiert werden.
smb.conf	Konfigurationsdatei für Samba
smbpasswd	Speichert die PW der Samba – Benutzer (Windows – Clients)
squid/squid.conf	Konfigurationsdatei für SQUID (Proxy – Server)
ssh	Verzeichnis mit Konfigurationsdateien für die SecureShell



## 7.12 Der Startvorgang im Detail:

Wie bereits erwähnt unterscheidet Linux mehrere Betriebsstufen. Welche Dienste in welchem Level gestartet werden, ist relativ einfach nachzuvollziehen:

Im Verzeichnis /etc/init.d befinden sich die Startskripte für die unterschiedlichsten Dienste am Rechner. In den meisten Fällen wird am Beginn des Skriptes eine Systemvariable auf ihren derzeitigen Wert abgefragt. (z.B: START\_HTTPD). Wenn diese Variable z.B. auf „no“ gesetzt wurde wird das Skript beendet. Im anderen Fall wird der entsprechende Dienst gestartet.

```
mc - /etc/init.d
Left File Command Options Right
<- /etc/init.d > <- /etc/init.d/rc5.d >
Name Size MTime Name Size MTime
/rc5.d 4096 Jun 9 08:45 @K20atd 6 Jun 9 08:45
/rc6.d 4096 Jun 8 20:24 @K20fbset 8 Jun 9 08:45
/rc8.d 4096 Jun 9 08:45 @K20isdn 7 Jun 8 20:27
README 6960 Mar 18 18:36 @K2Orandom 9 Jun 8 20:24
*SuSEfire-2_final 1640 Mar 14 01:26 @K2Orpmco~igcheck 17 Jun 9 08:45
*SuSEfire-12_init 1625 Mar 14 01:26 @S01alsasound 12 Jun 9 08:45
*SuSEfire-2_setup 1851 Mar 14 01:26 @S01atd 6 Jun 9 08:45
*acpid 4445 Mar 14 01:29 @S01fbset 8 Jun 9 08:45
*alsasound 6084 Mar 17 17:58 @S01isdn 7 Jun 8 20:27
*apache2 7885 Mar 17 16:53 @S01random 9 Jun 8 20:24
*apmd 2706 Mar 17 14:55 @S01rpmco~igcheck 17 Jun 9 08:45
*atd 3653 Mar 14 01:31 @S02kbd 6 Jun 9 08:45
*autofs 8834 Mar 17 16:40 @S02splash 9 Jun 9 08:45
*boot 3877 Jan 20 15:33 @S05network 10 Jun 8 20:45
*boot.clock 2274 Aug 8 2002 @S06syslog 9 Jun 8 20:31
*alsasound -> ../alsasound
Hint: % macros work even on the command line.
linux:/etc/init.d #
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit
```

Für die einzelnen Runlevel befinden sich nun Unterverzeichnisse in /etc/init.d. Wenn nun z.B. im Ordner rc3.d ein Link auf das Skript „apache“ gelegt wird, wird der Dienst gestartet sonst nicht. Im Allgemeinen findet man jeden Link jedoch 2-mal. Einmal mit S..... und einmal mit K.... . Einer dieser Links wird ausgeführt beim Wechsel in diesen Runlevel (Start....) und der andere beim Verlassen des Runlevels (Kill....)

Im Unterschied zu den Vorversionen, wo die Links grundsätzlich vorhanden sind und der Wert einer Startvariable (definiert über rc.config) überprüft wird, werden die Links für die notwendigen Runlevel unter SuSE 8.0 angelegt bzw. entfernt. Dazu sollte man das Modul Runlevel-Editor aus yast2 verwenden.

## 8 ZUGRIFF AUF ANDERE DATEISYSTEME:

### 8.1 LINUX-DATEIZUGRIFF auf NOVELLSERVER

Sie können auch vorhandene Novell - Filesysteme in ihren Linux - Verzeichnisbaum einbinden. Dies erfolgt mit einem ncpmount-Befehl (enthalten im Pakte ncpfs)

```
ncpmount -S server -U user -P password -V volume MOUNTPOINT
z.B:
ncpmount -S fs411 -U guest -V vol1 /novell
```

Bevor sie auf ein Novell-Dateisystem zugreifen können müssen sie noch die IPX-Unterstützung im Kernel aktivieren. Mit dem Befehl

```
ipx_configure - -auto_interface=on - -auto_primary=on
```

wird das ipx-Routing richtig eingestellt. Danach sollten sie eine Verbindung zum Novellserver herstellen können.



## 8.2 WINDOWSDRUCKER von LINUX aus ansprechen

Für diese Option finden sie in YAST unter „Administration“ einen Punkt mit dem Namen Drucker im Windows-Netzwerk ansprechen. In der darauf folgenden Eingabemaske können sie nun die Druckerverbindung definieren. Sie erhalten beim Beenden der Maske einige Informationen, welche Drucker das System für sie angelegt hat.

Von Linux aus können sie mit dem Befehl „lpr“ Druckbefehle auslösen und so die Funktion des soeben definierten Netzwerkdruckers testen.

## 8.3 ZUGRIFF auf freigegebene Ordner von Windowsrechnern

Sie können mittels SAMBA nicht nur Ordner für Windowsrechner freigeben, sondern auch in umgekehrter Richtung auf (von Workstation freigegebene) Ordner zugreifen. Wenn sie z.B. in ihrer Arbeitsgruppe einen Rechner mit Netbiosnamen „PC“ und einem freigegebenen Verzeichnis mit der Bezeichnung „C\_PLATTE“ besitzen, können sie mit einem smbmount-Befehl diesen Ordner in ihren Verzeichnisbaum mounten:

```
smbmount //PC/C_PLATTE /windowsrechner
```

(Der Mountpoint /windowsrechner muss vorher bereits erzeugt worden sein)

Wenn sie das Einbinden vereinfachen wollen können sie die passenden Veränderungen in der Datei /etc/fstab vornehmen: z.B :

```
//PC/C_PLATTE /windowsrechner smbfs noauto,user 0 0
```

(Der Parameter noauto verhindert, dass der Ordner automatisch (beim Systemstart) aktiviert wird)

## 9 KONFIGURATION VON NETZWERKDIENTSTEN:

### 9.1 Installation von weiteren Netzwerkkarten:

Die Installation von Netzwerkkarten erfolgt in 2 Schritten:

#### 9.1.1 Aktivierung des passenden Treibers (Modul):

Zur Laufzeit kann das passende Modul mit dem Befehl „insmod MODULNAME“ geladen werden. Um diese Einstellung bei jedem Systemstart zu aktivieren, wird das entsprechende Modul in die Datei /etc/conf.modules eingetragen. Als Synonym für Netzwerkkarten wird die Bezeichnung ethx (Ethernet) gewählt, wobei x für 0,1,2,.. steht. Eth0 steht hier stellvertretend für die 1. Netzwerkkarte, eth1 für die Zweite, u.s.w.. Die Zuweisung erfolgt z.B. mit dem Eintrag „alias eth0 ne“ wenn es sich um eine ne1000 bzw. ne2000 kompatible Karte handelt. Welches Modul für welche Karte verwendet wird, bzw. ob die Karte überhaupt unterstützt wird, kann man in der Hardwaredatenbank am SUSE-Webserver nachschlagen. Bei Nicht-PNP-Karten müssen eventuell noch die passenden Hardwareparameter über eine Options-Zeile in /etc/conf.modules eingestellt werden. (z.B.: `options ne io=0x300,0x320 irq=5,7` )

Bei PCI-Karten ist i.a. keine besondere Einstellung notwendig. ISA-PnP-Karten können entweder mit entsprechenden DOS-Utilities auf einen Standardwert eingestellt werden (PnP-deaktivieren) oder mit dem Befehl „pnpdump >/etc/isapnp.conf“ für Linux konfiguriert werden. Das Programm pnpdump ermittelt die möglichen Einstellungen der Karte. Diese werden über die Pipe-Umleitung in die Datei /etc/isapnp.conf geschrieben. Durch Auskommentieren einer passenden Einstellung wird die Karte bei jedem Systemstart auf diese Parameter gesetzt. (Sofern der Parameter START\_ISAPNP in der Datei /etc/rc.config auf „yes“ gesetzt ist.)

#### 9.1.2 Einstellen der entsprechenden Adresse



Dieser Schritt kann am Einfachsten über das Konfigurationsprogramm „yast“ vorgenommen werden. Über die Punkte Administration des System → Netzwerk konfigurieren → Netzwerk Grundkonfiguration gelangt man in eine Maske, mit der die Netzwerkkarten mit IP-Adressen verbunden werden können. Mit [F5] kann man der entsprechenden Einstellung ein Device zuweisen (eth0, eth1,...). Mit [F4] wird diese Einstellung dann aktiviert. Die Skripte, die beim Aussteigen aus Yast abgearbeitet werden, tragen die Parameter in die notwendigen Dateien ein und konfigurieren auch das Routing (Paketweiterleitung) zwischen den Karten.

ERGÄNZUNG: Im Hintergrund wird einfach der Befehl „ifconfig“ mit passenden Parametern gefüttert. Mit dem Konsolenbefehl „ifconfig –a eth0 10.0.1.3 netmask 255.255.255.0“ wird z.B. der Karte mit der Bezeichnung eth0 die Adresse 10.0.1.3 zugewiesen.

### 9.1.3 Prüfen der Einstellungen:

Nach den vorgenommenen Änderungen starten sie den Rechner am besten neu („reboot“ als Befehl ausführen). Beobachten sie den Startvorgang der einzelnen Dienste am Monitor. Wenn sie anstelle der (grünen) Vollzugsmeldung „done“ am rechten Bildschirmrand eine (rote) Meldung „failed“ erhalten konnten gewisse Dienste nicht gestartet werden.

Die momentan aktiven Netzwerkeinstellungen können sie sich mit dem Befehl „ifconfig“ am Bildschirm anzeigen lassen. (Falls sie nicht alle Einstellungen am Bildschirm sehen können können sie sich mittels „ifconfig |more“ diese Information seitenweise anzeigen lassen.)

Überprüfen sie ob die Daten dem entsprechen, was sie zuvor in yast eingestellt haben.

## 9.2 Routing:

Wenn sie mittels yast ihre Netzwerkkarten definiert haben und auch das Gateway des Systems eingetragen haben, sollte das Konfigurationsprogramm die passenden Eintragungen in die Datei /etc/route.conf vornehmen. Diese Datei enthält die statischen Routen, die bei jedem Systemstart aktiviert werden. (Routen können auch zur Laufzeit durch Aufrufen des „route“ – Befehles mit passenden Parametern gesetzt werden. – ein „route“ ohne Parameter listet am Bildschirm die momentan aktiven Routen auf).

**ACHTUNG: Wenn der Rechner als Router fungieren soll, müssen sie in rc.config (bzw. über YAST → sysconfigEditor) den Parameter IP\_FORWARD auf yes setzen.**

```
193.170.207.128 0.0.0.0 255.255.255.224 eth0
```

Diese Zeile in der Datei route.conf definiert das Netzwerk, das an die Karte eth0 direkt angeschlossen ist. Dabei ist 193.170.207.128 die Netzwerkadresse, 255.255.255.224 die Netzwerkmaske und 0.0.0.0 das Gateway zu diesem Netz. 0.0.0.0 bedeutet, dass das Netzwerk direkt an eine der lokalen Netzwerkkarten angeschlossen ist (in diesem Fall eth0).

Einen Eintrag dieser Art müsste es für jede der definierten Karten geben.

```
default 193.170.207.135
```

bedeutet, dass alle Pakete, die keinem der anderen definierten Subnetze zuzuordnen sind an die Adresse 193.170.207.135 weitergeleitet werden. Das passende Subnetz (für ....135) muss natürlich bereits vorher definiert werden.

```
192.168.100.64 192.168.100.33 255.255.255.224
```



Diese Zeile definiert ein weiteres Subnetz, das sich hinter einem weiteren Router versteckt. Alle Pakete für das Netzwerk 192.168.100.64/255.255.255.224 werden an die Adresse 192.168.100.33 weitergeleitet (z.B. der Novellserver, der diese Pakete dann weiterverteilt.)

### 9.3 Testen der Netzwerkverbindungen:

Bevor nun versucht wird, weitere Netzwerkdienste aufzubauen, sollte man versuchen, ob die Netzwerkverbindungen auf dieser Ebene bereits funktionieren. Dazu stehen (auf praktisch allen Systemen) zwei Routinen zur Verfügung: (Bevor man jedoch diese Tests durchführt empfiehlt sich eine Kontrolle der Netzwerkeinstellungen, die mit dem Befehl **ifconfig** an der Konsole durchgeführt werden kann.)

#### 9.3.1 PING

ping 131.130.1.11: Schickt Pakete an die Adresse 131.130.1.11 und notiert die Antwortzeiten. Damit sollte man überprüfen, ob man die eingeschalteten Rechner in den Subnetzen, bzw. ob man Rechner im Internet erreichen kann. Sollten hier Fehler sichtbar werden, deren Ursache noch nicht ganz klar ist, kann man weitere Informationen mittels Traceroute erhalten.

#### 9.3.2 TRACEROUTE

traceroute 131.130.1.11: Versucht ebenfalls den Rechner mit der Adresse 131.130.1.11 zu erreichen, wobei auch Informationen über den Weg zu diesem Rechner geliefert werden (Über welche Router). Damit kann man den Weg nachvollziehen und erkennen, bei welchem Router das Problem entsteht.

### 9.4 Zuweisen von mehreren IP-Adressen zu einer Netzwerkkarte:

Im AHS-Bereich in NÖ ist es erwünscht (da der Rechner auf den Namen mail, www und ftp hören soll), dass der Karte in Richtung Internet 3 IP - Adressen zugewiesen werden. Für die weiteren Schritte wird vorausgesetzt, dass das World Device die Schnittstelle eth0 ist. Dieser Netzwerkkarte wurde im ersten Konfigurationsschritt bereits eine IP-Adresse zugewiesen. Im Programm yast können sie in der Netzwerkgrundkonfiguration weitere IP-Adressen zuweisen. Als Device geben sie bei der 2. IP-Adresse eth0:1 ein (wird damit auch der eth0 zugewiesen). Bei der nächsten Adresse verwenden sie eth0:2 u.s.w.

Nachdem sie mittels yast die Grundkonfiguration vorgenommen haben, müssen sie in der Datei /etc/route.conf das Routing am Linuxrechner nochmals nachkonfigurieren.

#### **Annahme: Offizielle Schuladressen: 193.170.207.128 / 255.255.255.248**

→ Router: 193.170.207.134 (=Gateway)

```
eth0      193.170.207.133
eth0:1    193.170.207.132
eth0:2    193.170.207.131
```

→ Einträge in route.conf:

```
.....
193.170.207.128  0.0.0.0      255.255.255.248    eth0
193.170.207.132  0.0.0.0      255.255.255.255    eth0:1
```

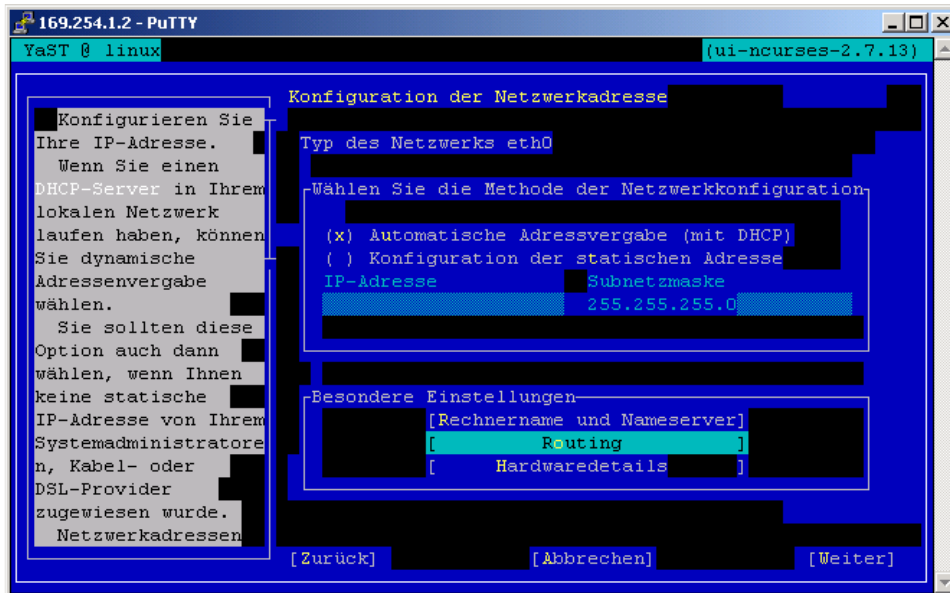


```
193.170.207.131  0.0.0.0      255.255.255.255      eth0:2
default        193.170.207.134
```

Testen Sie danach, ob sie von einer Workstation aus alle 3 IP-Adressen auf der eth0 anpingen können.

Ab SuSE 8.0 müssen sie diese Aufgabe anders lösen. Die Netzwerkeinstellungen befinden sich für jede Karte in einer eigenen Datei im Verzeichnis /etc/sysconfig/network. Die Datei für die Karte eth0 heißt ifcfg-eth0, die für eth1 ifcfg-eth1 u.s.w. Sollen nun einer Karte (z.B. eth0) mehrere Adressen zugewiesen werden, so kopiert man die Datei ifcfg-eth0 und benennt die Kopie ifcfg-eth0:1. Danach editiert man die Datei und weist ihr über die Parameter IPADDR, NETMASK, NETWORK,... die richtigen Werte zu.

Die Routingeinstellungen werden nun in der Datei /etc/sysconfig/network/routes eingetragen, es empfiehlt sich jedoch diese Einstellungen über yast → Netzwerkgeräte → Netzwerkkarte vorzunehmen:



## 9.5 Masquerade:

### 9.5.1 Bis SuSE 6.4:

Bevor man nun von z.B. Win9x-Rechnern aus den EDV-Räumen, ein Ping ins Internet durchführen kann ist noch ein weiterer Dienst zu konfigurieren: die MASQUERADE.

Erklärung: Laut Festlegung werden einige Adressbereiche für private Netzwerke reserviert. (z.B. 10.x.x.x; 192.168.x.x). Dies bedeutet, dass Pakete eines Rechners aus einem dieser Netze nicht im Internet weitergeleitet werden → keine Antworten von div. Servern aus dem Internet. Um dennoch privaten Netzwerken Zugang zum Internet zu ermöglichen, benötigt man einen Rechner mit einer gültigen IP-Adresse (World-Device) und Anschluss an das private Netzwerk (Internal Device). Ein Datenpaket aus dem internen Netzwerk wird nun vom Linuxrechner mit seiner gültigen Adresse maskiert. Die Antwortpakete werden dann ins interne Netz weitergeleitet. Dieser Dienst wird mit passenden Firewallbefehlen (ipchains) aktiviert und ist daher auch in der Firewallkonfigurationsdatei firewall.rc.config (im Verzeichnis /etc/rc.config.d) einzustellen. (Das Abarbeiten dieses Skriptes kann mit dem Parameter START\_FW in der allgemeinen Konfigurationsdatei rc.config eingestellt werden)



Wie im Kopf dieser Datei (firewall.rc.config) beschrieben, ist es für eine Masquerade mit Routing notwendig die Punkte 2), 3), 5), 6) und 9) entsprechend einzustellen.

FW\_DEV\_WORLD :

Bezeichnung der Netzwerkkarte die zum Internet zeigt (eth0, od. ppp0 (Modem), ipp0 (ISDN))

FW\_DEV\_INT: Bezeichnung der internen Netzwerkkarte(n) (eth1, ....)

FW\_ROUTE: Routing aktivieren (yes/no)

FW\_MASQUERADE: Masquerade aktivieren (yes/no)

FW\_MASQ\_NETS: Welche Netze sollen maskiert werden?

Welche Ports sollen freigeschaltet werden??  
„1: “ 1- (Alle) für alle 6 Parameter (reine Masquerade ohne Firewallfunktionalität)

Nachdem sie die Masquerade so definiert haben, starten sie den Rechner neu, oder wechseln sie mit „init 1“ in den Runlevel 1 und starten sie danach wieder (mit init 3) die Netzwerkdienste. Ab jetzt sollten sie mit einem Ping von internen Rechnern das Internet erreichen.

Wenn das nun möglich ist, können sie bei richtiger Netzwerkkonfiguration der Workstations bereits mit einem Browser ins World Wide Web einsteigen. (Bedingung: richtige Netzwerkkonfiguration, funktionsfähiger DNS-Server bei WS eingetragen: NÖ: 193.171.123.14)

### 9.5.2 Bis SUSE 7.1:

Hier werden für die Firewall die IPCHAINS verwendet (bis Kernel 2.2). Der Befehl zum Maskieren der internen Netze lautet:

```
Ipchains -A forward -j MASQ -i eth0
```

Wenn eth0 jene Karte ist, die zum Internet zeigt (WorldDevice). Damit nun dieser Befehl bei jedem Systemstart ausgeführt wird erstellt man in /etc/init.d eine Datei (z.B. mit der Bezeichnung masq\_start) die diesen Befehl enthält und legt in jene Runlevel, in denen die Masquerade gestartet werden soll einen Link auf diese Datei.

### 9.5.3 Ab SuSE 7.2:

Ab 7.2 wird der Kernel 2.4 als Standard installiert. Dieser Kernel sollte eigentlich als Firewallregeln mit den neueren iptables arbeiten. Beim SuSE-Kernel ist jedoch nach wie vor die ipchains-Unterstützung aktiviert, wodurch das alte Startskript weiterverwendet werden kann. Wie eine Masquerade mit den iptables konfiguriert wird, kann man im Teil 2 bei den Firewallregeln nachlesen.

### 9.5.4 Ab SuSE 8.0:

Ab 8.0 kann man die einfachen Firewallskripts von SuSE verwenden, die sich über yast2 konfigurieren lassen und auch eine Masquerademöglichkeit bieten. In Yast → Sicherheit und Benutzer → Firewall ist zunächst einzustellen, welche Netzwerkschnittstelle die Verbindung nach außen (Ext-Dev) herstellt, bzw. welche Karte für die Verbindung nach innen verantwortlich ist. Danach kann man einstellen, welche Ports (Dienste) von außen auf diesem Rechner erreichbar sein sollen. (eine genauere Definition der Firewall kann über die Datei /etc/sysconfig/SuSEfirewall2 erfolgen). Auf der nächsten Seite wird über die Option „Daten weiterleiten und Masquerading



durchführen“ die Masquerade eingestellt. Im letzten Formular wird nur noch eingestellt wie mit verworfenen Paketen umgegangen wird.

## 9.6 Web-Server:

Bei einer Standardinstallation wird der Webserver APACHE automatisch mitinstalliert und auch bereits gestartet (Einstellung über START\_HTTPD – Variable in rc.config). Bei konfigurierterem Netzwerk können sie von einer Workstation bereits mit einem Webbrowser unter <http://a.b.c.d> auf den Webserver des Linuxrechners zugreifen, wenn a.b.c.d stellvertretend für eine der Adressen des Linuxrechners steht. Wenn für diese Adresse bereits ein Rechner/Domainname vergeben ist und auch im entsprechenden Nameserver eingetragen wurde, kann der Zugriff auch über den Namen erfolgen.

Die Konfigurationsdatei des Servers ist /etc/httpd/httpd.conf.

Das Root-Verzeichnis des Webserver ist nun standardmäßig auf /srv/www (zuvor: /usr/local/httpd) eingestellt. Hier findet man u.a. 2 wichtige Verzeichnisse:

### cgi-bin:

Enthält ausführbare Dateien (\*.cgi). In der Konfigurationsdatei wird dieses Verzeichnis als Script-Verzeichnis definiert, wodurch ein Ausführen von Programmen via WEB in diesem Verzeichnis erlaubt wird. Wenn man in einem anderen Verzeichnis auch das Ausführen von Scripts erlauben will, muss die Option ExecCGI für diesen Ordner eingeschaltet werden.

### htdocs:

Das Wurzelverzeichnis für HTML-Dokumente. Beim Zugriff auf den Webserver wird zunächst nach der Datei index.html gesucht. Wenn Sie nun ihre eigene Homepage gestalten wollen, überspielen sie ihre Homepage (mit allen zugehörigen Dateien und Verzeichnissen) ins Verzeichnis /srv/www/htdocs, wobei die Startseite ihrer Homepage index.html heißen sollte. Benennen Sie vorher die bereits vorhandene Datei index.html z.B. in suse.html um. Damit können sie mit <http://a.b.c.d/suse.html> jederzeit auf die Suse-Dokumentation zugreifen.

Beachten Sie, dass alle Dateien ihrer Homepage mit dem Recht 644 (lesen für „others“) versehen werden und alle Verzeichnisse das Recht 755 (Ausführen/lesen für „others“) besitzen.

Es empfiehlt sich für die Wartung der Homepage einen eigenen Benutzer anzulegen, ihn als Besitzer aller Dateien unter /usr/local/httpd festzulegen. Mittels ftp-Zugriff kann dieser Benutzer dann die Homepage von anderen Rechnern aus warten.

Weitere Tipps und Tricks zu den Wartungsaufgaben finden sie am PI-Webserver (<http://www.pinoe-hl.ac.at>) unter AG-Informatik → Anbindung eines Netzes ans Internet → WWW-Server Apache.

### 9.6.1 Verwendung von PHP:

Das folgende Beispiel demonstriert die Verwendung von PHP in HTML Seiten. Damit die Seite richtig interpretiert wird muss die Datei die Endung .php besitzen. Damit diese Datei (index.php) vor einer eventuell vorhandenen Datei index.html verwendet wird, muss der Parameter DirectoryIndex in der Datei /etc/httpd/httpd.conf angepasst werden. Für die Funktion der nachfolgenden Datei, die einen Zähler realisiert, muss im selben Verzeichnis eine Datei counter.txt existieren, die für die Gruppe others auch Schreibrechte hat.

<html>



```
<head>
  <title>Homepage von USER</title>
  <meta content="">
  <style></style>
</head>
<body>
Dies ist eine Testseite für Benutzerhomepages <br>
<?PHP
  $fp = fopen("counter.txt","r");
  $zahl = fgets($fp,10);
  fclose($fp);
  $zahl++;
  echo "Sie sind der ".$zahl.". Besucher"; // Strings werden mit „.“ zusammengefügt
  $fp = fopen("counter.txt","w");
  fputs($fp,$zahl);
  fclose($fp);
?>
</body>
</html>
```

### 9.6.2 Zugriff auf bestimmte Webseiten mittels Authentifizierung:

Man kann einzelne Verzeichnisse am Webserver sperren, sodass ein Zugriff nur mit gültiger Benutzerauthentifizierung möglich ist. (Siehe Dokumentation PI-Server → Web-Zugriff auf Verzeichnisse sperren)

Dazu muss eine eigene Passwortdatei angelegt werden:

z.B.: mit dem Befehl: "htpasswd -c /usr/local/httpd/allowed.users lehrer" eine Passwortdatei z. B. für den Benutzer **lehrer** anlegen. Danach erfolgt eine zweimalige Abfrage eines Passwortes. Dieses Passwort wird dann verschlüsselt in der Datei **allowed.users** abgespeichert. In dem Verzeichnis, das gesperrt werden soll (z. B. **/usr/local/httpd/htdocs/geheim**) wird mit einem Editor eine Datei mit dem Namen **.htaccess** angelegt. In diese Datei wird folgendes geschrieben:

```
AuthName "Verzeichnis geheim"
AuthType Basic
AuthUserFile /usr/local/httpd/allowed.users
require user privat
(oder require valid-user → Zugriff kann mit jeder gültigen Benutzerkennung erfolgen)
```

Bei den Standardeinstellungen des Apache-Servers werden die Einstellungen der Datei **.htaccess** ignoriert. Es muss daher für das Verzeichnis **/usr/local/httpd/htdocs/geheim** diese Möglichkeit aktiviert werden. Dazu editiert man die Datei **/etc/httpd/httpd.conf** und fügt für dieses Verzeichnis eine eigene Definition ein:

```
<Directory "/usr/local/httpd/htdocs/geheim">
AllowOverride All
</Directory>
```



Wenn man weitere gültige Benutzerkennungen zur Passwortdatei hinzufügen will, so kann dies mit dem Befehl

“htpasswd /usr/local/httpd/allowed.users NocheinBenutzer“ (d.h. ohne –c) erfolgen.

## 9.7 Fernwartung:

Der Linuxrechner erlaubt ein Login auch von nicht-lokalen Konsolen. Stellvertretend seien hier 4 Möglichkeiten dafür genannt:

### 9.7.1 TELNET:

Wenn der Telnet-Dämon (Teil des inetd; inetd.conf bzw. neuerdings xinetd) gestartet ist, wartet er am Port 23 (nachzulesen in der Datei /etc/services) auf Anmeldungen via Netzwerk. Aus Sicherheitsgründen ist es dem Root-Benutzer nicht erlaubt direkt eine Telnetverbindung herzustellen. Um trotzdem Wartungsarbeiten durchführen zu können kann man folgende Lösung praktizieren:

Mit telnet a.b.c.d öffnet man eine Telnet-Verbindung (z.B. von einer Windows-WS aus) und loggt sich unter einer gültigen Benutzerkennung ein. Mit dem Befehl „su root“ (SwitchUser) wechselt man in die Root-Identität (Root-PW erforderlich!!) und kann nun alle Wartungsarbeiten am System durchführen. Der Nachteil der Wartung mittels Telnet ist, dass die Daten (auch die Passwörter!!) unverschlüsselt übers Netz (Internet) verschickt werden und somit einem eventuellen Mitlauscher direkt zur Verfügung stehen. Günstiger ist es daher, die Wartung über eine SecureShell (ssh) durchzuführen.

### 9.7.2 SSH (Secure Shell):

Hier werden die Daten mittels RSA-Algorithmus verschlüsselt. Der einzig unsichere Moment ist das Übertragen des Schlüssels (wenn dies über Internet geschieht). Wenn man das nicht will, kann man mittels Diskette, Abschreiben,.. den Public-Key des Fernzuwartenden Rechners in z.B: die Datei /root/.ssh/known\_hosts eintragen. (Wenn die Wartungsarbeit am anderen Rechner auch als Root-Benutzer durchgeführt wird).

Die Konfiguration des SecureShellDämons (sshd) erfolgt über die Datei /etc/ssh/sshd\_config. Es empfiehlt sich (aus Sicherheitsgründen) über den Parameter Protocol nur die Version 2 des ssh-Protokolls zuzulassen.

Als Client-Programm kann man von einem Linuxrechner aus die Kommandozeilenutility ssh verwenden. Von Windowsrechnern aus bietet sich das Programm putty an, das man z.B von [www.tucows.at](http://www.tucows.at) downloaden kann.

### 9.7.3 FISH:

Von einem Linux-Desktop mittels Web\_Browser Konqueror: in die Adresszeile folgende URL eingeben: fish://ip\_server Damit erfolgt eine SSH (bzw. SCP) Zugriff auf den Rechner ip\_server. Über den Browser kann man nun auf Dateien des entfernten Rechners zugreifen und diese direkt editieren und verändern.

### 9.7.4 Wartung mit WEBMIN:

Das Tool Webmin (<http://www.webmin.com>) ermöglicht eine weitestgehende Wartung des Rechners über einen Webbrowser. Dazu kann man das Paket als \*.rpm – Paket herunterladen und via yast installieren. Über dieses Tool kann das System und nahezu alle Serverdienste



gesteuert werden. Es ermöglicht außerdem das Anlegen einer Benutzerstruktur für die Webmin – Oberfläche.

## 9.8 FTP-Server:

Als Teil des Inetd-Dämons wird auch ein FTP-Server gestartet. (siehe entsprechende Zeile in inetd.conf). Man kann hier zwischen 3 verschiedenen Servern auswählen (sofern die entsprechenden Pakete installiert sind):

```
In.ftpd
Wu.ftpd
Proftpd
```

Verwenden Sie die Zeile in der wu.ftpd gestartet wird und kommentieren sie die beiden anderen Zeilen aus. Weitere Konfigurationsmöglichkeiten zum jeweilig verwendeten Server können sie sich über die entsprechende Manpage anzeigen lassen. Über die Datei /etc/ftphosts kann man den ftp-Zugriff einschränken (z.B. einen Benutzer auf das lokale Netzwerk beschränken)

```
/etc/ftppass
allow web 193.170.207.* 192.168.*
allow ftp *
allow guest *
```

Diese Einstellung erlaubt dem Webbenutzer den ftp-Zugriff nur aus der Schuldomäne und dem internen Netz, die Benutzer ftp und guest sind jedoch nicht eingeschränkt.

Zugeordnete Dateien:  
ftppass; ftphosts; ftpusers

Ab SuSE 8.0 wird als Standard vs.ftpd verwendet. Dieser Dienst wird über die Datei vsftpd.conf konfiguriert.

## 9.9 Benutzerplatzbeschränkung:

Bei einer größeren Anzahl von Systembenutzern ist es notwendig, dass der, den Benutzern zur Verfügung stehende Plattenplatz beschränkt wird. Dies kann mit den sog. QUOTAS erfolgen.

Wenn sie das Paket QUOTA (Serie ap1) installiert haben, müssen die Beschränkungen noch konfiguriert und aktiviert werden. Dazu legen sie im Wurzelverzeichnis des Dateisystems, auf dem sie Beschränkungen verwenden wollen die Datei "quota.user" an. Für die Standardinstallation in NÖ ist dies eigentlich nur für die Root-Partition sinnvoll und kann z.B. mit dem Befehl "touch /aquota.user" durchgeführt werden (als root-Benutzer). Weisen sie dieser Datei mit "chmod 600 aquota.user" die passenden Filerechte zu.

Editieren sie nun die Datei "/etc/fstab". In dieser Datei finden sie eine Zeile, in der die Optionen für das Mounten der Root-Partition stehen: (Quotas sind nur in z.B:

```
/dev/hda2 / ext2 defaults, usrquota .....
```

Fügen sie in dieser Zeile die Option usrquota hinzu. Diese Option aktiviert die Quotas auf der entsprechenden Partition.



Speichern sie die Änderungen ab und editieren sie danach die Datei "/etc/rc.config". Setzen sie in dieser Datei den Parameter START\_QUOTA auf yes. Jetzt müssen die bereits vorhandenen Dateien ihren Besitzern zugeordnet werden. Die erste Initialisierung wird mit dem Befehl quotacheck -acuvqm durchgeführt, danach starten sie den Rechner neu.

Nach dem neuerlichen Start des Rechners werden die Quotas aktiviert. Mit dem Befehl "quota *Benutzername*" können die Quotas eines Benutzers eingesehen werden. Zum Einstellen von Beschränkungen kann der Befehl "edquota *Benutzername*" verwendet werden. Sie können im Editor (vi) nun für den jeweiligen Benutzer ein Softlimit (darf einige Zeit überschritten werden) und ein Hardlimit einstellen. (Im vi speichert man mit ":w" ; Verlassen: ":q"). Für eine größere Anzahl verwendet man einen Beispielbenutzer und macht die anderen äquivalent zu diesem. (kann mit edquota gemacht werden)

Hilfe erhalten sie über die entsprechende Manpage. Quotas können übrigens auch mit den PSNTOOLS eingestellt werden.

Eine einfache Überprüfung und Einstellung der Quotas kann auch über das WEBMIN – Interface erfolgen.

## 9.10 WARTUNG VON BENUTZERDATEN

### 9.10.1 WEBMIN-USERINTERFACE

Man kann auch über das WEBMIN-Interface Benutzer aus eine Steuerdatei anlegen und löschen. Diese Variante ist durch die unsaubere Lösung der PSN-Tools bei neueren Distributionen vorzuziehen.

```
create:username:passwd:uid:gid:realname:homedir:shell:min:max:warn:inactive:expire
```

```
modify:oldusername:username:passwd:uid:gid:realname:homedir:shell:min:max:warn:inactive:expire
```

```
delete:username
```

*In create lines, if the uid field is left empty, Webmin will assign a UID automatically. If the gid field is empty, Webmin will create a new group with the same name as the user. The username, homedir and shell fields must be supplied for every user - all other fields are allowed to be empty. If the passwd field is blank, no password will be assigned for the user. If it contains just the letter x, the account will be locked. Otherwise, the text in the field will be taken as the cleartext password and encrypted.*

*In modify lines, an empty field will be taken to mean that the corresponding user attribute is not to be modified.*

## 9.11 Absichern des Systems

Grundregel: Alle Dienste die nicht benötigt werden, sollten nicht gestartet werden!!

Überprüfen sie die Dateien rc.config und inetd.conf auf eventuell nichtverwendete und derzeit trotzdem gestartete Serverdienste (finger, identd, ...)

Besuchen Sie in regelmäßigen Abständen die Suse-Homepage (Donwload → Patches → Version xx). Falls von einem Programmpaket eine neue Version verfügbar ist, die einen Fehler (Sicherheitslücke) des vorhergehenden Paketes löst, ist dies meist mit der Anmerkung „Update empfohlen“ versehen.

Erlauben sie die Fernwartung (z.B. der Homepage) nur von bestimmten Rechnern. Es ist bereits bei Schulen vorgekommen, dass die Schulhomepage durch einen Hackereinbruch verunstaltet wurde.



## 10 Prozessmanagement

Das gesamte Management der Prozesse unterliegt dem Kernel. Er verteilt die Rechenzeit, überwacht und ermöglicht die saubere Kommunikation unter den Prozessen und kontrolliert die Rechte. Linux ist ein Multitasking-System, was bedeutet, dass mehrere Prozesse gleichzeitig abgearbeitet werden können. Dies kann aber nur auf Mehrprozessormaschinen wirklich erfolgen, auf Single-Prozessor-Systemen wird ein System benötigt, das dem Anwender scheinbar die parallele Abarbeitung der Prozesse vorspielt. Hier kommt der Prozess-Scheduler zum Einsatz. Er verteilt die Rechenzeit und Systemressourcen auf die einzelnen Prozesse für ein bestimmtes Zeitintervall.

Der Mehrprogrammbetrieb steigert den Informationsdurchfluss der Hardware. Stoppt z.B. Programm n, weil es einen E/A-Kanal aufruft, so verarbeitet der Prozessor das bereitstehende Programm m, bis dieses einen Stop erreicht usw. Ein Zuteilungsprogramm legt zuvor die Prioritäten fest.

Steht jedem im Hauptspeicher befindlichen Programm eine feste Zeitspanne zu, so spricht man von Zeitscheibenverfahren (engl.: time slicing)." [Breuer, Seite 197]

Welcher Prozess als nächster für die Bearbeitung durch die CPU ausgewählt wird, errechnet der Prozess-Scheduler mit Hilfe von Prioritätsparametern, Prozesszustand und Wartezeit. Auf die Priorität von Prozessen kann auch direkt Einfluss genommen werden über den Befehl nice. Normale Anwender können Prozesse herunterstufen, allein dem Systemadministrator root ist das Recht vorbehalten, Prozesse heraufzustufen. Der Aufruf erfolgt über:

```
nice -<nice-Wert> <Kommandofolge>
```

Für den nice-Wert kommt ein Zahlenbereich von -20 bis 20 in Frage, wobei ein niedriger nice-Wert eine höhere Priorität bedeutet. Der in der Syntax angegebene Bindestrich gehört zum Befehl dazu, so dass bei heraufsetzen der Priorität zwei Bindestriche einzugeben sind. Nachträglich, wenn der Prozess schon gestartet ist, kann die Priorität mit dem Befehl renice geändert werden. Die Syntax hierfür lautet:

```
renice -<nice-Wert> <PID>
```

Der PID ist eine eindeutige Identifizierung eines Prozesses (Prozess Identifier). Jedem Prozess wird eine eindeutige Nummer zugewiesen. Wie schon in Kapitel 1 erwähnt, hat der Vater aller Prozesse, der init-Prozess, die PID 1.

Ein Prozess besteht immer aus einem Textsegment (Befehle), einem Datensegment und einem Stack-Bereich. Wird nun ein Befehl oder Kommando zwei oder mehrfach gestartet, so wird das Textsegment nicht noch einmal geladen, sondern im Speicher nur Bereiche für das Datensegment und den Stack-Bereich reserviert. Somit können die Speicherressourcen geschont werden.

Im Verzeichnis /proc befindet sich ein Abbild aller Prozesse sowie alle Informationen des Kernels. Es ist ein spezielles Dateisystem mit keinen echten Dateien und Verzeichnissen, sondern bildet die Schnittstelle zum Kernel. Hier kann der direkte Zugriff auf Informationen des Kernels realisiert werden. Bei Arbeiten in diesem Verzeichnis ist größte Sorgfalt angebracht, da Änderungen an den Dateien direkt den Kernel betreffen und das System in einen instabilen Zustand versetzen können, bis zum Absturz des Systems. Trotz allem ist dieses Verzeichnis eine der wichtigsten Informationsquellen zum System. Im nachfolgenden Listing ist ein Auszug aus dem Verzeichnis /proc wiedergegeben:

```
dr-xr-xr-x 3 root  root      0 Jul 10 08:51 bus
-r-r-r-  1 root  root      0 Jul 10 08:51 cmdline
-r-r-r-  1 root  root      0 Jul 10 08:51 cpuinfo
-r-r-r-  1 root  root      0 Jul 10 08:51 devices
-r-r-r-  1 root  root      0 Jul 10 08:51 dma
-r-r-r-  1 root  root      0 Jul 10 08:51 filesystems
dr-xr-xr-x 2 root  root      0 Jul 10 08:51 fs
```



```

dr-xr-xr-x 4 root  root      0 Jul 10 08:51 ide
-r-r-r- 1 root  root      0 Jul 10 08:51 interrupts
-r-r-r- 1 root  root      0 Jul 10 08:51 ioports
-r---- 1 root  root 134221824 Jul 10 08:51 kcore
-r---- 1 root  root 134221824 Jul 10 08:51 kcore_elf
-r---- 1 root  root      0 Jul 10 08:40 kmsg
-r-r-r- 1 root  root      0 Jul 10 08:51 ksyms
-r-r-r- 1 root  root      0 Jul 10 08:51 loadavg
-r-r-r- 1 root  root      0 Jul 10 08:51 locks
-r-r-r- 1 root  root      0 Jul 10 08:51 meminfo
-r-r-r- 1 root  root      0 Jul 10 08:51 memstat
-r-r-r- 1 root  root      0 Jul 10 08:51 misc
-r-r-r- 1 root  root      0 Jul 10 08:51 modules
-r-r-r- 1 root  root      0 Jul 10 08:51 mounts
dr-xr-xr-x 3 root  root      0 Jul 10 08:40 net
-r-r-r- 1 root  root      0 Jul 10 08:51 partitions
-r-r-r- 1 root  root      0 Jul 10 08:51 pci
dr-xr-xr-x 2 root  root      0 Jul 10 08:51 scsi
-r-r-r- 1 root  root      0 Jul 10 08:51 sound
-r-r-r- 1 root  root      0 Jul 10 08:51 stat
-r-r-r- 1 root  root      0 Jul 10 08:51 swaps
dr-xr-xr-x 9 root  root      0 Jul 10 08:51 sys
dr-xr-xr-x 4 root  root      0 Jul 10 08:51 tty
-r-r-r- 1 root  root      0 Jul 10 08:51 uptime
-r-r-r- 1 root  root      0 Jul 10 08:51 version
    
```

Zum Beispiel enthält die Datei /proc/cpuinfo Informationen über die CPU, die Datei /proc/modules Informationen, welche Module geladen sind und die Datei /proc/meminfo Informationen über Speicherverbrauch und -ressourcen.

### 10.1 Prozesse anzeigen

Alle laufenden Prozesse können über die Befehle ps, top oder pstree angezeigt werden. Dabei gibt der Aufruf von ps viele wichtige Informationen über das System. Das nachfolgende Listing zeigt eine gekürzte Ausgabe des Befehls ps -aux, die Parameter werden in Kapitel 10 beschrieben.

```

USER      PID %CPU %MEM    RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.1  204 ?        S   12:28   0:04 init [2]
root         2  0.0  0.0    0 ?        SW  12:28   0:00 [kflushd]
root         3  0.0  0.0    0 ?        SW  12:28   0:00 [kupdate]
root         4  0.0  0.0    0 ?        SW  12:28   0:00 [kpiod]
root         5  0.0  0.0    0 ?        SW  12:28   0:00 [kswapd]
bin         94  0.0  0.3  400 ?        S   12:29   0:00 /sbin/portmap
root       101  0.0  0.2  360 ?        S   12:29   0:00 /usr/sbin/scanlogd
root       107  0.0  0.4  632 ?        S   12:29   0:00 /usr/sbin/syslogd
root       185  0.0  0.4  560 ?        S   12:29   0:00 /usr/sbin/inetd
root       200  0.0  0.4  572 ?        S   12:29   0:00 /usr/sbin/lpd
root       231  0.0  0.4  628 ?        S   12:29   0:00 /usr/sbin/cron
nobody    235  0.0  0.5  692 ?        S   12:29   0:00 /usr/sbin/in.identd -e
root     236  0.0  0.5  692 ?        S   12:29   0:00 /usr/sbin/in.identd -e
    
```



```

root    270  0.0  0.7 1016 tty1   S   12:29  0:00 login - root
root    271  0.0  0.7 1016 tty2   S   12:29  0:00 login - tobias
root    272  0.0  0.3  428 tty3   S   12:29  0:00 /sbin/mingetty tty3
root    273  0.0  0.3  428 tty4   S   12:29  0:00 /sbin/mingetty tty4
root    274  0.0  0.3  428 tty5   S   12:29  0:00 /sbin/mingetty tty5
root    275  0.0  0.3  428 tty6   S   12:29  0:00 /sbin/mingetty tty6
root    301  0.0  1.0 1328 tty1   S   12:30  0:00 -bash
tobias  345  0.0  1.0 1328 tty2   S   12:30  0:00 -bash
tobias 2784  0.0  1.5 1545 tty2   R   12:34  0:00 find
root    2785  0.0  0.7  928 pts/1  R   12:40  0:00 ps -aux

```

Dabei bedeuten die Spalten im einzelnen

[USER:]der aufrufende Benutzer des Prozesses;

[PID:]eindeutige ID eines Prozesses;

[RSS:](Resident Set Size) Größe des Prozesses im Speicher in Kilobyte;

[TTY:]kontrollierender Terminal (ein ? bedeutet, daß diesem Prozess kein Terminal zugeordnet ist);

[Stat:]Laufzeitzustand des Prozesses;

[R:]Prozess läuft im Moment;

[S:]Prozess "schläft", befindet sich also im Wartezustand;

[D:]Prozess "schläft" und kann nicht zwingend unterbrochen werden;

[T:]Prozess ist gestoppt;

[Z:]Prozess ist ein Zombie;

[Time:]Bisher verbrauchte Prozessor-Zeit des Prozesses;

[Command:]Name des Kommandos, mit dem der Prozess erzeugt wurde;

Der Befehl top gibt dieselben Informationen wie der Befehl ps, jedoch werden einige Zusammenfassungen und Gesamtspeicherinformationen zusätzlich angezeigt. pstree zeigt einen Prozessbaum an, wobei gleiche Kommandos und Befehle zusammengefasst werden und die Anzahl dieser als Multiplikator vorangestellt wird. Das nachfolgende Listing gibt eine gekürzte Ausgabe des Befehls pstree wieder:

```

init+-atd
|-cron
|-httpd--httpd
|-in.identd--in.identd--5*[in.identd]
|-inetd
|-kflushd
|-klogd
|-kpiod
|-kswapd
|-kupdate
|-login--bash--pstree
|-5*[mingetty]
|-portmap
|-scanlogd
`-syslogd

```

## 10.2 Prozesse abbrechen

Um laufende Prozesse abzuberechnen gibt es zum einen die Möglichkeit, mit der Tastenkombination STRG-C einen Prozess abzuberechnen. Leider funktioniert dies nicht immer, besonders, wenn die



Prozesse abgestürzt sind oder als Zombies die Systemressourcen verbrauchen. Hier kommen die Kommandos kill und killall zu Hilfe.

**Tabelle:** Prozess-Signale

Signal	ID	Funktion
SIGHUP	1	Hangup, zwingt viele Programme, ihre Konfigurationsdateien neu einzulesen
SIGINT	2	Interrupt, STRG+C
SIGQUIT	3	Abbruch
SIGKILL	9	Zwingender Abbruch, durch das Programm nicht abfangbar
SIGUSR1	10	Benutzerdefiniertes Signal
SIGTERM	15	Prozess wird zum Beenden aufgefordert
SIGCONT	18	Ein gestoppter Prozess wird fortgesetzt
SIGSTOP	19	Stoppt den Prozess
SIGTSTP	20	Prozess-Stop vom Benutzer gesteuert

Mit kill können Signale an einen Prozess gesendet werden, nicht nur zum Abbrechen sondern z.B. auch um einen Prozess anzuhalten und nachher wieder fortzusetzen. Beim normalen Aufruf von kill über kill <PID> wird dem Prozess das Signal SIGTERM gesendet, welches den Prozess zum Beenden auffordert. Reagiert der Prozess auf dieses Signal nicht, kann die brachiale Methode verwendet werden, indem man über kill -9 <PID> oder kill -SIGKILL <PID> den zwingenden Abbruch des Prozesses fordert. Die Syntax des Befehls lautet:

kill -Signal <PID> oder kill -Signal-ID <PID>

Eine Auswahl der Signale ist in Tabelle wiedergegeben.

Beim Befehl killall muss nur als Parameter der Name des Kommandos oder Befehls, welcher abgebrochen werden soll, angegeben werden und alle (!) Prozesse mit diesem Kommando- bzw. Befehlsname werden abgebrochen.

### 10.3 Graphische Prozess-Tools

Natürlich gibt es unter Linux mehrere graphische Tools, um die Arbeit und die Übersicht zu erleichtern. Das Programm ktop ist ein graphisches Front-End zum Befehl top. Mit diesem Programm können die Prozesse ausgewählt werden, welche angezeigt werden sollen (alle, System, Nutzer, eigene), ob als Liste oder als Prozess-Baum und es können auch Prozesse abgebrochen werden. Zusätzlich kann man sich noch Informationen über die Systemauslastung anzeigen lassen (siehe Abbildung ).

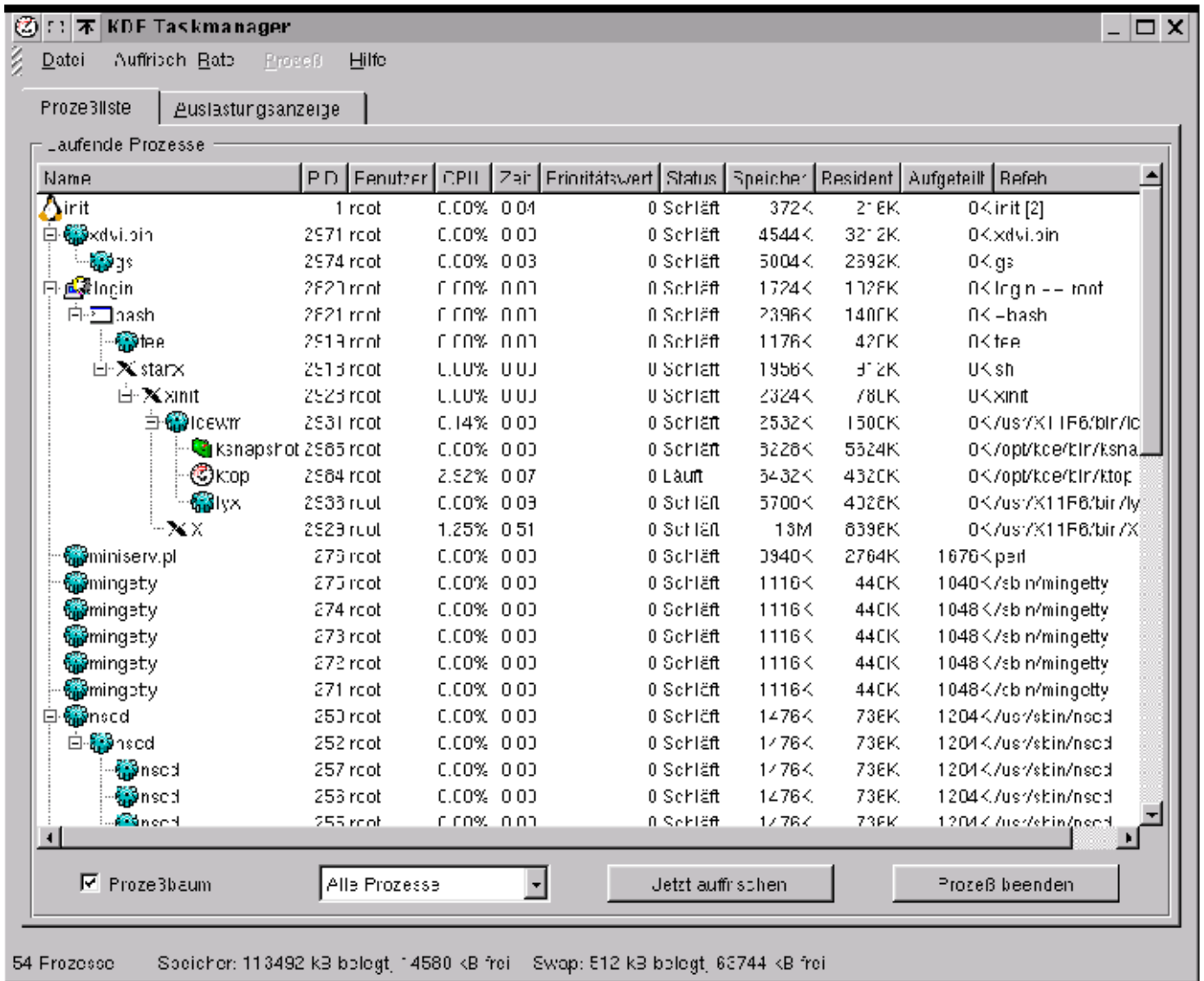


Abbildung: Das Programm ktop

Das Programm xosview zeigt zwar nicht einzelne Prozesse an, bietet aber durch die Anzeige der Systemauslastung und der Arbeitsspeicherbelegung eine gute Übersicht über das System (siehe Abbildung).

