



PPP-Verbindung

Autor: Jörn Absatz (mail.absatz.de)

Autor: Alexander Fischer (selflinux@tbanus.de)

Layout: Matthias Hagedorn (matthias.hagedorn@selflinux.org)

Lizenz: GFDL

Die Pfadbeispiele sind in diesem Dokument wie folgt geregelt:

`pppd`, `chat` und `diald` befinden sich in `/usr/sbin`.

Die Konfigurationsdateien befinden sich in `/etc/ppp`, `/etc/diald` und `/etc/chatscripts`.

Bei anderen Linux-Distributionen können es andere Verzeichnisse sein.

Inhaltsverzeichnis

1 PPP-Verbindung von Hand herstellen: pppd und chat

2 PPP-Verbindung von Hand beenden:

3 Parameter / Optionen für pppd und chat

4 Der Parameter connect bzw. der chat-Aufruf

4.1 Chatskript /etc/chatscripts/cis2 für Compuserve (neues Netz):

4.2 Chatskript /etc/chatscripts/cis für Compuserve (altes Netz):

4.3 Chatskript /etc/chatscripts/ton für T-Online:

4.4 Chatskript /etc/chatscripts/hangup für den pppd-Parameter disconnect:

4.5 Chatskript für andere Provider: minicom

5 Der Parameter /dev/modem für pppd:

6 Der Parameter user für pppd:

7 PPP-Verbindung automatisch herstellen: diald

8 Parameter-Datei /etc/diald/diald.options

9 Die Datei /etc/ppp/options muß für diald leer sein:

10 Erfolgskontrolle: früher in /var/log/ppp.log - jetzt in /var/log/debug:

11 Zum Surfen werden Nameserver-Eintragen in /etc/resolv.conf gebraucht:

12 Weitere Informationsquellen:

1 PPP-Verbindung von Hand herstellen: pppd und chat

Um eine PPP-Verbindung herzustellen, ruft man auf: `/usr/sbin/pppd`. `pppd` ruft seinerseits auf: `/usr/sbin/chat`. `chat` stellt das Modem ein, wählt den Provider an, identifiziert sich und verlangt vom Provider, den PPP-Server zu starten. Dann gibt es zurück an `pppd`.

`pppd` verhandelt dann mit dem PPP-Server über den Aufbau einer PPP-Verbindung und bekommt schließlich zwei temporäre IP-Nummern: eine für den lokalen Rechner und eine zweite für den Rechner auf der Provider-Seite. Über diese IP-Nummern wird dann der Verkehr mit dem Internet abgewickelt. Anfragen ans Internet gehen an die IP-Nummer des Providers, Antworten aus dem Internet gehen an die lokale IP-Nummer.

2 PPP-Verbindung von Hand beenden:

Die Verbindung kann beendet werden mit dem Shell-Skript `/usr/sbin/poff`.

3 Parameter / Optionen für pppd und chat

`pppd` benötigt einige Parameter (Optionen). Es sucht diese Parameter in der Datei `/etc/ppp/options` und in der Kommandozeile. Die Kommandozeilenparameter haben Vorrang.

Die Bedeutung jedes Parameters sieht man sich mit `man pppd` an. Bei mir lautet der Aufruf von `pppd` folgendermaßen:

```
root@linux / # /usr/sbin/pppd \ /dev/modem \ 115200 \ asyncmap 0 \ noauth \
crttscts \ debug \ lock \ modem \ defaultroute \ proxyarp \ noipdefault \
lcp-echo-interval 30 \ lcp-echo-failure 4 \ ipcp-accept-local \
ipcp-accept-remote \ noipx \ connect "/usr/sbin/chat -v -f
/etc/chatscripts/cis" \ user <meine user-id> \ disconnect "/usr/sbin/chat -v
-f /etc/chatscripts/hangup"
```

Das Ganze ist eine einzige Kommandozeile. Die Schrägstriche am Ende jeder Zeile bedeuten: **Diese Zeile wird in der nächsten Zeile fortgesetzt.**

Der komplizierteste ist der Parameter `connect`. Der Inhalt dieses Parameters ist selbst wiederum eine komplette Kommandozeile mit Parametern / Optionen. Nämlich der Aufruf des Programms `chat`.

4 Der Parameter connect bzw. der chat-Aufruf

Das Programm `chat` wickelt ein Frage-Antwort-Spiel ab nach dem Muster:

Warte auf <Ausgabe vom Provider>, antworte <lokale Eingabe>,
Warte auf <Ausgabe vom Provider>, antworte <lokale Eingabe>,
Warte auf <Ausgabe vom Provider>, antworte <lokale Eingabe>, usw.

Die Liste von Fragen und Antworten kann in einer Datei z.B. `/etc/chatscripts/cis` oder in der Kommandozeile übergeben werden.

4.1 Chatskript `/etc/chatscripts/cis2` für Compuserve (neues Netz):

<code>/etc/chatscripts/cis2</code> für Compuserve (neues Netz)
<pre>" " ATZ OK ATDT25449324 CONNECT " " "Login:" cisl >Password:" classic "User ID:" "<meine user-id>/noint/go:pppconnect" >Password: " "<mein passwort>"</pre>

Diese Zeilen bedeuten:

warte auf nichts - antworte mit Modembefehl: Reset,

warte auf Meldung vom Modem: OK -antworte mit Modembefehl: Tonwahl Compuserve,

warte auf Meldung vom Modem: CONNECT - antworte mit nichts (d.h. nur Zeilenvorschub),

Die restlichen 3 Zeilen sind die Verhandlung mit dem Provider.

4.2 Chatskript `/etc/chatscripts/cis` für Compuserve (altes Netz):

<code>/etc/chatscripts/cis</code> für Compuserve (altes Netz)
<pre>' ' 'ATZ' 'OK' 'ATDT0406913666' 'CONNECT' ' ' 'Host Name:' 'CIS' 'User ID:' '<meine user id>/GO:PPPCONNECT' 'Password:' '<mein passwort>'</pre>

4.3 Chatskript `/etc/chatscripts/ton` für T-Online:

<code>/etc/chatscripts/ton</code> für T-Online
<pre>" " ATZ OK ATDT0191011 CONNECT</pre>

Hinter `CONNECT` habe ich noch ein Leerzeichen stehen, ich bin allerdings nicht sicher, ob das wirklich nötig ist.

Aus verschiedenen Newsgroup-Beiträgen weiß ich, daß es bei T-Online sehr darauf ankommt, was man nach der Meldung **Connect** (vom Modem) noch an T-Online sendet, damit dort auch tatsächlich der PPP-Server gestartet wird.

Die obige Version meines Chat-Skripts läuft zur Zeit mit Debian-Linux 2.1 (*slink*). Vor einem dreiviertel Jahr lief mit Debian-Linux 2.0 (*hamm*) folgende Version:

Debian-Linux 2.1 /etc/chatscripts/ton für T-Online
<pre>" " ATZ OK ATDT0191011 CONNECT \d\c</pre>

4.4 Chatskript /etc/chatscripts/hangup für den pppd-Parameter disconnect:

/etc/chatscripts/hangup für den pppd-Parameter disconnect
<pre>' ' '\K\d' ' ' '+++ \d' ' ' 'ATH\d' ' ' 'ATZ'</pre>

Diese Zeilen bedeuten:

warte auf nichts - lass das Modem einen BREAK senden, warte 1 Sekunde,

warte auf nichts - Modembefehl Kommandomodus, warte 1 Sekunde,

warte auf nichts - Modembefehl Aufhängen, warte 1 Sekunde,

warte auf nichts - Modembefehl Reset

4.5 Chatskript für andere Provider: minicom

Ein Terminalprogramm wie z.B. minicom kann man benutzen, um das Chatskript für andere Provider bzw. andere Einwahlprozeduren zu ermitteln. Für Compuserve muß man die Modemparameter auf 7E1 einstellen (7 Bit Data, Even Parity, 1 Stop Bit), sonst auf 8N1.

5 Der Parameter /dev/modem für pppd:

/dev/modem bedeutet das Gerät bzw. die Schnittstelle für die PPP-Verbindung. Wenn das Gerät /dev/modem noch nicht vorhanden ist, kann man es z.B. erzeugen mit `cp /dev/ttyS0 /dev/modem`, dabei steht `ttyS0` für `COM1`, `ttyS1` für `COM2`, usw.

6 Der Parameter user für pppd:

Während der PPP-Verhandlung (das ist nach Beendigung des `chat`-Programms) benötigt `pppd` noch einmal die User-ID und das Passwort. Bei Compuserve gehen also User-ID und Passwort zweimal durch die Leitung: Einmal mit `chat` und einmal mit `pppd`.

Wenn User-ID und Passwort gesendet werden müssen, nimmt `pppd` den Inhalt des Parameters `user` und schlägt dann das Passwort in der Datei `/etc/ppp/pap-secrets` nach. Dort sucht es nach einer Zeile der Form

`<user-id> * <passwort>`

Die User-ID in Compuserve besteht üblicherweise aus einer sechsstelligen Zahl, gefolgt von einem Komma, gefolgt von einer dreistelligen Zahl, Beispiel: 123456,789

Die User-ID in T-Online besteht aus der Anschlusskennung (z.B. 000011112222), gefolgt von der T-Online-Nummer (z.B. 320044445555), gefolgt von einer Raute, gefolgt von der Mitbenutzernummer, (z.B. 0001). Beispiel: 000011112222320044445555#0001

Die T-Online-Nummer ist im allgemeinen von der Telefonnummer abgeleitet. Die Anschlusskennung ist die erste und längere der beiden Zahlen, die Sie geheimhalten sollten, und die Ihnen im Brief auf dunklem Untergrund mitgeteilt wurde. (Die zweite Zahl, die Sie geheimhalten sollten, ist das Passwort).

Wenn - wie z.B. bei T-Online - die User-ID eine Raute (#) enthält, oder wenn sie Leerzeichen enthält, sollte die User-ID in Anführungszeichen eingeschlossen werden. Die Raute könnte sonst als Kommentarzeichen (**alles was jetzt kommt, ist nur Kommentar ohne Bedeutung**) bzw. das Leerzeichen als Parameter-Trennzeichen interpretiert werden.

7 PPP-Verbindung automatisch herstellen: diald

Der Ablauf ist hier so: **diald** erzeugt beim Start einen Proxy-Server, über den alle Netzanfragen gehen. Wenn der Proxy merkt, daß eine gewünschte IP-Nummer im lokalen Netz nicht vorhanden ist, läßt er **diald** eine PPP-Verbindung aufbauen. Wenn die Verbindung eine gewisse Zeit nicht benutzt wird, baut **diald** die Verbindung wieder ab und läßt das Modem aufliegen.

Zum Herstellen der Verbindung ruft **diald** erst **chat** und anschließend **pppd** auf. Die Kommandozeile zum Aufruf von **chat** und sämtliche Parameter für **pppd** werden beim Aufruf von **diald** nur an **diald** übergeben. **diald** gibt sie dann bei Bedarf weiter an **chat** bzw. an **pppd**. Der Parameter **connect** fehlt also diesmal bei den Parametern für **pppd**, statt dessen bekommt ihn **diald**.

8 Parameter-Datei /etc/diald/diald.options

/etc/diald/diald.options

```
mode ppp
device /dev/modem
connect "/usr/sbin/chat -v -f /etc/chatscripts/ton"
speed 115200
modem
lock
crtscs
local 127.0.0.2
remote 127.0.0.3
dynamic
defaultroute
include /etc/diald/standard.filter
pppd-options \
debug \
noauth \
user <meine user-id> \
ipcp-accept-local \
ipcp-accept-remote
debug 0x007D
```

9 Die Datei /etc/ppp/options muß für diald leer sein:

Falls eine Datei **/etc/ppp/options** vorhanden ist, wird ihr Inhalt **diald** stören. Irgendwo habe ich gelesen, daß diese Datei aber auch nicht ganz fehlen darf, sie muß vorhanden, aber leer sein.

10 Erfolgskontrolle: früher in /var/log/ppp.log - jetzt in /var/log/debug:

Wenn **diald** richtig läuft, sollte die Logdatei etwa so aussehen:

/var/log/debug

```
Jul 26 15:04:54 server diald[136]: Running connect (pid = 392).
Jul 26 15:04:54 server chat[392]: abort on (BUSY)
Jul 26 15:04:54 server chat[392]: abort on (NO CARRIER)
Jul 26 15:04:54 server chat[392]: abort on (VOICE)
Jul 26 15:04:54 server chat[392]: abort on (NO DIALTONE)
Jul 26 15:04:54 server chat[392]: abort on (NO ANSWER)
Jul 26 15:04:54 server chat[392]: send (ATZ^M)
Jul 26 15:04:54 server chat[392]: expect (OK)
Jul 26 15:04:54 server chat[392]: ^M
Jul 26 15:04:54 server chat[392]: OK
Jul 26 15:04:54 server chat[392]: -- got it
Jul 26 15:04:54 server chat[392]: send (ATDT0191011^M)
Jul 26 15:04:54 server chat[392]: expect (CONNECT)
Jul 26 15:04:54 server chat[392]: ^M
Jul 26 15:05:21 server chat[392]: ^M
Jul 26 15:05:21 server chat[392]: CONNECT
Jul 26 15:05:21 server chat[392]: -- got it
Jul 26 15:05:21 server chat[392]: send (\d)
Jul 26 15:05:22 server diald[136]: Running pppd (pid = 425).
Jul 26 15:05:22 server diald[425]: Running pppd: /usr/sbin/pppd
-detach modem crtscts mtu 1500 mru 1500 debug noauth user <my user id>
ipcp-accept-local ipcp-accept-remote
Jul 26 15:05:22 server pppd[425]: pppd 2.3.5 started by root, uid 0
Jul 26 15:05:22 server pppd[425]: Using interface ppp0
Jul 26 15:05:22 server pppd[425]: Connect: ppp0 <--> /dev/ttyS1
Jul 26 15:05:22 server pppd[425]: sent [LCP ConfReq id=0x1 <magic
0x9b58> <pcomp> <accomp>]
Jul 26 15:05:23 server pppd[425]: rcvd [LCP ConfReq id=0x1 <mru 1524>
<asynmap 0xa0000> <auth pap> <pcomp> <accomp> < 13 09 03 00 c0 7b 7d
a5 75>]
Jul 26 15:05:23 server pppd[425]: sent [LCP ConfRej id=0x1 < 13 09 03
00 c0 7b 7d a5 75>]
Jul 26 15:05:23 server pppd[425]: rcvd [LCP ConfAck id=0x1 <magic
0x9b58> <pcomp> <accomp>]
Jul 26 15:05:23 server pppd[425]: rcvd [LCP ConfReq id=0x2 <mru 1524>
<asynmap 0xa0000> <auth pap> <pcomp> <accomp>]
Jul 26 15:05:23 server pppd[425]: sent [LCP ConfAck id=0x2 <mru 1524>
<asynmap 0xa0000> <auth pap> <pcomp> <accomp>]
Jul 26 15:05:23 server pppd[425]: sent [PAP AuthReq id=0x1 user="<my
user id>" password="<my password>"]
Jul 26 15:05:23 server pppd[425]: rcvd [PAP AuthAck id=0x1 ""]
Jul 26 15:05:23 server pppd[425]: Remote message:
Jul 26 15:05:23 server pppd[425]: sent [IPCP ConfReq id=0x1 <addr
192.168.13.17> <compress VJ 0f 01>]
Jul 26 15:05:23 server pppd[425]: rcvd [IPCP ConfReq id=0x1 <compress
VJ 0f 01> <addr 193.158.143.45>]
Jul 26 15:05:23 server pppd[425]: sent [IPCP ConfAck id=0x1 <compress
VJ 0f 01> <addr 193.158.143.45>]
Jul 26 15:05:23 server pppd[425]: rcvd [IPCP ConfNak id=0x1 <addr
62.156.32.53>]
Jul 26 15:05:23 server pppd[425]: sent [IPCP ConfReq id=0x2 <addr
62.156.32.53> <compress VJ 0f 01>]
Jul 26 15:05:23 server pppd[425]: rcvd [IPCP ConfAck id=0x2 <addr
62.156.32.53> <compress VJ 0f 01>]
Jul 26 15:05:23 server pppd[425]: local IP address 62.156.32.53
Jul 26 15:05:23 server pppd[425]: remote IP address 193.158.143.45
Jul 26 15:05:23 server diald[136]: New addresses: local 62.156.32.53,
remote 193.158.143.45.
Jul 26 15:08:35 server diald[136]: Closing down idle link.
Jul 26 15:08:35 server pppd[425]: Terminating on signal 2.
Jul 26 15:08:35 server pppd[425]: sent [LCP TermReq id=0x2 "User
request"]
Jul 26 15:08:35 server pppd[425]: rcvd [LCP TermAck id=0x2]
Jul 26 15:08:35 server pppd[425]: Connection terminated.
Jul 26 15:08:36 server pppd[425]: Exit.
Jul 26 15:08:38 server diald[136]: Delaying 30 seconds before clear to
dial.
```


11 Zum Surfen werden Nameserver-Eintragen in /etc/resolv.conf gebraucht:

/etc/resolv.conf
<pre>#Nameserver für Compuserve: search compuserve.com nameserver 195.232.32.6 nameserver 195.232.32.7 #Nameserver für T-Online: search btX.dtag.de nameserver 194.25.2.129</pre>

12 Weitere Informationsquellen:

- * Alberts Homepage - die umfassendste Seite zu T-Online und PPP, die ich bisher gesehen habe - (<http://www.albert-rommel.de/ppp.htm>)
- * T-Online / linux / ppp (<http://www.student.informatik.tu-darmstadt.de/~kflo/III/t-online/ppp.htm>)
- * Besonderheiten des Netzzugangs bei T-Online (<http://home.t-online.de/home/Diedrich.Ehlerding/besonderheiten.html>)
- * ZDNet-Special zum Thema Linux (<http://www.zdnet.de/produkte/artikel/sw/linuxspecial/linuxspecial01-wf.html>)
- * PPP Help (<http://howto.linuxberg.com/>)
- * T-Online PPP (<http://www.et.FH-Osnabrueck.de/~std7765>)
- * T-Online PPP unter Linux (http://raum.net/A/ar_ppp/linux-text.htm)
- * Suse Linux DB: T-Online Zugang mit PPP (<http://www.suse.de/Support/sdb/toppp.html>)
- * The diald Homepage (<http://www.loonie.net/~eschenk/diald.html>)