Linux ISP Hookup HOWTO

Egil Kvaleberg, egilk@sn.no v1.0, 9 March 1996

This document describes how to use Linux to connect to an Internet Service Provider via a dial-up modem connection. As well as the basic dial-up procedure and IP establishment, email and news handling is covered.

Contents

1 Introduction

This description has been made to answer a few questions about how dial-up ISP (Internet Service Provider) subscribers may configure and use Linux.

To ease the burden for those who will connect their Linux based machines to an ISP for the first time, an attempt has been made to cover most issues encountered. This quite unavoidably will create a certain degree of overlap with other Linux Howto-documents and LDP books. Reference should be made to these documents to provide better understanding and detail.

Much of the existing documentation is targeted towards users with a certain degree of experience, and first time users will often have trouble sorting out the relevant information.

To simplify, the examples given will assume the following:

- User name: *dirk*
- Password: *PrettySecret*
- Internet service provider: *acme.com*
- Email server: *mail.acme.com*
- News server: *news.acme.com*
- Name server: *193.212.1.0*
- Phone number: *12345678*

Our "dirk" will be calling his machine "roderick".

All references in the table above should naturally be replaced by whatever is valid for the ISP one is using. Often, just a minimum of changes will otherwise be required for users with different ISPs. I would like to be informed about what problems you encounter on this account.

1.1 New versions of this document

New versions of this document will be periodically posted to *comp.os.linux.answers.* They will also be added to the various anonymous FTP sites who archive such information, including:

`ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO`

In addition, you should generally be able to find this document on the Linux WorldWideWeb home page at

`http://sunsite.unc.edu/mdw/linux.html`

Finally, the very latest version of this document should also be available in various formats from:

`ftp://ftp.sn.no/user/egilk/ISP-HOWTO.txt`

`ftp://ftp.sn.no/user/egilk/ISP-HOWTO.ps.gz`

`ftp://ftp.sn.no/user/egilk/ISP-HOWTO.tar.gz`

`http://www.sn.no/ egilk/ISP-HOWTO.html`

1.2 Feedback

All comments, error reports, additional information and criticism of all sorts should be directed to:

`egilk@sn.no`

`http://www.sn.no/ egilk/`

1.3 Disclaimer

No liability for the contents of this documents can be accepted. Use the concepts, examples and other content at your own risk. Additionally, this is an early version, with many possibilities for inaccuracies and errors.

One of many possible setups will be described. In the Linux world, there is usually a number of ways in which to accomplish things. Paragraphs containing hints to alternatives are marked by **ALT.** Please also note that FTP-references often will change slightly as new versions of programs arrive.

As far as I know, only programs that under certain terms may be used or evaluated for personal purposes will be described. Most of the programs will be available complete with source under GNU-like terms.

1.4 Copyright information

2 How do I connect to the rest of the world?

It will be assumed that we have installed the essential networking software modules (e.g. essential parts of the Slackware N-series), and that you have set up which serial port that is to be used for the */dev/modem.*

The default configuration will usually only allow direct access to "/dev/modem" as user "root".

To connect to ISP shell accounts directly, and to experiment with connection sequences, you may use the "minicom" program. It is pretty straight forward to use.

2.1 The basic configuration

Configuration of the machine for use on the net should be done as user "root". Before proceeding any further, ensure that the file "/etc/hosts.deny" contains the following line:

```
ALL: ALL
```

You would normally want to allow yourself, so add the following line to "/etc/hosts.allow":

```
ALL: 127.0.0.1
```

For the following, note that it is meant for those connected via PPP and with a dynamic IP address. If you have the benefit of a fixed connection, there will be some differences.

It is nice to have a name connected to the machine, a name that the dynamic IP user really can select as he or she pleases. Put the name in "/etc/HOSTNAME":

```
roderick
```

The next step is to set up the name server in "/etc/resolv.conf":

```
search acme.com
nameserver 193.212.1.0
```

The name server must be specified by a numeric IP address, and will be different from ISP to ISP. If required, you can have up to three different servers. They will be requested in the sequence in which they are listed. A "search" statement has been added to allow abbreviation of names like "news.acme.com" to "news".

A certain minimum of configuration will also be required in "/etc/hosts". Most users will be able to manage with:

```
127.0.0.1 localhost
0.0.0.0   roderick
```

Those with a fixed IP-address will obviously replace 0.0.0.0 with this.

Likewise, a minimum "/etc/networks" is:

```
loopback  127.0.0.0
localnet  0.0.0.0
```

The username and password at the ISP must be specified in "/etc/ppp/pap-secrets"

```
dirk * PrettySecret
```

For those ISPs using CHAP instead of PAP, the filename is "/etc/ppp/chap-secrets".

Finally, the nitty gritty regarding the connection procedure itself must be specified before PPP can be initiated. This is done 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
```

Details here may have to be tuned somewhat. The phone number in the third line must of course be set as required. Some users may need to replace the "ATZ" modem initialization string with something more tailored for the modem being used. The last line specifies that one is expecting the prompt "name:", and that the response should be "ppp" when it arrives. Other systems may have other login procedures.

To actually initiate a call, the PPP-protocol may be initiated by issuing the following command:

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

We should now be on-air, and stay up until the program is killed by typing a Ctrl-C. Any messages concerning the connection will be appended to the system logs. To read them, try:

```
trail /var/adm/messages
```

As long as PPP is up, you will have direct access to the Internet, and may use programs like ftp, ncftp, rlogin, telnet, finger etc. All these programs should be part of the network package.

Further information concerning PPP is also available from:

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

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

Finally, an additional word about safety. The file "/etc/inetd.conf" lists all services that your machine will offer externally. With the "/etc/hosts.deny" file we have made, no external access will be allowed. For those who need it, access must be allowed explicitly in "/etc/hosts.allow". Local traffic may be allowed by:

```
ALL: LOCAL
```

See also "man 5 hosts_access".

A final little thing. A certain confusion exists regarding the names of the POP-protocols. A definition in "/etc/services" compatible with just about everything is:

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

ALT Instead of "chatscript", one might use the much more flexible "dip". But not in connection with "diald".

ALT Those lucky enough to have a permanent TCP/IP connection via e.g. an Ethernet may safely ignore anything about PPP, and rather start concentrating about setting up their network card.

ALT Others may not have the possibility of using PPP, but may be able to use SLIP instead, for which there is support in much the same manner as for PPP. Another possibility is UUCP. Others again may have to rely on exchange of news and email be means of SOUP. A description for the latter case may be found in:

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

The TERM program is also an option. Refer to the "Term-HOWTO".

3 How do I "surf"?

If you think that text is the most important, you might want to use the Lynx Web-browser. It is available from:

`ftp://sunsite.unc.edu/pub/Linux/system/Network/info-systems/lynx-2.3.bin2.tar.gz`

If you have installed X-windows, you can also use one of the many graphical browsers. Chimera may be found at:

`ftp://sunsite.unc.edu/pub/Linux/system/Network/info-systems/chimera-1.65.bin.ELF.tar.gz`

`http://www.unlv.edu/chimera/`

Mosaic:

`ftp://sunsite.unc.edu/pub/Linux/system/Network/info-systems/Mosaic-2.7b1-aout.tgz`

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

Mozilla (Netscape):

`ftp://sunsite.unc.edu/pub/Linux/system/Network/info-systems/netscape-v11b3.tar.gz`

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

These browsers are constantly available in new and in various ways *exciting* versions.

Use and evaluation of these programs is subject to certain terms. Please observe them.

4 How do I send and receive email?

First of all, ensure that "sendmail" is installed. Sendmail sorts internal and out-bound mail, and will buffer out-bound mail until such time it is possible to forward it.

Sendmail is based on a configuration found in "/etc/sendmail.cf". An example suitable for ISP users can be found in:

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

It is if course required to have an **official** domain address for out-bound mail, something which is specified in "/etc/sendmail.cf":

```
# who I masquerade as (null for no masquerading)
DMacme.com
```

Sendmail is now configured for sending **directly** to the recipient. To avoid long and repeated connections in those cases where the connection to the receiving end is slow and irregular, is is usually nice to use ones ISP as a buffer store. This can be specified by the DS specification:

```
# "Smart" relay host (may be null)
DSmail.acme.com
```

Beware that sendmail is somewhat sensitive to handling of tab stop characters in "sendmail.cf". You might want to use the "vi" editor to ensure that these tab characters are retained unchanged.

Email reception can often be performed via the POP3 protocol, which can be initiated every time the connection is brought up. A script for testing this is:

```
sendmail -q
popclient -3 -v mail.acme.com -u dirk -p "PrettySecret" \
       -k -o /usr/spool/mail/dirk 2> $HOME/pop.log
```

This script may be started after PPP connection has been established. Beware that this script is just for testing, so ensure that the local mailbox is left untouched while it runs. The "-k" option means that the mail is **kept** in the ISP mailbox, and you are simply given a copy of the mail. You would of course want to remove this option once you are confident that your setup is working.

Beware that the password will show on the command line. This really should be fixed ASAP.

A safe and better version of this script may be found at:

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

This version of the script requires that "procmail" is installed, but that is something you'll never regret anyway:

`ftp://sunsite.unc.edu/pub/Linux/system/Mail/mailhandlers/procmail-3.10-2.tar.gz`

Procmail is a simple and flexible tool that can sort incoming email based on a large range of criteria. In addition to being able to handle automated tasks like vacation messages and such.

The user interface for reading and sending of email can be found in programs like Pine or Elm.

ALT For an ordinary dial-up ISP user it is not really necessary to have the sendmail daemon active. To reduce the resource usage, one may thus comment out any startup of sendmail, as is usually found in "/etc/rc.d/rc.M".

ALT In place of sendmail one might use the simpler "smail". You'll find a good description of it (as well as most other things mentioned here) in the "Linux Network Administrator's Guide".

ALT There is also an m4 macro package for making a fresh "sendmail.cf". For a simple installation it might be just as well to modify an existing configuration.

ALT There are also simpler although less flexible alternatives. Pine may run stand-alone as long as it is configured properly, for instance. It might even be possible to use newer versions of some web-browsers.

ALT Many are very enthusiastic regarding the Emacs companion Gnus as an email and news handler. Further information can be found at:

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

ALT An alternative to popclient is "pop-perl5". It is available from:

`ftp://sunsite.unc.edu/pub/Linux/System/Mail/pop-perl5-1.1.tar.gz`

5 News

5.1 How do I set up an online news-reader?

As long as PPP is active, it will be possible to read news *online.* There are several available programs. One can use the "rtin" or "trn", one can configure "pine", or even use a WWW-browser.

The only thing required in terms of configuration in most cases is to do (usually once and for all in the file ".profile"):

```
export NNTPSERVER=news.acme.com
```

5.2 How do I set up an offline news-reader?

To be able to read news while offline, and thus reduce phone bills and give greater flexibility, one must set up a local news-spool of one sort or the other. This requires some configuration, and there will also be a certain amount of disk space involved. After initial setup, things should run more or less by themselves, with only some attention needed from time to time.

The solution described here is based on the news-server *Cnews* and the NNTP protocol. Cnews was originally targeted towards another sort of configuration, but is flexible enough to handle our situation too. One might also use the new and popular *Inn* news server, but it might require a bit more in terms of resources. Any way, be careful **not** to install both; they don't live together easily.

It is crucial that all maintenance of news is done while logged in as user "news", and that all configuration files is placed in "/usr/lib/news". One way of handling this is, while logged in as "root" to write "su news; cd".

The most important files in the configuration are:

- "active" is an overview over active newsgroups. It is updated as required by the command addgroup, e.g. "addgroup comp.os.linux.networking y".

- "organization" should simply contain whatever you want in the "Organization:" header field, in our case:

  ```
  Dirk Gently's Holistic Detective Agency
  ```

- "mailname" should in our case be set to "acme.com".
- "whoami" is set to the name of your "site" in the "Path:" thread. In a setup as described here, using "postit", this name will never leave the machine, so you can set this to whatever you like as long as you are pretty sure it is unique. In this case "roderick".
- "sys" controls fetching and further distribution of news. We will assume the ISP in our case adds "acme.com" to the Path, and that this is the only news source we have. The example given really tells that we will accept everything that arrives, and that we will only post news to "acme.com" that it hasn't seen before, and is originally posted at our own site. In this simplified setup we assume that the decision about which newsgroups to really fetch is done by "slurp" (see below). "/all" specifies the distribution, and **must** be included. The letter "F" says that (pointers to) outgoing news articles will be collected in a file.

  ```
  ME:all/all::
  sn/acme.com:all,!junk/all:FL:
  ```

- A subdirectory for the outgoing news must be created, in our case:

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

- "mailpaths" controls posting in moderated groups, although this task may usually be left to the ISP.

Cnews needs a certain degree of daily maintenance, but this can be specified once and for all via the command "crontab -e" issued as user "news". A suggested setup follows; it can be tuned as required:

```
# maintain incoming and outgoing batches
10,40 *  * * * /usr/lib/newsbin/input/newsrun

# expire C-news, once a day
30 0  * * * /usr/lib/newsbin/expire/doexpire

# monitor and report if needed
00 2  * * sat /usr/lib/newsbin/maint/addmissing
40 3  * * * /usr/lib/newsbin/maint/newswatch
50 3  * * * /usr/lib/newsbin/maint/newsdaily
```

Newsrun moves articles in and out (twice every hour), doexpire will delete articles as they get old (every night at 00:30), and the three last commands does various supervisory and error correcting tasks.

One should also ensure that things are cleaned up when starting the machine. As user root, add the following line to "/etc/rc.d/rc.local":

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

News may be collected via the program "slurp", picking news from an NNTP-server. The program can for instance be found at:

`ftp://sunsite.unc.edu/pub/Linux/apps/comm/slurp-1.10.tar.Z`

Setting up "slurp" is quite simple. The program itself may be copied to til /usr/lib/news. You should also create a /usr/lib/news/slurp.sys that describes which newsgroups that you want to read. A simple example, collecting all linux groups in addition to rec.humor.funny:

```
news.acme.com:comp.os.linux.*,rec.humor.funny
```

Note that Slurp uses "*" instead of "all".

Additionally, a file must be created for each news-server, in our case "slurp.news.acme.com". The contents of this file reflects the latest date/time that news have been collected, and will be updated for every news batch. The file must be initialized (initial date may be set as desired):

```
960101 000000
```

To fetch articles, user "news" issues the following command (assuming communication via PPP or similar is up):

```
slurp news.acme.com
```

The option "-d" gives continuous printout to the screen.

Most ISPs will not allow posting of news by the NNTP IHAVE-procedure, only POST. This requires special handling of news posting. A suitable tool can be collected from:

`ftp://ftp.sn.no/user/egilk/postit.tar.gz`

Article transfer may then be performed by:

```
postit news.acme.com acme
```

To control disposal of articles as they get old, a file "explist" is required. The comments in this example should explain what we want to do:

```
# hold onto history lines 14 days, nobody gets >120 days
/expired/           x    14   -
/bounds/            x    0-1-120 -

# retain these for 2 months
comp.sources,comp.os.linux.all     x    60   -

# noise gets thrown away fast
junk,control             x    2    -

# default:  14 days, no archive
all                 x    14   -
```

ALT In a small news-spool, one will often not need the newsgroup "control". The traffic is **huge** compared to the usefulness. The main point is that articles will be canceled, and that groups may be created automatically. To ensure that control messages containing "newgroup" not shall mess up things for us, a file called "newgroupperm" specifies what we will allow:

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

In this example, all proper groups under comp.os.linux will be created (y), and the user "news" will be notified (v). Everything else will be silently (q) ignored (n). The last line is sufficient if you want to create all groups manually.

ALT An alternative to slurp/postit is "suck", that is not using the NNTP NEWNEWS command. This program is simple, although quite improved compared to the earliest versions. Version 2.51 will be replaced by 2.6:

`ftp://sunsite.unc.edu/pub/Linux/system/News/suck-2.5.1.tar.gz`

ALT A different solution altogether is to install the integrated package "leafnode". This will handle all tasks required for a personal news spool, and is easy to configure. It is available from:

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

6 How do I automate the connection procedure?

Automated handling of news and email is quite easy to implement in Linux.

First and foremost one should make a "/usr/lib/ppp/ppp-on" that initiates the ISP connection. Often, this file will simply contain the following:

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

Further specification will be performed in "/etc/ppp/options":

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

To end a connection, use the supplied version of "/usr/lib/ppp/ppp-off".

Having tested the functionality of these two scripts, one must then write scripts that perform the various tasks. The script to collect email has been described before, and we will here assume it is located at "/home/dirk/pop".

A script for exchange of email can then be produced in "/root/mail":

```
#! /bin/sh
#
# exchange mail
# 10 minutes timeout:
TIMEOUT=600
DT=10

# kick sendmail:
sendmail -q \&

# retrieve mail:
su dirk -c /home/dirk/pop

# wait for sendmail to terminate:
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
```

The script to exchange news may be placed in "/usr/lib/news/news":

```
#!/bin/sh
#
# exchange news
# must be run as news:
cd /usr/lib/news

#update the outgoing batch (Cnews):
/usr/lib/newsbin/input/newsrun <lt; /dev/null

#send batched news:
/usr/lib/newsbin/postit news.acme.com acme

#and get the new one:
slurp news.acme.com
```

A script to connect the various bits and pieces remains, and can be placed in "/root/news+mail":

```
#!/bin/sh
#
# exchange news and email
```

```
# must be run as root
#
if ! /usr/lib/ppp/ppp-on; then
    exit 1
fi
trap "/usr/lib/ppp/ppp-off" 1 2 3 15

#exchange news+mail:
/root/mail \&
su news -c ~news/news
wait

#disconnect..
/usr/lib/ppp/ppp-off

#update the incoming batch (Cnews):
su news -c /usr/lib/newsbin/input/newsrun <lt; /dev/null \&

exit 0
```

It is quite easy to make an extension to the above that only will establish a connection if outgoing email and news is present. Lets call it "/root/news+mail.cond", and keep in mind that the name of the outgoing news-spool must be updated to suit:

```
#!/bin/sh
#
# exchange news and email, only if outgoing news or mail
# (Cnews spool)
if [ -s /var/spool/news/out.going/acme/togo ] ||
    ! ( mailq | grep -q "Mail queue is empty"); then
     /root/news+mail
fi
```

The only thing remaining is to specify when all this is going to happen. This is done using the command "crontab -e" as root. Let us assume that we always want to exchange news and mail at 07:00 in the morning, and after that every 4th hour assuming there are outgoing email and news:

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

Ensure that every component is tested well before you connect them together. One may later add several other tasks, such as adjustment of the time of day (using ntpdate), and automatic update (mirroring) of locally maintained WWW and FTP files up to the ISP (using make and ftp).

ALT Depending on ones preferences, it is also possible to turn the process upside down. Every time a PPP link is initiated, the script "/etc/ppp/ip-up" will be started. One may here add whatever magic is required to start exchange of email and news. See "man pppd" for further detail.

ALT It is also possible to automatically connect PPP whenever network traffic is detected. This is in many ways the more elegant solution, but it is quite dependent on a good configuration to avoid frequent (and costly) connections being made. More information can be found at:

`http://www.cs.toronto.edu/ schenk/diald.html`

The "diald" utility is available from:

`ftp://sunsite.unc.edu/pub/Linux/system/Network/serial/diald-0.13.tar.gz`

At the same location one will also find other variations on PPP connections.

7 Final words

7.1 Other things I should know about?

- Various error messages in the system will normally be issued as internal email. To ensure that these will actually be read, one should create an "/etc/aliases". Remember the command "newaliases" every time you change this. An example that should cover most eventualities is:

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

- Many programs for Linux may be found at "Sunsite", which is usually quite busy. But there are many mirrors, and every time there is a reference to "ftp://sunsite.unc.edu/pub/Linux/.." one should try to use a mirror close to home, e.g. "ftp://ftp.nvg.unit.no/pub/linux/sunsite/.."
- If you happen to be migrating from Yarn, it should be possible to convert these to standard folders using the "yarn2mf" available at:

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

7.2 How do I learn more?

The *Linux Documentation Project* book called *Linux Network Administrator's Guide* by Olaf Kirch is pretty mandatory for anyone that will set up and maintain anything involving TCP/IP and Internet:

`ftp://sunsite.unc.edu/pub/Linux/docs/linux-doc-project/network-guide/nag-1.0.ascii.tar.gz`

The documentation that follows each software package will normally give you all the detailed information you need, if not always the overview. The man-pages will be the first place to look. Try for instance:

```
man pppd
```

You will also find some documentation about certain programs in the "/usr/doc" tree, although this is not always well structured.

The following HOWTOs will be highly relevant:

- **Installation-HOWTO** will get the basics sorted.
- **NET-2-HOWTO** is a very thorough description of installation and setup of the NET code. Much of this should already have been done if you use a standard Linux distribution (e.g. Slackware, RedHat, Debian). But many sections on setup and troubleshooting will be very worthwhile.
- **Mail-HOWTO** explains how to configure various tools. Again, much of this will already have been done for you when you install a standard Linux distribution.
- **News-HOWTO** is for setting up a (conventional) news spool.
- **Tiny-News** covers yet an alternative to "slurp" of picking up news.
- **PPP-HOWTO** is a good description of problems you may encounter when setting up a PPP connection.
- **Serial-HOWTO** contains everything you need to know about setting up serial ports.

7.3 Thanks to

Information here is collected from many sources. Thanks to the following that either indirectly or directly have contributed:

```
Arne Coucheron      arneco@oslonett.no
Arne Riiber         riiber@oslonett.no
Arnt Gulbrandsen    agulbra@troll.no
Bjorn Steensrud     bjorn@oslonett.no
Gisle Hannemyr      gisle@a.sn.no
Hans Amund Rosbach  haro@sesam.dnv.no
Hans Peter Verne    hpv@ulrik.uio.no
Harald T Alvestrand Harald.T.Alvestrand@uninett.no
Haavard Engum       hobbes@interlink.no
Jorn Lokoy          jorn@oslonett.no
Kenneth Tjostheim   kenneth.tjostheim@asplanviak.no
Kjell M. Myksvoll   kjell.myksvoll@fou.telenor.no
Kjetil T. Homme     kjetilho@math.uio.no
Nicolai Langfeldt   janl@math.uio.no
Ove Ruben R Olsen   Ove.R.Olsen@ub.uib.no
Steinar Fremme      steinar@fremme.no
Sverre H. Huseby    sverrehu@ifi.uio.no
Trond Eivind Glomsrod    teg@stud.imf.unit.no
Tommy Larsen        tommy@mix.hive.no
```