Linux Offline HOWTO

Christoph Kania (kania@uni-duesseldorf.de) v1.0-3 24. Februar 1998

In diesem Dokument wird eine Lösung zur offline-Nutzung der Dienste Mail und News auf Standalone-Rechnern mit temporärem Internetanschluß über ein Modem dargestellt.

Inhaltsverzeichnis

1 Einführung

1.1 Vorwort

Auch wenn der Telekommunikationsmarkt nun privatisiert wurde, ist das Telefonieren - vor allem im Ortsbereich - nach wie vor nicht gerade günstig. Daher haben sich schon vor längerer Zeit Programme, vornehmlich in der Windows-Welt bewährt, die es ermöglichen, die Dienste News und Mail offline zu nutzen. Bis vor einiger Zeit fehlte diese Möglichkeit auf Unix-/Linux-Systemen ganz; mittlerweile gibt es aber, zumindest für den Dienst News, einige Programme zur Auswahl, die aber nicht immer ganz problemlos arbeiten und durch ihre »anwenderfreundliche Beschränkung« naturgegeben nicht sehr flexibel sind.

Deshalb habe ich mich entschieden, den klassischen Weg der Einrichtung von lokalem News- und Mailserver zu gehen. Dies entspricht zum einen der Anlage von Unixsystemen als Netzwerksysteme und bietet die Möglichkeit, z.B. sehr komfortable Datenbankfunktionen etc. einzubinden. Außerdem habe ich die Erfahrung gemacht, daß diese Server stabiler laufen als die - zugegeben vor längerer Zeit - getesteten »Einfachlösungen«. Die Entwickler der in diesem Bereich vertretenen Programme mögen meine Kritik verzeihen; ich gehe davon aus, daß, wie immer in der Linux-Welt, auch hier der Entwicklungsfortschritt sehr zügig vorangeht und in Zukunft leistungsfähige »Einfachlösungen« zur Verfügung stehen, durch welche die Einsteiger-Kompatibilität von Linux weiter gesteigert werden kann.

1.2 Neue Versionen dieser HOWTO

Neuere Versionen dieser Anleitung werden auf folgenden Servern bereitgestellt:

- `http://www-public.rz.uni-duesseldorf.de/~kania/mailnews.htm`
- `http://www.tu-harburg.de/dlhp/`

1.3 Feedback

Linux lebt von der Mitarbeit der Anwender. Deshalb bitte ich jeden, nicht zu zögern, mir seine Anmerkungen, Kritiken und Wünsche mitzuteilen:

`kania@uni-duesseldorf.de`

1.4 Copyright

1.5 Danksagungen

Eine ganze Reihe begeisterter Linuxer haben mir mit ihren Tips und Anmerkungen viel geholfen. Euch allen HERZLICHEN DANK!

1.6 Gundlegende und weiterführende Literatur

- HOWTOs des Deutschen Linux HOWTO Projektes (DLHP):

`http://www.tu-harburg.de/dlhp/`

- Internationale Linux-HOWTOs:

  `metalab.unc.edu:/pub/Linux/docs/HOWTO`

- Kania, Christoph: *Offline-Mail und Offline-News mit LINUX und Einwahl in die Heinrich-Heine-Universität Düsseldorf Version 1.5*

  `http://www-public.rz.uni-duesseldorf.de/~kania/mailnews.htm`

- Kania, Christoph: *Mails und News offline unter Debian GNU/Linux*

  `http://www-public.rz.uni-duesseldorf.de/~kania/mailnews.htm`

- Hailer, Bernhard: *Zugang zum Internet unter Linux über ISDN und das Leibniz-Rechenzentrum*, 19. Juli 1996

  `http://www.lrz-muenchen.de/~ui161ab/`

- `http://www.umr.edu/~mark/linux/ppp1.html`

- Dickebohm, Markus: *Konfiguration von Newsservern und Spoolprogrammen zum Offline-Lesen von Usenet-News*, Version 0.3, Köln 1996

  `http://www.uni-koeln.de/~acp66/`

- Hetze, Sebastian et al.: *LinuX Anwenderhandbuch und Leitfaden für die Systemverwaltung*, 5. erweiterte und aktualisierte Auflage, LunetIX Softfair 1995, ISBN 3-929764-0

  `http://www.lunetix.de`

- Bauer, Bodo et al.: *S.u.S.E. Linux 5.0 – Installation, Konfiguration und erste Schritte*, ISBN 3-930419-45-9

  `http://www.suse.de`

- Welsh, Matt und Kaufmann, Lar: *Linux - Wegweiser zur Installation & Konfiguration*, O'Reilly, ISBN 3-89721-133-5

  `http://www.oreilly.de/german/freebooks/linux_netz/inhalt.html`

- Kirsch, Olaf: *Linux Wegweiser für Netzwerker*, ISBN 3-930673-18-5

  `http://www.oreilly.de/german/freebooks/linux_install/inhalt.html`

- Costales, Bryan und Allman, Eric: *sendmail kurz & gut*, O'Reilly, ISBN 3-89721-202-1

  `http://www.oreilly.de`

1.7 Anmerkungen zur Textformatierung

Wo vom Anwender Angaben ergänzt werden müssen, wird dies durch die Verwendung eckiger Klammern dargestellt; diese sind nicht zu übernehmen. Z.B.:

```
domain [Domainname] # wenn eine bestimmte Domain verlangt
                    # wird
search [Domain des Providers]
nameserver [vom Provider erfragte IP]
nameserver [vom Provider erfragte IP]
```

ist zu ersetzen durch

```
domain uni-duesseldorf.de
search rz.uni-duesseldorf.de
nameserver 134.99.128.2
nameserver 134.99.128.5
```

2 Vorbereitungen

2.1 Programme und Versionen

Folgende Programme werden im folgenden benötigt. Da ich die Einrichtung auf einer Debian GNU/Linux 1.3.1 nachvollzogen habe, stehen in Klammern die Versionsnummern der Debian-Pakete (*.deb)

- Kernel: 2.0.30
- Netzwerk: netbase 2.16(-1)
- Verbindung: ppp 2.2.0f(-23)
- News
  - inn 1.5.1(-4)
  - suck 3.4.1(-2)
- Mail
  - sendmail 8.8.5(-1)
  - procmail 3.10.4(-2)
  - fetchmail 3.8(-0)

2.2 Grundvoraussetzungen

Folgende Grundvoraussetzungen müssen erfüllt sein:

- Der Kernel muß mit den Optionen `TCP/IP networking`, `PPP (point-to-point) support` und `Dummy net driver support` übersetzt sein.
- Man benötigt vom Provider folgende Angaben:
  - Telefonnummer
  - UserID
  - Paßwort
  - Wird Authentifizierung per *PAP* oder *CHAP* vorgenommen? Meistens wird *PAP* verwendet, so daß wir dieses in unseren Beispielen benutzen.
  - IP-Nummer des eigenen Rechners. Meistens wird die IP-Nummer dynamisch vergeben.
  - Wenn eine statische IP Verwendung findet, wird die Netmask benötigt.
  - IPs des/der Nameserver/s
  - Wird auf der Providerseite das PPP Protokoll automatisch gestartet? Dieses ist meistens der Fall.
  - Soll der eigene Rechner zu einer bestimmten Domain gehören?

2.3 Fundquellen der Programme

Bei allen hier verwendeten Programmen handelt es sich um frei verfügbare Dateien, die auf vielen FTP-Sites und auf den diversen Sunsite-Mirrors zu finden sind; sie sind aber schon in den meisten Distributionen enthalten.

2.4 Pfade

Da die verschiedenen Distributionen den FSSTND unterschiedlich bzw. ungenau umsetzen, können sich die hier angegebenen Pfade, die der Debian Distribution entnommen sind, von denen auf anderen Distributionen unterscheiden.

2.5 Programminstallation

Hierzu sind die READMEs unbedingt zu beachten. Da mittlerweile alle gängigen Distributionen über ein Paketmanagement verfügen und damit Binärpakete anbieten, sollte die Installation kein Problem mehr darstellen.

3 Allgemeine Netzwerkkonfiguration

3.1 /etc/hostname

Zunächst muß der zu verwendende Rechner einen Namen haben. Meiner heißt ganz einfach

```
kania
```

Dies ist der einzige Eintrag in der Datei `/etc/hostname`.

3.2 /etc/resolv.conf

Hier wird festgelegt, welcher Domain unser Rechner angehört und auf welche Nameserver zugegriffen werden soll.

```
domain [Domainname] # wenn eine bestimmte Domain verlangt
                    # wird
search [Domain des Providers]
nameserver [vom Provider erfragte IP]
nameserver [vom Provider erfragte IP]
```

3.3 /etc/host.conf

Um einen Namen aufzulösen (Netzwerkname, Netzwerkadresse, offizielle und inoffizielle Namen), soll, bevor einer der Nameserver befragt wird, in der Datei `/etc/hosts` nachgeschaut werden; außerdem dürfen die in `/etc/hosts` angegebenen Rechner mehrere IP-Adressen haben (`multi on`).

```
order hosts bind
multi on
```

3.4 /etc/networks

Hier werden den Netzwerkadressen (IPs) Netzwerknamen zugeordnet.

```
loopback 127.0.0.0
```

3.5 /etc/hosts

Mit dieser Datei werden den IP-Nummern einzelner Rechner Namen zugeordnet.

```
127.0.0.1     localhost
```

4 Die Verbindung herstellen

4.1 Was mache ich hier?

Wenn die hier beschriebene Konfiguration nachvollzogen wird, so wird folgendes eingerichtet:

- Anwahl des Providers; Befehl `pon`, `ppp-on`, `ppp-up` (abhängig von der verwendeten Distribution)
- Einloggen über PAP. Es wird hier angenommen, daß dieses vom Provider gefordert wird.
- Kappen der Verbindung mit `poff`, `ppp-off`, `ppp-down` (wiederum abhängig von der verwendeten Distribution).

4.2 Was geht ab?

Zunächst wählt `chat` den Provider an und stellt die Verbindung zwischen den beiden Modems her (Einigung über Protokolle etc.). Entsprechend PAP nimmt `chat` das Einloggen am Server vor und übergibt dann die Kontrolle an `pppd`, welcher das PPP-Protokoll weiter initiiert.

4.3 Konfiguration

Die für die Konfiguration wichtigen Dateien sind:

- `/etc/ppp/ppp.chatscript`
- `/etc/ppp/options`
- `/etc/ppp/pap-secrets`
- `/etc/ppp/ip-up`
- `ip-down`

4.3.1 /etc/ppp/ppp.chatscript

Folgendes `chat` Skript `ppp.chatscript` sollte im Verzeichnis `/etc/ppp` gespeichert werden:

```
TIMEOUT 60
ABORT "NO CARRIER"
ABORT BUSY
ABORT "NO DIALTONE"
ABORT ERROR
"" +++ATZ
OK ATDT[Telefonnummer des Providers]
CONNECT ""
```

4.3.2 /etc/ppp/options

Die Datei `options` nimmt einige Einstellung für den PPP-Daemon vor:

```
disconnect "chat -- \d+++\d\c OK ath0 OK"
asyncmap 0
crtscts
lock
115200  # maximale Geschwindigkeit des Modems
modem
[netmask 255.255.255.0]  # wenn nötig nach Providerangabe
noipdefault
debug
user [UserID]  # eigene UserID beim Provider
```

4.3.3 /etc/ppp/pap-secrets

Die Datei `pap-secrets` enthält das Paßwort, mit dem man sich beim Provider einloggt.

```
# Client  Server  Paßwort  IP-Adressen
UserID      *     passwd  # UserID und Password einsetzen
```

Da das eigene Paßwort auf keinen Fall in fremde Hände fallen sollte, da es ansonsten zum Mißbrauch des eigenen Accounts verwendet werden kann, sollten unbedingt die Rechte der Datei überprüft werden. Bedenken Sie bitte, daß nicht nur Sie über Ihre PPP-Verbindung auf das Internet zugreifen können, sondern auch Benutzer aus dem Internet auf Ihren Rechner.

Die Rechte der Datei kann man sich mit folgendem Befehl anschauen:

```
# ls -l /etc/ppp/pap-secrets
```

Als Ausgabe sollte der Befehl folgendes liefern:

```
-rw-------   1  root    root  1501  Feb  6 20:38 pap-secrets
```

Falls dieses nicht der Fall ist, sollte man die Rechte ändern:

```
chmod 600 /etc/ppp/pap-secrets
```

4.3.4 pon, ppp-on oder ppp-up

Dieses Skript baut eine PPP-Verbindung auf. Bitte beachten Sie, daß hier die Pfade zu `ppp.chatscript` und `ppp.options` korrekt angegeben sind.

Bei der Debian Distribution sieht die Datei so aus:

```
#!/bin/sh
if [ -r /etc/ppp/options -a -r /etc/ppp/ppp.chatscript ];
then
  /usr/sbin/pppd connect "/usr/sbin/chat -v \
     -f /etc/ppp/ppp.chatscript" `cat /etc/ppp/options`
else
  echo "You do not have permissions to access \
       /etc/ppp/ppp.chatscript or /etc/ppp/options"
fi
```

4.3.5 poff, ppp-off oder ppp-down

Dieses Skript beendet die PPP-Verbindung wieder. Bei der Debian Distribution sieht das Skript so aus:

```
#!/bin/sh

# Wieviele pppds laufen?
N=`ls /var/run/ppp* 2>/dev/null| wc -l`

# Wenn kein PPP Daemon läuft, dann mach poff nicht
# viel Sinn.
if [ $N = 0 ]; then
        echo "Es läuft kein pppd."
        exit 1
fi

# Wenn einer läuft, kann diese mit killall beendet werden.
if [ $N = 1 ]; then
        killall pppd
        exit 0
fi

# Es läuft mehr als ein Daemon. Es ist nicht klar, welcher
# beendet werden soll.
echo "Es läuft mehr als ein pppd. Keiner beendet."
exit 1
```

4.3.6 /etc/ppp/ip-up und /etc/ppp/ip-down

Wenn eine PPP-Verbindung hergestellt bzw. beendet wurde, wird das Skript `ip-up` bzw. `ip-down` ausgeführt. Dies kann man sich zunutze machen, um Aufgaben, die regelmäßig nach einem Verbindungsaufbau erledigt werden müssen, im Hintergrund abzuarbeiten. Ein gutes Beispiel hierfür ist z.B. das Holen oder Verschicken von Mails und News. Dazu aber später mehr. Vorerst lassen wir diese beiden Files unberührt.

4.3.7 Tuning der seriellen Schnittstelle

Aus historischen Gründen kann eine serielle Schnittstelle unter Linux nur auf Geschwindigkeiten bis maximal 38,4 kBit/s eingestellt werden, auch wenn wir es hier einige Male anders gemacht haben. Für neue Modems reicht dieses aber bereits ohne Komprimierung nicht mehr aus. Um die serielle Schnittstelle entsprechend zu tunen, wird in `/etc/rc.boot/0setserial`, das Skript kann je nach Distribution auch einen anderen Namen haben, folgender Eintrag aufgenommen:

```
${SETSERIAL} -b /dev/ttyS0 ${AUTO_IRQ} skip_test \
             autoconfig spd_vhi
```

`/dev/ttyS0` entspricht dem Anschluß COM1 unter DOS; entsprechend ist `/dev/ttyS1` COM2. Wenn jetzt ein Programm `dev/ttyS0` mit 38,4 kBit/s anspricht, wird die Hardware in Wirklichkeit mit 115,2 kBit/s angesprochen. Mit

```
setserial -b /dev/ttyS*
```

läßt sich die Einstellung überprüfen.

4.4 Ein erster Test

Wenn jetzt das Modem angeschlossen ist, zeigt uns die Eingabe von `pon`, `ppp-on` oder `ppp-up` als root, ob wir alles richtig gemacht haben.

Als Test machen wir hier ein `ping` auf eine vom Provider angegebene IP eines Nameservers:

```
# ping [IP]
```

Das Ergebnis sollte wie folgt aussehen, wobei die Ausgabe mit `<Strg>+<C>` abgebrochen werden kann:

```
PING 134.99.128.5 (134.99.128.5): 56 data bytes
64 bytes from 134.99.128.5: icmp_seq=0 ttl=253 time=189.4 ms
64 bytes from 134.99.128.5: icmp_seq=1 ttl=254 time=180.5 ms
64 bytes from 134.99.128.5: icmp_seq=2 ttl=254 time=240.1 ms
64 bytes from 134.99.128.5: icmp_seq=3 ttl=254 time=180.4 ms

--- 134.99.128.5 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 180.4/197.6/240.1 ms
```

Nachdem die eigentliche PPP-Verbindung nun, wie uns der Test bestätigt hat, einwandfrei funktioniert, können wir uns an die nächste Aufgabe begeben: die Einrichtung des Mailservers.

5 Mail

5.1 sendmail

5.1.1 /etc/sendmail.cf

Nach der Installation von `sendmail` müssen meist nur noch ein paar Änderungen in der Konfigurationsdatei von `sendmail` vorgenommen werden:

- Der Eintrag

  ```
  O DeliveryMode=background
  ```

  wird durch

  ```
  O DeliveryMode=deferred
  ```

  ersetzt. Hierdurch wird erreicht, daß `sendmail` alle Mails in einer Warteschlange zwischenspeichert und erst beim Aufruf mit der Option `-q` weiterleitet.

- Der Eintrag für den »Smart Host« muß von

  ```
  # "Smart" relay host (kann leer sein)
  DS
  ```

  in

  ```
  DS[Name des Mailservers]
  ```

  geändert werden. Die abzuschickenden Mails werden dann ohne weitere Fragen an `[Name des Mailservers]` übergeben.

- Außerdem muß dieser Eintrag von

  ```
  # who I masquerade as (null for no masquerading)
  # (see also $=M)
  DM
  ```

  in

  ```
  DM[Domain des Mailservers]
  ```

  geändert werden. Die Domain (steht hinter dem @) des Senders auf `localhost` wird abgeschnitten und durch `[Domain des Mailservers]` ersetzt; die UserID des Senders bleibt erhalten. Diese Option wird deshalb benötigt, damit Mails unter dem Domain des Providers verschickt werden, so daß das Antworten auf diese Mails möglich ist. Sollte der lokale User nicht die gleiche UserID haben wie auf dem Mailserver, muß dies über ein Mapping geregelt werden.

5.1.2 /etc/init.d/sendmail

Auch die Datei `/etc/init.d/sendmail` muß modifiziert werden. Aus

```
/usr/sbin/sendmail -qbd -om
```

(oder was dort auch immer stehen mag) wird

```
/usr/sbin/sendmail -bd -om
```

5.1.3 Wie werden die Mails abgeschickt?

Nun kann man, sobald eine Verbindung über PPP besteht, mit

```
# sendmail -q
```

alle Mails auf die Reise schicken. Um zu sehen, welche Mails sich noch in der Warteschlange befinden, kann folgendes Kommando benutzt werden:

```
# mailq
```

5.2 fetchmail

Nach der Installation von `fetchmail` muß die Datei `/root/.fetchmailrc` angelegt werden:

```
poll [Mailserver] protokoll POP3 user [UserID]
     password [passwd] is [User2]
```

`UserID` ist dabei die UserID auf dem Mailserver, `passwd` das dortige Paßwort und `User2` die lokale UserID, welche also nicht mit der auf dem Mailserver identisch sein muß. Die Post abholen kann man dann mit

```
fetchmail -a -L /var/log/fetchmail
```

Diese Optionen sagen `fetchmail`, daß es alle Mails holen, diese auf dem Mailserver löschen und, wenn nötig, ein Logfile `fetchmail` in `/var/log` anlegen soll.

6 News

Folgende Dateien sind wichtig für die Arbeit und Konfiguration von INN und `suck`:

- `/etc/hosts`
- `/etc/news/inn.conf`
- `/etc/news/hosts.nntp`
- `/etc/news/nnrp.access`
- `/etc/news/newsfeeds`
- `/etc/news/expire.ctl`
- `/etc/suck/get-news.conf`
- `/etc/suck/sucknewsrc`
- `/var/lib/news/active`

6.1 Den Benutzer »news« anlegen

Dies sollte schon bei der Installation geschehen sein; ansonsten geht es wie folgt:

- `# useradd -u 9 -g news -d /home/news -s /bin/bash -m news`
  Die numerische BenutzerID, hier wurde die Zahl 9 gewählt, darf auf dem System nur einmal vorkommen; deshalb sollte man zunächst in der Datei `/etc/passwd` nachschauen, welche Zahlen schon belegt sind, und eventuell eine andere auswählen, die aber kleiner als 99 sein sollte.
- In `/etc/aliases` sollte man folgende Einträge einfügen:

  ```
  bin: root
  news: root
  usenet: root
  newsmaster: root
  ```

  Damit werden Mails an `bin`, `news`, `usenet` und `newsmaster` an `root` weitergeleitet. Nun muß die Datenbank der Aliase noch neu erzeugt werden:

  ```
  # newaliases
  ```

6.2 Die Konfiguration von INN

6.2.1 /etc/news/inn.conf

```
organization:   [Firmenname o.ä.]
server:         localhost
```

6.2.2 /etc/news/hosts.nntp

Die Datei `/etc/news/hosts.nntp` enthält nur einen einzigen Eintrag:

```
localhost
```

6.2.3 /etc/news/nnrp.access

Die Datei `/etc/news/nnrp.access` sollte folgendermaßen aussehen:

```
*:: -no- : -no- :!
(none):ReadPost:::*
localhost:ReadPost:::*
```

6.2.4 /etc/news/expire.ctl

Die `/etc/news/expire.ctl` Datei bestimmt, nach welcher Zeitspanne alte Artikel gelöscht werden. Sie sollte ungefähr so aussehen:

```
/remember/:14
*:A:4:10:10
junk:A:1:1:2
control:A:1:1:2
*.test:A:1:1:1
```

6.2.5 /etc/news/newsfeeds

Die Datei `/etc/news/newsfeeds` muß so aussehen:

```
ME:*::
OVERVIEW!:*:Tc,WO:/usr/lib/news/bin/overchan
newsserv/[Name des Newsservers]:*:Ap,Tf,Wnm:newsserv
```

6.2.6 /var/lib/news/active

- Da die Datei `/var/lib/news/active` sehr lang ist, hier nur ein kurzer Ausschnitt:

  ```
  control 0000000000 0000000001 y
  junk 0000000000 0000000001 y
  local.general 0000000000 0000000001 y
  local.test 0000000001 0000000002 y
  to 0000000000 0000000001 y
  de.comp.os.linux.x 0000002735 0000001784 y
  de.comp.os.linux.misc 0000014074 0000007994 y
  de.comp.os.linux.networking 0000004226 0000003514 y
  hhu.forum 0000000320 0000000265 y
  hhu.linux 0000000016 0000000016 y
  hhu.modem 0000000087 0000000083 y
  hhu.bibliothek 0000000003 0000000004 y
  de.comp.text.tex 0000002821 0000002428 y
  de.comp.lang.perl 0000001178 0000000947 y
  hhu.wohnheime 0000000135 0000000122 y
  hhu.test 0000000033 0000000034 y
  de.alt.comp.kde 0000000581 0000000211 y
  ```

  In `active` ist der aktuelle Downloadstand der Newsgroups festgehalten. Es sollten keine unvorsichtigen Änderungen vorgenommen werden, da die Syntax sehr empfindlich ist.

- Zunächst einmal müssen wir uns die Datei bei unserem Newsserver besorgen. Dazu dient das Programm `/usr/lib/news/bin/getlist`. So muß man vorgehen:

```
# pon
# /usr/lib/news/bin/getlist -h newsserv active \
    > /var/lib/news/active.raw
```

- Jetzt setzen wir die Artikelnummern auf den Wert Null:

```
# cd /var/lib/news
# sed 's/ [0-9* [0-9] / 0000000000 0000000001 /' \
    active.raw > active
```

- Nun schauen wir noch nach, ob die folgenden drei Einträge vorhanden sind:

```
to 0000000000 0000000001 y
junk 0000000000 0000000001 y
control 0000000000 0000000001 y
```

- Als letztes muß noch der Besitzer und die Rechte gesetzt werden:

```
# chown news.news active
# chmod 644 active
```

6.2.7 Der letzte Schliff

- Da der `innd` Zugang zum Verzeichnis `/var/run` benötigt, setzen wir die Rechte folgermaßen:

```
# chown root.root /var/run
```

- Nun muß man noch unter `/var/spool/news` nach den folgenden Verzeichnissen suchen: `in.coming`, `news.archive`, `out.going` und `over.view`. Sollten sie nicht vorhanden sein, so muß man sie erzeugen:

```
# su news -c "mkdir ..."
```

- Jetzt müssen noch die Besitzer einiger Dateien geändert werden:

```
# chown -R news.news /usr/lib/news*
# chown -R news.news /var/lib/news*
# chown -R news.news /var/spool/news*
```

- Und schließlich müssen folgende Programme aufgerufen werden:

```
# su news -c "/usr/lib/news/bin/makehistory -o"
# su news -c "/usr/lib/news/bin/news.daily"
```

- Und ganz am Ende muß noch folgende Zeile in `/etc/inetd.conf` mit einem # auskommentiert werden:

```
nntp strem tcp nowait root /usr/sbin/tcpd in.nntpd
```

6.2.8 Der Moment der Wahrheit

Spannung. Ob es wohl funktioniert? Ein erster Test kann lokal auf dem eigenen Rechner durchgeführt werden:

```
# /usr/lob/news/bin/rc.news
```

Es sollte folgende Meldung erscheinen:

```
Starting innd
```

Nun kann man sich mit Telnet auf den eigenen Newsserver einloggen:

```
# telnet localhost 119
Trying 127.0.0.1....
Connected to localhost
Escape charakter is ']'
200 Rechner InterNetNews server INN 1.5.1 17-Dec-1996 ready
```

Mit `quit` geht's wieder zum gewohnten Prompt!

Wenn das geklappt hat: Herzlichen Glückwunsch. Die Wahrscheinlichkeit, daß der `innd` auch sonst richtig arbeitet, ist sehr hoch.

Sollte es nicht funktioniert haben: Herzliches Beileid, denn jetzt heißt es, den Fehler zu suchen: Schauen Sie nach, ob `root` eine Mail bekommen hat, überprüfen Sie die Konfiguration (vor allem Besitzer und Rechte kontrollieren); sehr hilfreich kann auch ein Aufruf von `/usr/lib/news/bin/inncheck` sein.

6.2.9 INN dauerhaft einrichten

Damit INN bzw. sein Daemon `innd` beim Hochfahren des Rechners automatisch gestartet wird, muß bei den meisten Distributionen nichts mehr eingerichtet werden. Anderenfalls kann man, je nach Distribution, entweder in `/etc/rc.d/rc.local` folgenden Abschnitt einfügen

```
# INN starten
if [ -x /usr/lib/news/bin/rc.news ] then
   /bin/sh /usr/lib/news/bin/rc.news
fi
```

oder man erzeugt in `/etc/rc.boot` ein File mit Namen `inn`, welches den oben gezeigten Inhalt hat. Ein korrekterer Weg wäre jedoch, in `/etc/rc2.d` einen Link names `S50inn` auf die Datei `/etc/init.d/inn` zu setzen, die folgenden Inhalt hat:

```
#!/bin/sh
# Startet/beendet den News Server

test -f /usr/sbin/innd || exit 0

case "$1" in
start)
        echo "Starte News Server: innd"
        start-stop-daemon --quiet --start --user root \
           --pidfile /var/run/innd/innd.pid \
           --startas /etc/news/boot | \
           sed -e '/Starting innd./d' \
               -e 's/^/innd: boot: /' &
        sleep 2
        ;;
stop)
        echo "Stoppe News Server: innd"
        if [ -f /var/run/innd/innwatch.pid ]
        then
                start-stop-daemon --quiet --stop \
                   --pidfile /var/run/innd/innwatch.pid
        fi
```

```
        start-stop-daemon --quiet --stop \
                --pidfile /var/run/innd/innd.pid \
                --exec /usr/sbin/innd
        ;;
*)      echo "Syntax: /etc/init.d/inn start|stop";
        exit 1 ;;
esac

exit 0
```

`rc2.d` gilt für Runlevel 2; wenn das System auch in anderen Runleveln läuft, z.B. Runlevel 3 beim Einsatz von `xdm`, so muß auch im entsprechenden `rc[n].d` Verzeichnis ein Link `S20inn` vorhanden sein. Sicherheitshalber sollte man dieses für alle Runlevel machen. Eine Ausnahme stellen jedoch die Runlevel 0 (halt), 1 (Single-User) und 6 (reboot) dar; hier lautet der Link `K50inn`, hat aber das gleiche Ziel. Zur Behandlung der Runlevel sollte unbedingt die Dokumentation der jeweiligen Distribution beachtet werden.

6.2.10 Die regelmäßigen Arbeiten

Damit die News, welche auf den Rechner geschaufelt und hier geschrieben werden, einer regelmäßigen Wartung unterzogen werden, engagieren wir einen kleinen Helfer, der diese Arbeiten erledigt. Wann er was machen soll, wird in `/etc/crontab` festgelegt.

- Hier als Beispiel meine `crontab`:

```
# /etc/crontab

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:\
/usr/sbin:/usr/bin

# m h dom mon dow user  command
42 12   * * *   root    run-parts /etc/cron.daily
47 12   * * 7   root    run-parts /etc/cron.weekly
52 12   1 * *   root    run-parts /etc/cron.monthly
```

- In `/etc/cron.daily` findet man die Skripte, die täglich ausgeführt werden sollen. Für INN steht dann in `/etc/cron.daily/inn`:

```
#!/bin/sh

su news -c /usr/lib/news/bin/news.daily
```

6.3 suck

6.3.1 Wichtige Dateien

Zu beachten sind folgende Datein:

- `/etc/suck/get-news.conf`
- `/etc/suck/sucknewsrc`

6.3.2 /etc/suck/get-news.conf

Die Datei /etc/suck/get-news.conf sollte so aussehen:

```
server: localhost
remoteserver: [Newsserver]
outgoingfile: newsserv
#userid:
#password:
sedcmd: /^NNTP-Posting-Host:\|^Xref:/d
```

6.3.3 /etc/suck/sucknewsrc

Mit der Datei /etc/suck/sucknewsrc wird suck mitgeteilt, welche Gruppen zu holen sind. Die Datei könnte so aussehen:

```
#comp.os.linux.announce -1
#comp.security.announce -1
#gnu.announce -1
#news.announce.newusers -1
#news.newusers.questions -1
de.comp.os.linux.x 16314
de.comp.os.linux.misc 67222
de.comp.os.linux.networking 25000
hhu.forum 4588
hhu.linux 225
hhu.modem 2160
hhu.bibliothek 130
#hhu.test 1383
de.comp.text.tex 13466
de.comp.lang.perl 6338
hhu.wohnheime 278
de.alt.comp.kde 653
```

Bei der ersten Einrichtung einer Gruppe gibt man als Zahl z.B. -20 an. Das bedeutet, suck soll die von diesem Zeitpunkt an 20 neuesten Nachrichten der Gruppe holen. Der Wert wird dann bei jedem Polling aktualisiert.

Der entsprechende Befehl lautet

```
# get-news
```

Dies wiederum ist ein Link auf /usr/sbin/get-news.inn. suck übernimmt dann sowohl das Saugen neuer News als auch das Posten der Artikel, die lokal verfasst wurden. Dabei werden aber nur Artikel an den Newsserver weitergeleitet, die nicht von ihm stammen, so daß keine Dupes erzeugt werden.

7 Nach der Arbeit sollst Du ruhen

Nun kann man ein paar Mails, z.B. mit pine, schreiben oder die aktuellen News, z.B. mit tin, lesen.

Damit, nachdem die anstrengende Arbeit vollendet ist, in Zukunft möglichst selten Hand angelegt werden muß, kann der Vorgang des Holens und Verschicken von Mails und News automatisch durch das Skript in /etc/ppp/ip-up erledigt werden.

7.1 /etc/ppp/ip-up

Nochmals zur Erinnerung, das Skript `/etc/ppp/ip-up` wird ausgeführt, wenn die PPP-Verbindung hergestellt ist. Die Datei `ip-up` könnte z.B. so aussehen:

```
PATH=/usr/local/sbin:/usr/sbin:/sbin:/usr/local/bin:\
/usr/bin:/bin
export PATH

cat /usr/lib/sounds/doorbell.au > /dev/audio
echo > /dev/xconsole
echo > /dev/xconsole
echo "Verbindung hergestellt" > /dev/xconsole
echo "lokale IP: $4      entfernte IP: $5" > /dev/xconsole
#echo "Geschwindigkeit $3" > /dev/xconsole
echo > /dev/xconsole
echo "Post holen ..." > /dev/xconsole
fetchmail -a -L /var/log/fetchmail
echo > /dev/xconsole
echo "Post schicken ..." > /dev/xconsole
sendmail -q
echo > /dev/xconsole
echo "... news holen ..."  > /dev/xconsole
get-news
echo > /dev/xconsole
echo "... fertig"  > /dev/xconsole
echo > /dev/xconsole
```

Die `xconsole` starte ich mit

```
xconsole -geometry 480x130-0-0 -daemon -notify -verbose \
         -fn fixed -exitOnFail -file /dev/xconsole
```