

SelfLinux-0.10.0



## Dokumentation zu ADSL

Autor: Simon Schwarz (*s.sch-@gmx.de*)  
Formatierung: Axel Gross (*axelgross@web.de*)  
Lizenz: GFDL

## **Inhaltsverzeichnis**

**1 Warum rp-pppoe?**

**2 Die Voraussetzungen für den rp-pppoe**

**3 Installation der Netzwerkkarte**

**4 Die Verkabelung**

**5 Die Konfiguration**

**6 pap-secrets**

**7 dsl-provider**

**8 Dial-on-demand**

**9 Der erste Start**

**10 Manuelle Einwahl**

**11 Danksagung**

## **1 Warum rp-pppoe?**

Der Roaring-Penguin Treiber ist eine sogenannte Userspace-Lösung, ist also ein eigenständiges Programm und nicht Teil des Kernels. Das hat den Nachteil, dass der rp-pppoe mehr CPU-Zeit benötigt als die Kernel Lösungen, dafür aber größtenteils Kernel unabhängig ist und schon längere Zeit als stabil eingestuft wird.

## 2 Die Voraussetzungen für den rp-pppoe

- \* Ein Linuxsystem mit einem Kernel  $\geq 2.0$
- \* Kernelunterstützung für "packet sockets". (Das ist bei den meisten Standard Kernen der Fall)
- \* ppp daemon, Version 2.3.7 oder besser. Wenn dial-on-demand (Einwählen wenn gebraucht) benutzt wird, Version 2.3.10 oder besser. Für Kernel der 2.4 Serie muss der pppd 2.4 installiert werden.
- \* Eine funktionierende Ethernet-Netzwerkkarte
- \* Ein ADSL Modem
- \* ADSL Provider

### 3 Installation der Netzwerkkarte

Um die in Deutschland üblichen DSL Anschlüsse nutzen zu können, braucht man eine funktionierende Netzwerkkarte (von internen Modems abgesehen). Das kann auch eine alte 10BaseT (RJ45-Anschluss) Karte sein, solange Linux Treiber für sie verfügbar sind. Um die Karte zu installieren, müssen die entsprechenden Module geladen werden. Mithilfe von `modconf` kann der Treiber über eine Oberfläche bequem installiert werden. Sollte dieses Programm nicht zur Verfügung stehen, kann man `insmod` benutzen:

```
root@linux ~/ # insmod <modulname>
```

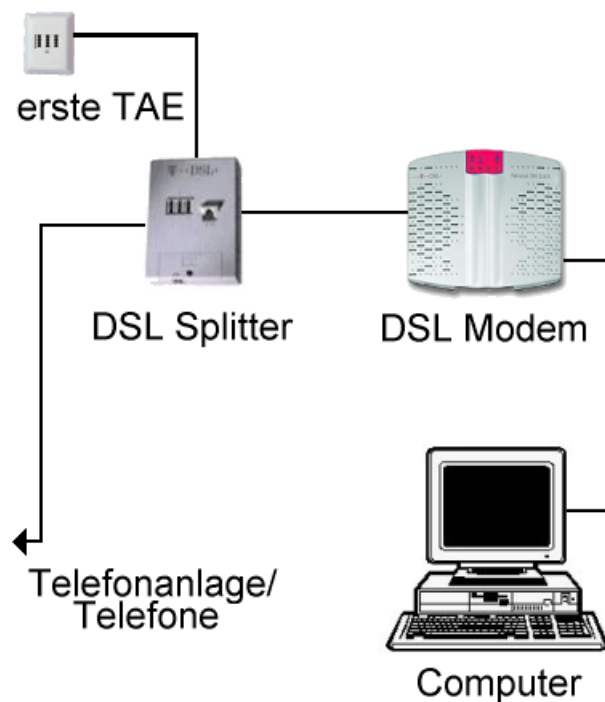
Man sollte noch einen Blick in die Datei `/etc/modules` werfen. Dort sollte nun ein Eintrag mit dem Kernel-Modul vorhanden sein, damit es bei jedem Start des Systems automatisch geladen wird.

Um die Einrichtung abzuschließen, muss noch eine eventuell vorhandene default Route gelöscht werden:

```
root@linux ~/ # route del default
```

## 4 Die Verkabelung

Als erstes muss vor dem NTBA (bei ISDN) der Splitter installiert werden, der das ISDN/Analog Signal vom DSL-Signal trennt. An den Splitter wird dann mit einem RJ-45 Netzkabel das DSL-Modem angeschlossen. Eventuell muss noch beim Splitter eingestellt werden ob das Telefonsignal Analog oder ISDN ist. Um die Verkabelung abzuschließen, verlegt man noch ein Netzkabel von der Netzwerkkarte zum DSL-Modem.



Verkabelung mit einem externen Modem

Mithilfe des folgenden Kommandos kann man überprüfen, ob die Verkabelung richtig vorgenommen wurde:

```
root@linux ~/ # pppoe -I eth1 -A
```

Wenn alles funktioniert sollte die Ausgabe wie folgt aussehen:

```
Access-Concentrator: OSNC13-nrp3
Got a cookie: 50 64 87 59 6d bf 73 00 1d 67 98 7d c2 d3 a5 32
AC-Ethernet-Address: 00:01:96:99:50:b7
```

## 5 Die Konfiguration

### 6 pap-secrets

Um den Anschluss zu konfigurieren, wird die Datei `/etc/ppp/pap-secrets` mit einem beliebigen Editor geöffnet:

```
root@linux ~/ # vi /etc/ppp/pap-secrets
```

In dieser Datei müssen nun die Einwahldaten eingetragen werden.

Achtung: Das hier aufgeführte Beispiel gilt für T-Online Anschlüsse. Bei anderen Providern können die Einwahldaten anders aussehen.

Anschlusskennung: 1111111111  
T-Onlinenummer: 2222222222  
Mitbenutzer: 0001  
Passwort: passwort

Bei der Mitbenutzernummer muss darauf geachtet werden, nur die 0001 zu verwenden, da eine Einwahl mit z.B. 0002 zusätzliche Kosten verursachen kann. Am Anfang steht die Anschlusskennung gefolgt von der T-Onlinenummer, die durch ein # von der Anschlusskennung getrennt wird. Dann wird wieder durch ein # getrennt die Mitbenutzernummer und "@t-online.de" angehängt, so dass sich folgende Zeichenkette ergibt:

```
1111111111#2222222222#0001@t-online.de
```

Diese muss zusammen mit dem Passwort in die Datei geschrieben werden:

#### Auszug der Datei pap-secrets

```
# Secrets for authentication using PAP
# client      server      secret      IP addresses
"1111111111#2222222222#0001@t-online.de" * "passwort"
```

### 7 dsl-provider

In der Datei `/etc/ppp/peers/dsl-provider` müssen nun die für den `ppp` nötigen Optionen eingestellt werden.

Die Option `user` muss mit den Einwahldaten, wie sie auch in der `pap-secrets` stehen, gefüttert werden (ohne das Passwort natürlich). Ausserdem muss das Device (z.B. `eth0`) entsprechend angepasst werden. Mit dem Befehl `man pppd` kann eine nähere Beschreibung der einzelnen Optionen aufgerufen werden.

## Auszug der Datei dsl-provider

```
# Configuration file for PPP, using PPP over Ethernet
# to connect to a DSL provider.
#
# See the manual page pppd(8) for information on all the options.

##
# Section 1
#
# Stuff to configure...

# MUST CHANGE: Uncomment the following line, replacing the user@provider.net
# by the DSL user name given to you by your DSL provider.
# (There should be a matching entry in /etc/ppp/pap-secrets with the password.)
#user myusername@myprovider.net

# Use the pppoe program to send the ppp packets over the Ethernet link
# This line should work fine if this computer is the only one accessing
# the Internet through this DSL connection. This is the right line to use
# for most people.
pty "/usr/sbin/pppoe -I eth0 -T 80 -m 1452"

# If the computer connected to the Internet using pppoe is not being used
# by other computers as a gateway to the Internet, you can try the following
# line instead, for a small gain in speed:
#pty "/usr/sbin/pppoe -I eth0 -T 80"

# An even more conservative version of the previous line, if things
# don't work using -m 1452...
#pty "/usr/sbin/pppoe -I eth0 -T 80 -m 1412"

# The following two options should work fine for most DSL users.

# Assumes that your IP address is allocated dynamically
# by your DSL provider...
noipdefault
# Comment out if you already have the correct default route installed
defaultroute

##
# Section 2
#
# Uncomment if your DSL provider charges by minute connected
# and you want to use demand-dialing.
#
# Disconnect after 300 seconds (5 minutes) of idle time.

#demand
#idle 300

##
# Section 3
#
# You shouldn't need to change these options...

hide-password
lcp-echo-interval 20
lcp-echo-failure 3
# Override any connect script that may have been set in /etc/ppp/options.
connect /bin/true
noauth
persist
mtu 1492
```

## 8 Dial-on-demand



Für all diejenigen die einen zeitabhängigen Tarif haben, ist die **Section 2** in der Datei **dsl-provider** interessant. Hier kann durch Auskommentieren von **demand** die automatische Einwahl und **idle xxx** das automatische Trennen der Verbindung aktiviert werden. Die xxx muss durch die Idle Zeit in Sekunden ersetzt werden.

## 9 Der erste Start

Bevor wir zum ersten Mal die Verbindung testen, sollte noch in der `dsl-provider` die Option `nodetach` angefügt werden, so dass der `pppd` beim Start nicht in den Hintergrund rutscht und wir die Ausgaben verfolgen können. Um alle Ausgaben verfolgen zu können, geben wir in einer extra Konsole, als root, noch folgenden Befehl ein:

```
root@linux ~/ # tail -f /var/log/messages
```

Nun können wir die Verbindung zum erstenmal testen:

```
root@linux ~/ # pppd call dsl-provider
```

Die Ausgabe des `pppd` sollte in etwa so aussehen:

```
pppd[1578]: pppd 2.4.1 started by root, uid 0
pppd[1578]: Serial connection established.
pppd[1578]: Using interface ppp0
pppd[1578]: Connect: ppp0 <--> /dev/pts/2
pppoe[1579]: PADS: Service-Name: ''
pppoe[1579]: PPP session is 31959
pppd[1578]: local IP address 62.226.75.9
pppd[1578]: remote IP address 62.225.254.169
```

Um zu testen, ob die Verbindung funktioniert, muss ein `ping` auf eine ip abgesetzt werden:

```
user@linux ~/ # ping 134.100.212.78
```

War der Ping erfolgreich, so steht auch unsere Verbindung. Nun können wir noch testen, ob der Nameserver richtig übernommen wurde:

```
user@linux ~/ # ping www.selflinux.de
```

Sollte die Namensauflösung fehlschlagen, kann man alternativ auch einen DNS-Server per Hand eintragen. Dies muss in der Datei `/etc/resolv.conf` geschehen:

### Auszug der Datei resolv.conf

```
search
nameserver 212.185.253.9
nameserver 194.25.2.129
```

Normalerweise sollte dies aber nicht nötig sein, da die Adresse des Nameservers automatisch bezogen werden sollte.



## 10 Manuelle Einwahl

Die manuelle Einwahl ist für all diejenigen interessant, die einen zeitabhängigen Tarif haben und die Kosten gerne unter Kontrolle haben (was bei Dial-on-Demand nicht der Fall ist).

Um eine manuelle Einwahl durchführen zu können, muss ein Script in `/usr/local/bin/` angelegt werden. Das Script kann man nennen wie man will, sinnvoll ist z.B. `connectadsl` o.ä.

## Connectadsl

```
#!/bin/sh
#
# Connectadsl: Skript zur Erstellung einer Verbindung zum Provider
# via ADSL/T-DSL fuer Debian GNU/Linux
#
# Verbindung starten: connectadsl start
# Verbindung stoppen: connectadsl stop
#
# written by: Michael Schlenstedt
# Michael@adsl4linux.de
# http://www.adsl4linux.de
#
# Version: 0.2a
#
#
# Einstellungen:
#
#
# Pfad zu pppd
#
PPPD="/usr/sbin/pppd"
#
# Pfad zu sudo
#
SUDO="/usr/bin/sudo"
#
#
# Ende der Einstellungen
#
#
# Wenn UID ungleich "0" (root), dann führe Skript mit sudo aus
#
#
if test -x $SUDO; then
    if test $UID -ne 0; then
        exec $SUDO $0 $*
    fi
fi
#
# Mit welchem Parameter wurde das Skript gestartet: Start oder Stop
#
case "$1" in
#
# Bei Stop wird aufgelegt
#
stop)
    echo Verbindung beenden
    killall pppd
    ;;
#
# Beim Parameter (Start) wird die Verbindung aufgebaut
#
start)
    echo Verbindung starten
    $PPPD call dsl-provider
    ;;
*)
    echo Falscher oder garkein Parameter angegeben!
    echo Bitte starten sie $0 mit dem Parameter start oder stop
    ;;
esac
```

Der große Vorteil bei der Verwendung dieses Scriptes ist, dass auch ein non-root user eine Internetverbindung


herstellen kann. Diese Funktion wird über das Programm `sudo` realisiert.

Soll nur root die Möglichkeit des Einwählens haben, so reichen die folgenden Befehle aus, und das Script wird nicht benötigt:

```
root@linux ~/ # pon dsl-provider
```

Verbindungs Abbau:

```
root@linux ~/ # poff
```

Um nun als normaler user die frisch eingerichtete Internetverbindung nutzen zu können, muss noch das Programm `sudo` installiert werden. Normalerweise sollte jede Distribution ein entsprechendes Paket bereitstellen. Sollte dies nicht der Fall sein, kann man es hier  <http://www.sudo.ws/sudo/> herunterladen.

Ist die Installation abgeschlossen, muss `sudo` nur noch konfiguriert werden. Hierzu ruft man `visudo` auf:

```
root@linux ~/ # visudo
```

Visudo lässt sich wie der Editor `vi` benutzen.

Nun trägt man wie im unteren Beispiel die User ein, die das Internet nutzen können sollen.

sudoers file	
<pre># sudoers file. # # This file MUST be edited with the 'visudo' command as root. # # See the man page for the details on how to write a sudoers file. # # Host alias specification # # User alias specification # # Cmnd alias specification # # User privilege specification root    ALL=(ALL) ALL Peter   ALL=_NOPASSWD:/usr/local/bin/connectadsl</pre>	


Loggt man sich nun mit dem Benutzer Peter ein, kann er mit den folgenden zwei Befehlen die eingerichtete Verbindung nutzen:

```
user@linux ~/ # connectadsl start
user@linux ~/ # connectadsl stop
```

Damit ist die Einrichtung des A-DSL Anschlusses abgeschlossen.

Viel Spaß im WWW (Welt Weites Warten \*g\*)

## 11 Danksagung

Viele Informationen und Texte für diesen Artikel stammen von  <http://www.adsl4linux.de>