# Linux ISP-Verbindung HOWTO

Egil Kvaleberg (egil@kvaleberg.no) und Christoph Seibert (seibert@cs.uni-bonn.de) v1.24-3, 3. April 1998

Dieses Dokument beschreibt, wie man unter Linux über ein Modem Verbindung mit einem Internet Service Provider aufnimmt. Außer den Grundlagen des Einwahlverfahrens werden auch die Themen EMail und News behandelt.

## Inhaltsverzeichnis



# 1 Einleitung

Dies ist die Übersetzung des englischen *ISP-Hookup HOWTO* von Egil Kvaleberg. Die Versionsnummer entspricht der des Originals. Die Bezeichnung ISP wurde beibehalten, weil sie eine praktische Abkürzung ist und man die Zugehörigkeit zum Original erkennt. Außerdem hat sich auch im deutschen Sprachraum der Begriff Provider eingebürgert. Der Übersetzer nimmt aber gerne andere Vorschläge entgegen.

Diese Beschreibung wurde erstellt, um Fragen zu beantworten, wie man als Kunde eines ISP's (Internet Service Providers) Linux konfigurieren und benutzen kann.

Um denen zu helfen, die mit ihrem Linux-System zum ersten Mal eine Verbindung zu einem ISP herstellen wollen, wird hier der Versuch unternommen, alle relevanten Themen zu behandeln. Es wird daher unausweichlich Überschneidungen mit anderen Linux HOWTOs und Büchern des LDP (Linux Documentation Project) geben. Zum besseren Verständnis und für Detailfragen sollte man diese Dokumente zu Rate ziehen.

Ein großer Teil der vorhandenen Dokumentation ist auf Benutzer mit einer gewissen Erfahrung zugeschnitten. Unerfahrene Benutzer haben oft Schwierigkeiten, die für sie wichtigen Informationen herauszusuchen.

Der Einfachheit halber gehen die Beispiele von folgenden Annahmen aus:

- Benutzername: `dirk`
- Paßwort: `UnheimlichGeheim`
- Internet Service Provider: `acme.net`
- E-Mail-Server: `mail.acme.net`
- News-Server: `news.acme.net`
- Nameserver: `193.212.1.0`
- Telefonnummer: `12345678`

Der Benutzer `dirk` nennt seinen Rechner `roderick`.

Alle Einträge in der obigen Tabelle sollten natürlich abhängig vom benutzten ISP ersetzt werden. Oft reichen sonst schon minimale Änderungen für Kunden anderer ISPs. Der Autor würde gerne über eventuelle Probleme in dieser Hinsicht informiert werden.

## 1.1 Neue Versionen dieses Dokuments

Neue Versionen des englischen Originals *ISP-Hookup HOWTO* werden regelmäßig in `comp.os.linux.answers` gepostet. Sie werden auch den verschiedenen FTP-Servern hinzugefügt, die solche Informationen archivieren, unter anderem:

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

Zusätzlich sollte man das Original auf der Homepage des *Linux Documentation Project* finden können:

```
http://metalab.unc.edu/LDP/
```

Die jeweils neueste deutsche Version dieses HOWTO ist auf dem WWW Server des *Deutschen Linux HOWTO Projekts* unter folgender Adresse zu finden:

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

Hier kann man die jeweils aktuelle Version auch gleich online lesen.

### 1.2 Feedback

Alle Kommentare, Berichte über Fehler, zusätzliche Informationen und Kritiken aller Art sollten an die Adresse des Übersetzers gehen:

`seibert@cs.uni-bonn.de`

Fragen, die nicht die Übersetzung betreffen, werden dann an den Autor des Originals weitergeleitet:

- `egil@kvaleberg.no`
- `http://home.sn.no/home/egilk/`

### 1.3 Disclaimer

Für den Inhalt dieses Dokuments kann keine Haftung übernommen werden. Die Ausführungen, Beispiele und der restliche Inhalt werden auf eigene Gefahr benutzt. Außerdem handelt es sich hier um eine frühe Version, die möglicherweise viele Ungenauigkeiten und Fehler enthält.

Beschrieben wird eine von vielen denkbaren Methoden. In der Welt von Linux gibt es normalerweise mehrere Wege, auf denen man zum Ziel kommt. Es werden einige Hinweise auf Alternativen gegeben. Außerdem ändern sich FTP-Angaben oft ein wenig, wenn neue Programmversionen erscheinen.

So weit ich weiß, werden nur Programme beschrieben, die unter bestimmten Bedingungen im privaten Bereich benutzt oder ausprobiert werden können. Die meisten Programme sind unter GNU-ähnlichen Bedingungen mit Quellcode erhältlich.

### 1.4 Copyright

Dieses Dokument ist urheberrechtlich geschützt. Das Copyright für die englische *ISP-Hookup HOWTO*, auf der dieses Dokument basiert, liegt bei Egil Kvaleberg. Das Copyright für die deutsche Version liegt bei Christoph Seibert.

Das Dokument darf gemäß der GNU *General Public License* verbreitet werden. Insbesondere bedeutet dieses, daß der Text sowohl über elektronische wie auch physikalische Medien ohne die Zahlung von Lizenzgebühren verbreitet werden darf, solange dieser Copyright-Hinweis nicht entfernt wird. Eine kommerzielle Verbreitung ist erlaubt und ausdrücklich erwünscht. Bei einer Publikation in Papierform ist das Deutsche Linux HOWTO Projekt hierüber zu informieren.

## 2 Wie verbinde ich mich mit dem Rest der Welt?

Es wird davon ausgegangen, daß die wichtigsten Teile der Netzwerk-Software Ihrer Distribution bereits installiert sind. Auch sollte `/dev/modem` auf die zu verwendenen serielle Schnittstelle zeigen.

Die Standard-Konfiguration erlaubt normalerweise nur dem Benutzer `root` den direkten Zugriff auf `/dev/modem`.

Um sich direkt mit dem Benutzeraccount des ISP zu verbinden und mit den Befehlen zum Aufbau der Verbindung zu experimentieren, kann man das Programm `minicom` benutzen. Die Bedienung ist sehr leicht erlernbar.

### 2.1 Die Grundkonfiguration

Die Konfiguration des Rechners zum Anschluß an das Netz sollte man als Benutzer `root` durchführen. Bevor es weitergeht, versichert man sich, daß die Datei `/etc/hosts.deny` folgende Zeile enthält:

```
ALL: ALL
```

Normalerweise erlaubt man sich selbst den Zugang, also fügt man folgende Zeile der Datei `/etc/hosts.allow` hinzu:

```
ALL: LOCAL
```

Oder wenn man darauf besteht:

```
ALL: 127.0.0.1
```

Es ist zu beachten, daß es im folgenden um diejenigen geht, die eine Verbindungsmöglichkeit über PPP und mit einer dynamischen IP-Adresse haben. Hat man das Glück, eine feste Verbindung zu besitzen, ergeben sich einige Unterschiede.

Es ist schön, wenn der Rechner einen Namen hat; einen Namen, den der Benutzer einer dynamischen IP-Adresse völlig frei wählen kann. Diesen Namen trägt man in `/etc/HOSTNAME` ein:

```
roderick
```

Als nächstes trägt man den Nameserver in `/etc/resolv.conf` ein:

```
search .
nameserver 193.212.1.0
```

Der Nameserver muß als numerische IP-Adresse eingetragen sein und unterscheidet sich von ISP zu ISP. Falls man es benötigt, kann man bis zu drei verschiedene Server angeben. Diese werden dann in der Reihenfolge abgefragt, in der sie aufgeführt sind.

Will man Namen wie `einrechner` als Abkürzung für `einrechner.acme.net` verwenden können, muß die erste Zeile folgendermaßen ersetzt werden:

```
search acme.net
```

Ein gewisses Minimum an Konfiguration findet auch in `/etc/hosts` statt. Für die meisten Benutzer wird folgendes ausreichen:

```
127.0.0.1 localhost
0.0.0.0   roderick
```

Wer eine feste IP-Adresse hat, trägt sie natürlich anstelle von 0.0.0.0 ein.

Ähnlich sieht eine minimale `/etc/networks` aus:

```
loopback  127.0.0.0
localnet  0.0.0.0
```

Außerdem sollte man seine externe Maildomain in `/etc/mailname` setzen:

```
acme.net
```

Der Benutzername und das Paßwort beim ISP muß in `/etc/ppp/pap-secrets` angegeben sein:

```
dirk * UnheimlichGeheim
```

Benutzt der ISP CHAP statt PAP, heißt die Datei `/etc/ppp/chap-secrets`.

Schließlich müssen noch die Befehle zur Verbindungsaufnahme angegeben werden, bevor PPP gestartet werden kann. Dies geschieht in `/etc/ppp/chatscript`:

```
TIMEOUT 5
"" ATZ
OK ATDT12345678
ABORT "NO CARRIER"
ABORT BUSY
ABORT "NO DIALTONE"
ABORT WAITING
TIMEOUT 45
CONNECT ""
TIMEOUT 5
"name:" ppp
```

Ein paar Einzelheiten müssen wahrscheinlich noch geändert werden. Die Telefonnummer in der dritten Zeile wird natürlich wie benötigt angepaßt. Manche Benutzer müssen vielleicht die Initalisierung des Modems von `ATZ` auf einen auf ihr Modem abgestimmten Befehl ändern. Die letzte Zeile gibt an, daß die Eingabeaufforderung des Providers `name:` lautet und daß darauf mit `ppp` geantwortet wird. Andere Systeme benutzen vielleicht andere Login-Prozeduren.

Um schließlich einen Anruf zu tätigen, kann PPP durch folgenden Befehl gestartet werden:

```
exec pppd connect \
     'chat -v -f /etc/ppp/chatscript' \
      -detach crtscts modem defaultroute \
      user dirk \
      /dev/modem 38400
```

Jetzt sollte man verbunden sein, und zwar so lange, bis das Programm durch Strg-C abgebrochen wird. Eventuelle Nachrichten betreffs der Verbindung werden der Logdatei des Systems hinzugefügt. So kann man sie lesen:

```
tail /var/adm/messages
```

Solange die PPP-Verbindung besteht, hat man direkten Zugang zum Internet und kann Programme wie `ftp`, `ncftp`, `rlogin`, `telnet`, `finger` usw. benutzen. Diese Programme sollten alle Teil des Netzwerkpakets sein.

Weitere Informationen über PPP sind hier zu finden:

- `/usr/lib/ppp/README.linux`
- `/usr/lib/ppp/README.linux-chat`

Zuletzt noch ein paar Worte zur Sicherheit. Die Datei `/etc/inetd.conf` führt alle Dienste auf, die der Rechner der Außenwelt anbietet. Durch die oben erstellte `/etc/hosts.deny` wird kein Zugriff von außen zugelassen. Wird er benötigt, muß der Zugriff ausdrücklich in `/etc/hosts.allow` erlaubt werden. Lokale Zugriffe werden hierdurch zugelassen:

```
ALL: LOCAL
```

Siehe auch `man 5 hosts_access`.

Noch eine letzte Kleinigkeit: Über die Namen der POP-Protokolle herrscht eine gewisse Verwirrung. Folgende Definition in `/etc/services` sollte fast immer funktionieren:

```
pop2       109/tcp   pop-2       # PostOffice V.2
pop3       110/tcp   pop-3 pop   # PostOffice V.3
```

Statt `chatscript` kann man auch das wesentlich flexiblere `dip` benutzen, jedoch nicht in Verbindung mit `diald`.

Wer glücklicher Besitzer einer ständigen TCP/IP-Verbindung z.B. über Ethernet ist, kann getrost alles vergessen, was PPP betrifft, und sich damit beschäftigen, seine Netzwerkkarte zu konfigurieren.

Andere wiederum haben vielleicht nicht die Möglichkeit, PPP zu benutzen, können aber statt dessen SLIP benutzen, das ähnlich wie PPP unterstützt wird. Eine weitere Möglichkeit ist UUCP. Wieder andere müssen sich eventuell auf den Austausch von News und E-Mail über SOUP verlassen. Für diesen Fall befindet sich eine Beschreibung in:

```
ftp.sn.no:/user/bjorn/Linux-offline.tgz
```

Eine zusätzliche Möglichkeit ist TERM. Darüber kann man sich im *Term HOWTO* informieren.

# 3 Wie surfe ich?

Sollte einem der Text am wichtigsten sein, kann man den WWW-Browser Lynx benutzen. Er ist hier erhältlich:

```
metalab.unc.edu:/pub/Linux/system/Network/info-systems/
```

Ist X11 installiert, kann man einen der vielen graphischen Browser benutzen. Chimera befindet sich im gleich Verzeichnis wie Lynx. Der veraltete Browser Mosaic ist hier zu bekommen:

```
ftp.NCSA.uiuc.edu:/Web/Mosaic/Unix/binaries/2.6
```

Am weitesten ist Netscape verbreitet. Dieser Browser ist auf folgendem Server zu bekommen:

```
ftp.cs.uit.no:/pub/www/netscape
```

Diese Browser erscheinen ständig in neuen und in mancher Hinsicht oft »aufregenden« Versionen.

Diese Programme können unter bestimmten Bedingungen benutzt und ausprobiert werden. Dieses sollte man beachten.

# 4 Wie verschicke und empfange ich E-Mail?

Zunächst versichert man sich, daß `sendmail` installiert ist. Sendmail sortiert interne und ausgehende Post und sammelt ausgehende Post, bis es möglich ist, sie abzusenden.

Sendmail wird in `/etc/sendmail.cf` konfiguriert. Ein Beispiel, das Kunden eines ISP benutzen können, befindet sich in:

```
ftp.sn.no:/user/egilk/sendmail.cf
```

Dies stützt sich auf `procmail` als das Programm, das E-Mail versendet. Es kann aber leicht so geändert werden, daß `deliver` benutzt wird.

Man braucht natürlich eine *offizielle* Adresse für ausgehende Post, die in `/etc/sendmail.cf` angegeben wird.

```
# für wen ich mich ausgebe (leer für kein Masquerading)
DMacme.net
```

Hier wird angenommen, daß der lokale Benutzername der gleiche ist, den man auch beim ISP hat. Jetzt ist `sendmail` so eingerichtet, daß es Post *direkt* an den Empfänger schickt. Um lange und wiederholte Verbindungen in dem Fall zu vermeiden, daß die Verbindung zum Empfänger langsam und unregelmäßig ist, ist es gut, wenn man seinen ISP als Puffer benutzt. Das kann in der DS-Angabe eingetragen werden:

```
# "Smart" relay host (kann leer sein)
DSmail.acme.net
```

Zu beachten ist, daß `sendmail` ein wenig empfindlich ist, was die Behandlung von Tabulatorzeichen in `sendmail.cf` angeht. Man sollte vielleicht den Editor `vi` benutzen, um sicherzugehen, daß diese Tabulatorzeichen unverändert bleiben.

Der Empfang von E-Mail kann oft über das POP3-Protokoll geschehen, das bei jeder Verbindungsaufnahme aufgerufen werden kann. Das kann man über folgendes Skript ausprobieren:

```
sendmail -q
popclient -3 -v mail.acme.net -u dirk \
          -p "UnheimlichGeheim" \
          -k -o /usr/spool/mail/dirk
```

Dieses Skript kann man aufrufen, nachdem die PPP-Verbindung hergestellt worden ist. Man sollte darauf achten, daß dieses Skript nur Testzwecken dient; darum sollte man sich versichern, daß der eigene Briefkasten unverändert bleibt, während es läuft. Die Option `-k` bedeutet, daß die Post im Briefkasten beim ISP *bleibt* und daß man nur eine Kopie erhält. Diese Option sollte man natürlich entfernen, wenn man sicher ist, daß alles funktioniert.

Man beachte, daß das Paßwort in der Kommandozeile steht. Dem sollte wirklich so bald wie möglich abgeholfen werden.

Eine sichere und bessere Version dieses Skripts befindet sich hier:

```
ftp.sn.no:/user/egilk/pop-script.tar.gz
```

Diese Version des Skripts benötigt ein installiertes `procmail`, aber das wird man nie bereuen:

```
metalab.unc.edu:/pub/Linux/system/Mail/mailhandlers/
```

Bei `procmail` handelt es sich um ein einfaches und leistungsfähiges Werkzeug, daß eingehende E-Mail aufgrund vieler verschiedener Kriterien sortieren kann. Außerdem kann es sich um automatische Aufgaben wie eine Ferienvertretung kümmern.

Wenn man `procmail` wie in diesem Fall direkt benutzt, liegt der Fall etwas anders, als er in der Dokumentation zu `procmail` beschrieben ist. Man braucht *keine* `.forward`-Datei und auch keine `.procmailrc`. Letztere braucht man nur, wenn die Post sortiert werden soll.

Eine Benutzeroberfläche, mit der man Post lesen und versenden kann, findet man in Programmen wie Pine oder Elm.

In letzter Zeit hat sich `fetchmail` als sehr interessante Alternative zu `popclient` erwiesen. Die neueste Version gibt es auf:

```
ftp.ccil.org:/pub/esr/fetchmail/
```

Als normaler Kunde eines ISP benötigt man eigentlich keinen aktivierten `sendmail`-Daemon. Um Ressourcen zu sparen, kann man deswegen den Aufruf von `sendmail` unterdrücken, wie er sich normalerweise in `/etc/rc.d/rc.M` befindet.

Anstelle von `sendmail` kann man auch das einfachere `smail` benutzen. Eine gute Beschreibung der Programme und auch vieles andere, was hier erwähnt wird, findet man im *Linux Network Administrator's Guide*.

Es gibt auch ein m4 Makropaket, um eine neue `sendmail.cf` zu erzeugen. Für eine einfache Installation kann man genausogut eine bestehende Konfiguration verändern.

Es gibt auch noch einfachere, aber weniger leistungsfähige Alternativen. Pine zum Beispiel kann auch alleine laufen, wenn es richtig konfiguriert ist. Eventuell kann man auch neuere Versionen einiger WWW-Browser verwenden.

Viele sind vom Emacs-Modul Gnus zur Behandlung von E-Mail und News begeistert. Weitere Informationen hierzu befinden sich in:

```
http://www.ifi.uio.no/~larsi/
```

Eine Alternative zu `popclient` ist `pop-perl5`. Es ist hier erhältlich:

```
metalab.unc.edu:/pub/Linux/System/Mail/
```

# 5 News

## 5.1 Wie lese ich News online?

Solange eine PPP-Verbindung besteht, ist es möglich, News *online* zu lesen. Dazu sind viele Programme erhältlich. Zwei einfache Alternativen sind `rtin` und `trn`.

Das einzige, was man in den meisten Fällen konfigurieren muß, um News lesen zu können, ist die folgende Umgebungsvariable, die man normalerweise in der Datei `.profile` setzt:

```
export NNTPSERVER=news.acme.net
```

Manche Programme benötigen vielleicht folgendes, um die `From`-Adresse in eigenen Beiträgen richtig einzutragen:

```
export NNTP_INEWS_DOMAIN=acme.net
```

## 5.2 Wie lese ich News offline?

Man muß auf irgendeine Weise einen lokalen News-Spool einrichten, um News lesen zu können, während man offline ist, womit man Telefonkosten spart und flexibler ist. Dafür benötigt man einiges an Konfiguration und auch ein gewisses Maß an Platz auf der Festplatte. Hat man das einmal eingerichtet, sollte alles mehr oder weniger von selbst laufen, und man muß sich nur ab und zu etwas darum kümmern.

Hier werden zwei verschiedene Lösungen beschrieben.

## 5.3 Wie richte ich C News ein?

Die hier beschriebene Lösung legt den Newsserver *C News* und das NNTP-Protokoll zugrunde. C News war ursprünglich für einen anderen Verwendungszweck gedacht, ist aber leistungsfähig genug, um mit unserer Situation auch fertig zu werden. Man könnte auch den neueren Newsserver *INN* benutzen, der aber vielleicht mehr Ressourcen benötigt. Man sollte auf jeden Fall darauf achten, *nicht* beide zu installieren, da sie sich nicht gut vertragen.

Es ist sehr wichtig, daß man die Wartung des Newssystems als Benutzer `news` durchführt und daß sich alle Konfigurationsdateien in `/usr/lib/news` befinden. Ist man als Benutzer `root` eingeloggt, kann man das durch Eingabe von `su news; cd` bewerkstelligen.

Die wichtigsten Dateien zur Konfiguration sind:

- `active` ist eine Übersicht über vorhandene Newsgruppen. Sie wird - wenn nötig - über den Befehl `addgroup` aktualisiert, z.B.:

  ```
  addgroup comp.os.linux.networking y
  ```

- `organization` sollte nur das enthalten, was im Feld `Organization:` im Kopf der Nachricht stehen soll. In unserem Fall ist das folgendes:

  ```
  Dirk Gently's Holistische Detektei
  ```

- `mailname` enthält in unserem Fall `acme.net`.
- `whoami` wird für den Namen unseres Rechners in der `Path:`-Zeile verwendet. Wird wie in diesem Fall `NewsX` benutzt, verläßt dieser Name nie den eigenen Rechner. Man kann hier also eintragen, was man will, solange der Name einmalig ist. In unserem Beispiel verwenden wir `roderick`.
- `sys` kontrolliert den Erhalt und die weitere Verbreitung der News. In unserem Fall nehmen wir an, daß der ISP dem Pfad `acme.net` hinzufügt, und daß dies unsere einzige Quelle für News ist. Das folgende Beispiel besagt, daß wir alles annehmen, was ankommt, und daß wir an `acme.net` nur die News weitergeben, die dort noch nicht waren und die ursprünglich auf unserem Rechner abgesendet wurden. `/all` gibt die Verbreitung an und *muß* enthalten sein. Der Buchstabe `F` besagt, daß Referenzen auf ausgehende Artikel in einer Datei gesammelt werden.

  ```
  ME:all/all::
  acme/acme.net:all,!junk/all:FL:
  ```

- Es muß ein Unterverzeichnis für ausgehende News erstellt werden, in unserem Fall:

  ```
  mkdir /var/spool/news/out.going/acme
  ```

- `mailpaths` beschäftigt sich damit, wie Beiträge in moderierte Gruppen gesendet werden; aber diese Aufgabe sollte man normalerweise dem ISP überlassen.

C News benötigt ein gewisses Maß an täglicher Wartung, was man aber ein für alle Mal erledigen kann, wenn man den Befehl `crontab -e` als Benutzer `news` ausführt. Es folgt ein Vorschlag, der nach Wunsch geändert werden kann:

```
# ein- und ausgehende News behandeln
10,40 *  * * * /usr/lib/newsbin/input/newsrun

# täglich alte Nachrichten verfallen lassen
30 0   * * * /usr/lib/newsbin/expire/doexpire

# überwachen und wenn nötig Meldung erstatten
00 2   * * sat /usr/lib/newsbin/maint/addmissing
40 3   * * * /usr/lib/newsbin/maint/newswatch
50 3   * * * /usr/lib/newsbin/maint/newsdaily
```

`newsrun` holt und versendet zweimal pro Stunde Artikel, `doexpire` löscht um 0:30 Uhr Artikel, die veraltet sind, und die letzten drei Befehle erledigen verschiedene Aufgaben zur Überwachung und zur Behebung von Fehlern.

Man sollte sich auch versichern, daß bei einem Neustart des Rechners aufgeräumt wird. Dazu fügt man der Datei `/etc/rc.d/rc.local` als root folgende Zeile hinzu:

```
su news -c /usr/lib/newsbin/maint/newsboot
```

News kann man über das Programm *NewsX* beziehen, das sie von einem NNTP-Server holt. Das Programm befindet sich hier:

- `metalab.unc.edu:/pub/Linux/system/news/transport/`
- `ftp.sn.no:/user/egilk/`

Es ist ziemlich einfach, NewsX einzurichten. Die Installation ist ein klassischer Fall von:

```
make
su
make install
exit
```

Mit der hier beschriebenen Vorgehensweise muß man nur die Gruppen, die man haben möchte, mit dem Befehl `addgroup` erstellen.

Der Benutzer `news` führt folgende Befehle aus, um News zu holen, wobei natürlich eine PPP-Verbindung bestehen muß:

```
newsrun
newsx acme news.acme.net
newsrun
```

Über die Option `-d` erhält man eine fortlaufende Ausgabe auf dem Bildschirm. Die Dokumentation zu NewsX enthält weitere Informationen.

NewsX kümmert sich auch um das Versenden ausgehender Nachrichten.

Die Datei `explist` wird benötigt, um zu bestimmen, wie Artikel entfernt werden, wenn sie veraltet sind. Die Kommentare im Beispiel sollten erklären, was wir erreichen wollen:

```
# History-Einträge werden 14 Tage behalten, keiner erhält
# >120 Tage
/expired/           x    14   -
/bounds/            x    0-1-120 -

# behalte diese Gruppen 2 Monate lang
comp.sources,comp.os.linux.all     x    60   -

# Müll wird schnell weggeworfen
junk,control             x    2    -

# Normalfall: 14 Tage, keine Archivierung
all                 x    14   -
```

Bei einem kleinen News-Spool braucht man die Gruppe `control` oft nicht. Im Vergleich zum Nutzen enthält sie *sehr viele* Nachrichten. Der Zweck besteht darin, daß Artikel entfernt und Gruppen automatisch erstellt werden dürfen. Damit control-Nachrichten, die `newgroup` enthalten, nichts durcheinanderbringen, gibt es eine Datei `newgroupperm`, die angibt, was erlaubt ist:

```
comp.os.linux  tale@uunet.com yv
all       any       nq
```

In diesem Beispiel werden alle gültigen Gruppen unter `comp.os.linux` erstellt (`y`), und der Benutzer `news` wird davon in Kenntnis gesetzt (`v`). Alles andere wird stillschweigend (`q`) ignoriert (`n`). Es reicht die letzte Zeile, wenn man alle Gruppen von Hand erstellen möchte.

Alternativen zu NewsX sind `suck` oder `slurp` in Verbindung mit `postit`. Slurp benutzt den NNTP-Befehl NEWNEWS, der viele Newsserver schwer belastet.

## 5.4 Wie richte ich Leafnode ein?

Eine vollständig andere Lösung ist die Installation des integrierten Pakets `leafnode`. Es kümmert sich um alle Aufgaben, die ein eigener Newsspool mit sich bringt und ist leicht einzurichten. Es ist erhältlich über:

```
http://www.troll.no/freebies/leafnode.html
```

Wie bei C News sollten man alle Wartungsaufgaben als Benutzer `news` durchführen.

Das Hauptverzeichnis für `leafnode` ist `/usr/lib/leafnode`. Um es zu installieren, gibt man ein:

```
cd /usr/lib/leafnode
tar -xzvf leafnode-0.8.tgz
cd leafnode-0.8
make
su
make install
```

Während man noch als `root` eingeloggt ist, ändert man die Zeile in `inetd.conf`, die NNTP behandelt:

```
nntp  stream  tcp  nowait  news  /usr/sbin/tcpd /usr/local/sbin/leafnode
```

Aktivieren sie diese Änderung durch:

```
killall -HUP inetd
```

Durch Eingabe von `exit` kehrt man zum Benutzer `news` zurück. Man ändert in `/usr/lib/leafnode/config` die Zeile, die den NNTP-Server angibt, in unserem Fall:

```
server = news.acme.net
```

Leafnode kümmert sich um alles, wenn man als Benutzer `news` per `crontab -e` folgenden Befehl einträgt:

```
# täglich alte Nachrichten verfallen lassen
0 4 * * * /usr/local/sbin/texpire
```

Den Austausch von News erledigt man ebenfalls als Benutzer `news` durch folgenden Befehl, wobei natürlich bereits eine PPP-Verbindung bestehen muß:

```
/usr/local/sbin/fetch
```

Benutzer, die News lesen wollen, sollten dann der Vorgehensweise im Abschnitt 5.1 folgen, nur daß sie den lokalen Rechner eintragen, d.h.:

```
export NNTPSERVER=localhost
```

Das sollte schon alles sein. Der erste `fetch` überträgt eine Liste erhältlicher Newsgruppen. Leafnode überwacht dann, welche Gruppen von den Benutzern angefordert werden, und paßt sich daran beim *nächsten* Start an.

Eine Alternative zu `leafnode` ist `nntpcache`, das hier erhältlich ist:

```
ftp.suburbia.net:/pub/nntpcache/nntpcache.tgz
```

# 6 Automatisierung

Die automatische Erledigung von News und E-Mail ist unter Linux ziemlich leicht einzurichten.

Als Allererstes sollte man die Datei `/usr/lib/ppp/ppp-on` erstellen, die die Verbindung zum ISP herstellt. Oft enthält diese Datei nur folgendes:

```
/usr/sbin/pppd
```

Weitere Angaben stehen in der Datei `/etc/ppp/options`:

```
connect "/usr/lib/ppp/chat -v -f /etc/ppp/chatscript"
crtscts
modem
defaultroute
asyncmap 00000000
user dirk
/dev/modem 38400
```

Um die Verbindung zu beenden, benutzt man die mitgelieferte Version von `/usr/lib/ppp/ppp-off`.

Hat man überprüft, daß diese zwei Skripte funktionieren, dann muß man sich Skripte schreiben, die die verschiedenen Aufgaben erledigen. Das Skript, das E-Mail holt, wurde schon beschrieben und es wird angenommen, daß es als `/home/dirk/pop` gespeichert ist.

Ein Skript für den Austausch von E-Mail kann dann als Datei `/root/mail` so aussehen:

```
#! /bin/sh
#
# E-Mail - Austausch
# timeout von 10 Minuten:
TIMEOUT=600
DT=10

# sendmail anwerfen:
sendmail -q &

# Post holen:
su dirk -c /home/dirk/pop

# warten, bis sendmail fertig ist:
t=0
while ! mailq | grep -q "Mail queue is empty"; do
    t=$[$t+$DT]
    if [ $t -gt $TIMEOUT ] ; then
     echo "sendmail -q timeout ($TIMEOUT).."
     exit 1
    fi
```

```
    sleep $DT
done

exit 0
```

Das Skript zum Austausch von News kann als Datei `/usr/lib/news/news` abgelegt werden:

```
#!/bin/sh
#
# News - Austausch
# muß als Benutzer news gestartet werden
cd /usr/lib/news

# ausgehende Nachrichten sammeln (C News):
/usr/lib/newsbin/input/newsrun < /dev/null

# News austauschen:
/usr/lib/newsbin/newsx acme news.acme.net

# und die eingehenden Nachrichten intern übergeben:
/usr/lib/newsbin/input/newsrun < /dev/null
```

Es fehlt noch ein Skript, daß die einzelnen Teile verbinden. Es kann als Datei `/root/news+mail` gespeichert werden:

```
#!/bin/sh
#
# Austausch von News und E-Mail
# muß als Benutzer root gestartet werden
#
if ! /usr/lib/ppp/ppp-on; then
    exit 1
fi
trap "/usr/lib/ppp/ppp-off" 1 2 3 15

# News und E-Mail austauschen:
/root/mail &
su news -c ~news/news
wait

# aufhängen..
/usr/lib/ppp/ppp-off

# eingehende Nachrichten intern übergeben (C News):
su news -c /usr/lib/newsbin/input/newsrun < /dev/null &

exit 0
```

Es ist ziemlich einfach, eine Erweiterung zum obigen Skript zu erstellen, so daß nur dann eine Verbindung hergestellt wird, wenn es ausgehende E-Mail und News gibt. Nennen wir es `/root/news+mail.cond` und beachten, daß der Name des ausgehenden News-Spools angepaßt werden muß:

```
#!/bin/sh
#
# Austausch von News und Email, wenn ausgehende News oder
```

```
# E-Mail vorhanden sind.
# (C News Spool)
if [ -s /var/spool/news/out.going/acme/togo ] ||
    ! ( mailq | grep -q "Mail queue is empty"); then
     /root/news+mail
fi
```

Bleibt noch übrig, anzugeben, wann das alles passiert. Dazu führt man als root den Befehl `crontab -e` aus. Nehmen wir an, wir wollen den Austausch von News und E-Mail morgens um 7 Uhr durchführen und dann jede vierte Stunde, wenn es ausgehende E-Mail und News gibt:

```
00 7             * * *   /root/news+mail
00 11,15,19,23   * * *   /root/news+mail.cond
```

Man sollte sich versichern, daß alle Teile durchprobiert sind, bevor man sie verbindet. Später kann man dann weitere Aufgaben hinzufügen, wie das Stellen der Uhr über `ntpdate` und das automatische Aktualisieren von lokal geführten WWW- und FTP-Dateien beim ISP.

Wenn man es bevorzugt, kann man das Verfahren auch umdrehen. Sobald eine PPP-Verbindung hergestellt wird, wird das Skript `/etc/ppp/ip-up` ausgeführt. Hier kann man alles hineinschreiben, was man braucht, um den Austausch von E-Mail und News durchzuführen. Weitere Details finden sich in `man pppd`.

Es ist auch möglich, PPP-Verbindungen automatisch aufbauen zu lassen, sobald Daten über das Netz gehen. Dies ist in vieler Hinsicht eine elegantere Lösung, hängt aber sehr von einer guten Konfiguration ab, damit nicht zu viele und damit teure Verbindungen hergestellt werden. Weitere Informationen findet man in:

```
http://www.dna.lth.se/~erics/diald.html
```

Das Programm `diald` ist hier erhältlich:

```
metalab.unc.edu:/pub/Linux/system/network/serial/
```

Am gleichen Ort findet man auch weitere Variationen zum Thema PPP-Verbindungen.

# 7 Letzte Worte

## 7.1 Was sollte ich noch wissen?

- Verschiedene Fehlermeldungen des Systems werden normalerweise als interne E-Mail an bestimmte Accounts verschickt. Damit diese auch gelesen werden, sollte man die Datei `/etc/aliases` erzeugen. Bei jeder Änderung muß man den Befehl `newaliases` ausführen. Folgendes Beispiel sollte in den meisten Fällen ausreichen:

  ```
  PostMaster: root
  ftp: root
  news: root
  usenet: root
  FaxMaster: root
  fax: root
  WebMaster: root
  MAILER.DAEMON: root
  ```

- Man kann viele Programme für Linux auf der *Sunsite* finden, die normalerweise gut besucht ist. Es gibt aber viele Mirrors, und jedesmal, wenn es einen Bezug auf `metalab.unc.edu:/pub/Linux/...` gibt, sollte man versuchen, einen nahen Mirror zu benutzen, z.B. `ftp.nvg.unit.no:/pub/linux/metalab/...`.
- Wenn man von Yarn überwechselt, sollte man dessen Folder mit `yarn2mf` in Standard-Folder umwandeln können:

  ```
  ftp.sn.no:/user/egilk/yarn2mf.zip
  ```

# 8 Wo erfahre ich mehr?

Das Buch *Linux Network Administrator's Guide* von Olaf Kirch aus dem *Linux Documentation Project* ist so ziemlich Pflicht für jeden, der irgend etwas in Verbindung mit TCP/IP und Internet anfangen will.

```
metalab.unc.edu:/pub/Linux/docs/linux-doc-project/network-guide/
```

In der Dokumentation zu jedem Softwarepaket findet man normalerweise alle genaueren Informationen, wenn auch nicht immer den Überblick. Als erstes sollte man in den Manual Pages nachsehen, z.B.:

```
man pppd
```

Manchmal befinden sich Dokumentationen zu bestimmten Programmen auch im Verzeichnisbaum `/usr/doc`, der aber nicht immer gut strukturiert ist.

Folgende HOWTOs sind von großer Bedeutung:

- Das *Installation HOWTO* beschreibt die Installation von Linux selbst.
- Das *NET-3 HOWTO* ist eine sehr umfangreiche Dokumentation der Netzwerkunterstützung von Linux.
- Das *Mail HOWTO* erklärt die Konfiguration verschiedener Werkzeuge. Es sollte vieles schon erledigt sein, wenn man eine Standarddistribution installiert hat.
- Das *News HOWTO* beschreibt, wie man einen (konventionellen) News-Spool einrichtet.
- *Tiny-News* behandelt eine weitere Möglichkeit, News zu erhalten.
- Das *PPP HOWTO* beinhaltet eine gute Beschreibung der Probleme, auf die man beim Einrichten einer PPP-Verbindung treffen kann.
- Das *Serial HOWTO* enthält alles, was man über die Einrichtung von seriellen Schnittstellen wissen muß.
- *Mail-Queue* beschreibt, wie man `sendmail` so konfiguriert, daß es ausgehende Mail immer in die Warteschlange stellt, aber lokale Post sofort zustellt.