

Package dsl

Frank Meyer *email: frank@fli4l.de*

Das fli4l-Team

23. Februar 2006

Inhaltsverzeichnis

1	Dokumentation des DSL-Paketes	3
1.1	DSL - DSL über PPPoE, Fritz!DSL und PPTP	3
1.1.1	Allgemeine Konfigurationsvariablen	3
1.1.2	OPT_PPPOE - DSL über PPPoE	7
1.1.3	OPT_PPPOE_CIRC - Mehrere DSL-Circuits über PPPoE (Experimentell)	9
1.1.4	OPT_FRITZDSL - DSL per Fritz!Card DSL	9
1.1.5	OPT_PPTP - DSL über PPTP in Österreich/Holland (EXPERIMENTAL)	10
1.1.6	OPT_POESTATUS - PPPoE-Status-Monitor auf fli4l-Console . . .	12
1.1.7	OPT_PFC - Packet-Filter-Compiler	12
A	Anhang zum DSL-Paket	13
A.1	PPPD und Active Filter	13

1 Dokumentation des DSL-Paketes

1.1 DSL - DSL über PPPoE, Fritz!DSL und PPTP

Fli4l unterstützt DSL in drei verschiedenen Varianten:

- PPPoE (externe, über Ethernet angeschlossene DSL-Modem, über die pppoe gefahren wird)
- PPTP (externe, über Ethernet angeschlossene Modem, über die pptp gefahren wird)
- Fritz!DSL (DSL über DSL-Adapter von AVM)

Man kann immer nur eine Variante auswählen, gleichzeitiger Betrieb ist leider noch nicht möglich.

Die Konfiguration dieser Varianten ähnelt sich, daher werden die allgemeinen Parameter vorab beschrieben und anschließend wird auf die Spezial-Optionen der einzelnen Varianten eingegangen. Der DSL-Zugang wird vom imond als Circuit¹ verwaltet, daher muß bei Aktivierung einer der DSL-Varianten auch der imond aktiviert werden (siehe START_IMOND (Seite ??)).

1.1.1 Allgemeine Konfigurationsvariablen

Die Pakete verwenden alle die gleichen Konfigurationsvariablen, sie unterscheiden sich nur durch den vorangestellten Paketnamen. Z.B. wird in allen Paketen der Nutzernamen verlangt, die Variable heißt lediglich je nach Paket PPPOE_USER, PPTP_USER oder FRITZDSL_USER. Im folgenden werden die Variablen ohne ihre Präfixe beschrieben, der fehlende Präfix wird durch einen Stern repräsentiert und in konkreten Beispielen wird von PPPOE ausgegangen (sie sind aber auch mit jedem anderen Präfix gültig).

***_NAME** Hier sollte ein Name für den Circuit vergeben werden - max. 15 Stellen lang. Dieser wird im imon-Client imonc angezeigt. Leerstellen (Blanks) sind nicht erlaubt.

Beispiel: PPPOE_NAME='DSL'

***_USEPEERDNS** Hiermit wird festgelegt, ob die vom Internet-Provider bei der Einwahl übergebenen Nameserver für die Dauer der Onlineverbindung in die Konfigurationsdatei des lokalen Nameservers eingetragen werden sollen.

¹Im Augenblick ist leider nur genau ein DSL-Circuit möglich – will man mehrere Circuits verwenden, muß man sich mehrere Floppies bauen

Sinnvoll ist die Nutzung dieser Option also nur bei Circuits für Internet-Provider. Inzwischen unterstützen fast alle Provider diese Art der Übergabe.

Nachdem die Nameserver-IP-Adressen übertragen wurden, werden die unter DNS_FORWARDERS eingetragenen Nameserver durch die vom Provider vergebenen IP-Adressen als Forwarder verwendet. Danach wird der lokale Nameserver veranlasst, seine Konfiguration neu einzulesen. Dabei gehen bis dahin aufgelöste Namen nicht aus dem Nameserver-Cache verloren.

Diese Option bietet den Vorteil, immer mit den am nächsten liegenden Nameservern arbeiten zu können, sofern der Provider die korrekten IP-Adressen übermittelt - dadurch geht die Namensauflösung schneller.

Im Falle eines Ausfalls eines DNS-Servers beim Provider werden in der Regel die übergebenen DNS-Server-Adressen sehr schnell vom Provider korrigiert.

Trotz allem ist vor jeder ersten Einwahl die Angabe eines gültigen Nameservers in DNS_FORWARDERS zwingend erforderlich, da sonst die erste Anfrage nicht korrekt aufgelöst werden kann. Außerdem wird beim Beenden der Verbindung die originale Konfiguration des lokalen Nameservers wieder hergestellt.

Standard-Einstellung: *_USEPEERDNS='yes'

- *_DEBUG Soll pppd zusätzliche Debug-Informationen ausgeben, muss man *_DEBUG auf 'yes' setzen. In diesem Fall schreibt pppd zusätzlichen Informationen über die syslog-Schnittstelle.

WICHTIG: Damit diese auch über syslogd ausgegeben werden, muss die Variable OPT_SYSLOGD (s.o.) ebenso auf 'yes' gesetzt sein.

- *_USER, *_PASS Hier sind Benutzerkennung und Passwort für den jeweils benutzten Provider anzugeben. *_USER enthält die Benutzerkennung, *_PASS das Passwort.

WICHTIG: Für einen T-Online-Zugang ist folgendes zu beachten:

Der Username AAAAAAAAAAATTTTTT#MMMM setzt sich aus der zwölfstelligen Anschlußkennung, der T-Online-Nummer und der Mitbenutzernummer zusammen. Hinter der T-Online-Nummer muß ein '#' angegeben werden, wenn die Länge der T-Online-Nummer kürzer als 12 Zeichen ist.

Sollte dies in Einzelfällen nicht zum Erfolg führen (offenbar abhängig von der Vermittlungsstelle), muß zusätzlich zwischen der Anschlußkennung und der T-Online-Nummer ein weiteres '#'-Zeichen eingefügt werden.

Ansonsten (T-Online-Nr ist 12stellig) sind keine '#'-Zeichen anzugeben.

Die Benutzerkennung muß bei T-Online mit '@t-online.de' abgeschlossen werden!

Beispiel:

```
PPPOE_USER='111111111111222222#0001@t-online.de'
```

Infos zu der Benutzerkennung bei anderen Providern finden sich in der FAQ:

- http://extern.fli4l.de/fli4l_faengine/faq.php?list=category&catnr=3&prog=1

***_HUP_TIMEOUT** Hier kann die Zeit in Sekunden angegeben werden, nach welcher die Verbindung beendet werden soll, wenn nichts mehr über die DSL-Leitung läuft.

***_CHARGEINT** Charge-Interval: Hier ist der Zeittakt in Sekunden anzugeben. Dieser wird dann für die Kosten-Berechnung verwendet.

Die meisten Provider rechnen minutengenau ab. In diesem Fall ist der Wert '60' richtig. Bei Providern mit sekundengenauer Abrechnung setzt man *_CHARGEINT besser auf '1'.

Leider wird bei DSL der Zeittakt nicht voll ausgenutzt, so wie es bei ISDN der Fall ist. Hier wird immer nach der Zeit, die in *_HUP_TIMEOUT angegeben ist, eingehängt.

Hier ist deshalb *_CHARGEINT lediglich für die Berechnung von Gebühren maßgeblich.

***_TIMES** Die hier angegebenen Zeiten bestimmen, wann dieser Circuit aktiviert werden soll und wann er wieviel kostet. Dadurch wird es möglich, zu verschiedenen Zeiten verschiedene Circuits mit Default-Routen zu verwenden (Least-Cost-Routing). Dabei kontrolliert der Daemon imond die Routen-Zuweisung.

Aufbau der Variablen:

```
PPPOE_TIMES='times-1-info [times-2-info] ...'
```

Jedes Feld times-?-info besteht aus 4 Unterfeldern - durch Doppelpunkt (':') getrennt.

1. Feld: W1-W2

Wochentag-Zeitraum, z.B. Mo-Fr oder Sa-Su usw. Sowohl die deutsche als auch die englische Schreibweise ist erlaubt. Soll ein einzelner Wochentag eingetragen werden, ist zu W1-W1 schreiben, also z.B. Su-Su.

2. Feld: hh-hh

Stunden-Bereich, z.B. 09-18 oder auch 18-09. 18-09 ist gleichbedeutend mit 18-24 plus 00-09. 00-24 meint den ganzen Tag.

3. Feld: Charge

Hier werden in Euro-Werten die Kosten pro Minute angegeben, z.B. 0.032 für 3.2 Cent pro Minute. Diese werden unter Berücksichtigung der Taktzeit umgerechnet für die tatsächlich anfallenden Kosten, welche dann im imon-Client angezeigt werden.

Feld: LC-Default-Route

Der Inhalt kann Y oder N sein. Dabei bedeutet:

- Y: Der angegebene Zeitbereich wird beim LC-Routing als Default-Route verwendet.
- N: Der angegebene Zeitbereich dient nur zum Berechnen von Kosten, er wird beim automatischen LC-Routing jedoch nicht weiter verwendet.

Beispiel (als eine lange Zeile zu lesen):

```
PPPOE_TIMES='Mo-Fr:09-18:0.049:N  
             Mo-Fr:18-09:0.044:Y  
             Sa-Su:00-24:0.039:Y'
```

Wichtig: Die bei *_TIMES angegebenen Zeiten muessen die ganze Woche abdecken. Ist das nicht der Fall, kann keine gültige Konfiguration erzeugt werden.

Wenn die Zeitbereiche aller LC-Default-Route-Circuits ("Y") zusammengenommen nicht die komplette Woche beinhalten, gibt's zu diesen Lückenzeiten keine Default-Route. Damit ist dann Surfen im Internet zu diesen Zeiten ausgeschlossen!

Noch ein ganz einfaches Beispiel:

```
PPPOE_TIMES='Mo-Su:00-24:0.0:Y'
```

für diejenigen, die eine Flatrate nutzen.

Und noch eine letzte Bemerkung zum LC-Routing: *Feiertage werden wie Sonntage behandelt.*

- *_FILTER** Fli4l legt automatisch auf, wenn während der über hangup timeout angegebenen Zeit keine Daten über das pppoe-Interface gehen. Leider wertet das Interface auch Datentransfers mit, die von außen kommen, z.B. durch Verbindungsversuche eines P2P-Clients wie eDonkey. Da man heutzutage eigentlich permanent von anderen kontaktiert wird, kann es passieren, daß fli4l die DSL-Verbindung nie beendet. Hier hilft die Option *_FILTER. Setzt man es auf yes, wird nur noch Verkehr gewertet, der von der eigenen Maschine generiert wird und externer Traffic wird komplett ignoriert. Da von draußen reinkommender Traffic in der Regel dazu führt, daß der Router oder dahinter liegende Rechner reagieren, indem sie z.B. Verbindungswünsche ablehnen, werden zusätzlich noch einige rausgehende Pakete ignoriert. Wie das genau funktioniert, kann man hier nachlesen:

- <http://www.fli4l.de/german/extern/articles/dsl-hangup.html> und
- <http://www.linux-bayreuth.de/dcforum/DCForumID2/46.html>.

Eine genauere Beschreibung des Ausdrucks und seiner Einbindung ist im Anhang zu finden - das ist aber nur interessant, wenn man Änderungen vornehmen will.

1.1.2 OPT_PPPOE - DSL über PPPoE

Für die Kommunikation über einen DSL-Anschluss ist in der Regel das PPPoE-Paket notwendig, weil die Provider keinen richtigen Router, sondern lediglich ein DSL-Modem zur Verfügung stellen. Zwischen dem fli4l-Router und dem Modem wird das Protokoll PPP benutzt, jedoch hier speziell über das Ethernet.

Dabei können eine oder zwei Ethernet-Karten im fli4l-Router zum Einsatz kommen:

- Nur eine Karte mit IP für das LAN und PPP zum DSL-Modem
- Zwei Karten: eine für IP im LAN, die andere für PPP zum DSL-MODEM

Die bessere Wahl ist die Alternative mit den zwei Ethernet-Karten. Dann sind beide Protokolle - IP und PPPoE - sauber voneinander getrennt.

Aber die Methode mit einer Ethernet-Karte funktioniert ebenso. In diesem Fall wird das T-DSL-Modem einfach mit an den Netzwerk-Hub angeschlossen. Es muss dann aber eventuell mit leichten Einbußen bei der maximalen Übertragungsgeschwindigkeit gerechnet werden.

Ob man nun eine oder zwei Karten im fli4l-Router hat, in jedem Fall sollte die Netzwerkkarte zum DSL-Modem im 10-Mbit/s-Half-Duplex-Betrieb laufen, um Kommunikationsprobleme zwischen Modem und Netzwerkkarte auszuschließen. Alle neueren PCI- aber nur einige ISA-Netzwerkkarten lassen sich auf verschiedene Modi konfigurieren. Dazu ist es notwendig, sich eine DOS-Boot-Diskette zu erstellen und das Konfigurationsstool der Karte dort mit abzuspeichern. Dann bootet man den fli4l-Router mit dieser Diskette, startet das Konfigurationsprogramm und stellt die Karte fest auf den oben genannten Modus ein. Das Konfigurationsprogramm für die Karte liegt beim Kauf auf der Treiberdiskette bei oder kann von der Webseite des Kartenherstellers heruntergeladen werden. Eventuell findet man es auch bei einer Recherche in der NIC-Datenbank:

- <http://www.fli4l.de/hilfe/nic-db.html>

Wenn man zwei Karten benutzt, sollte man die erste Karte für das LAN und die zweite für die Verbindung zum DSL-Modem verwenden.

Dabei braucht nur die erste Karte mit IP-Adressen belegt werden.

Das heißt:

```
IP_NET_N='1'                # Nur *eine* Karte mit IP-Adresse!
IP_NET_1xxx='...'           # Die üblichen Parameter
```

Bei PPPOE_ETH gibt man 'eth1' für die zweite Ethernetkarte an und definiert *keine* IP_NET_2-xxx-Variablen.

OPT_PPPOE Aktiviert die Unterstützung für PPPoE. Standard-Einstellung: OPT_PPPOE='no'.

PPPOE_ETH Name des Ethernet-Interfaces

```
'eth0' 1. Ethernet-Karte
'eth1' 2. Ethernet-Karte
...    ...
```

Standard-Einstellung: PPPOE.ETH='eth1'

PPPOE_TYPE *PPPOE* steht für die Übertragung von PPP-Paketen über Ethernet-Leitungen. D.h., die zu übertragenden Daten werden im ersten Schritt vom pppd-Daemon in ppp-Pakete und dann in einem zweiten Schritt für die Übertragung übers Ethernet nochmals in pppoe-Pakete verpackt, um dann ans DSL-Modem geschickt zu werden. Das zweite Verpacken kann durch den pppoe-Daemon oder durch den Kern erfolgen. Mittels PPPOE_TYPE wird die Art und Weise der pppoe-Paketerzeugung definiert.

Wert	Beschreibung
async	Die Pakete werden durch den pppoe-Daemon erzeugt; die Kommunikation zwischen <i>pppd</i> und <i>pppoed</i> erfolgt asynchron.
sync	Die Pakete werden durch den pppoe-Daemon erzeugt; die Kommunikation zwischen <i>pppd</i> und <i>pppoed</i> erfolgt synchron. Das führt zu einer effizienteren Kommunikation und damit zu einer geringeren Prozessorlast.
in_kernel	Die ppp-Pakete werden direkt an den Linux-Kern gereicht, der daraus pppoe-Pakete macht. Dadurch entfällt die Kommunikation mit einem zweiten Daemon und damit ein Menge Kopieraufwand, was wiederum zu geringerer Prozessorlast führt.

Tabelle 1.1: Arten der pppoe-Paketerzeugung

Jemand hat mal einen Vergleich der verschiedenen Varianten gemacht und kam auf einem Fujitsu Siemens PCD-H, P75 zu den in Tabelle 1.2 dargestellten Ergebnissen².

Fli4l	NIC	Bandbreite (down stream)	CPU-Auslastung
2.0.8	rtl8029 + rtl8139	310 kB/s	100%
2.0.8	2x 3Com Etherlink III	305 kB/s	100%
2.0.8	SMC + 3Com Etherlink III	300 kB/s	100%
2.1.7	SMC + 3Com Etherlink III	375 kB/s	40%

Tabelle 1.2: Bandbreite und CPU-Auslastung bei pppoe

PPPOE_HUP_TIMEOUT Verwendet man als PPPoE-Typ in_kernel und als Dialmode auto, kann man als Timeout never angeben. Der Router legt dann nicht mehr auf

²Die Zahlen wurden einem Posting in spline.fli4l entnommen und nicht weiter geprüft. Der Artikel trug die Message ID <caf9fk\$ala\$1@bla.spline.inf.fu-berlin.de>.

und wählt sich nach einer Zwangstrennung des Providers automatisch wieder ein. Nachträgliche Änderungen des Dialmodes ist nicht mehr möglich.

PPPOE_MTU **PPPOE_MRU** Mit diesen optionalen Variablen können die sog. **MTU** (maximum transmission unit) und die **MRU** (maximum receive unit) eingestellt werden. Optional bedeutet, die Variable muß nicht in der Konfigurationsdatei stehen, sie ist bei Bedarf durch den Benutzer einzufügen!
Normal beträgt die MTU und die MRU jeweils 1492. Diese Einstellung sollte nur in Sonderfällen geändert werden!

1.1.3 OPT_PPPOE_CIRC - Mehrere DSL-Circuits über PPPoE (Experimentell)

Will man mehrere DSL-Zugänge verwalten, kann man das über OPT_PPPOE_CIRC tun. Setzt man dieses Opt auf *yes*, kann man mehrere PPPOE-Circuits definieren. Die Anzahl wird über PPPOE_CIRC_N bestimmt, die restlichen Optionen entsprechen denen der OPT_PPPOE, es wird lediglich ein *CIRC_x* eingeschoben, z.B. PPPOE_CIRC_x_NAME statt PPPOE_NAME.

1.1.4 OPT_FRITZDSL - DSL per Fritz!Card DSL

Hier wird die Internetverbindung per Fritz!Card DSL aktiviert. Fuer die Internetverbindung wird die Fritz!Card DSL von AVM benutzt. Da die Treiber fuer diese Karten nicht der GPL unterliegen ist es nicht moeglich diese mit dem DSL Paket zu liefern. Es ist daher unbedingt nötig diese Treiber vorher von <http://www.fli4l.de/download/stabile-version/avm-treiber.html> herunter zu laden und einfach in das fli4l Verzeichnis zu entpacken.

Da diese Treiber zu gross fuer eine Diskette sind, ist es ausserdem unbedingt noetig, fli4l auf Festplatte o.ae. zu installieren, wenn man diese Treiber verwenden moechte.

Die Circuit-Unterstützung für die Fritz!Card DSL wurde mit freundlicher Unterstützung von Stefan Uterhardt (*email: zer0@onlinehome.de*) realisiert.

OPT_FRITZDSL Aktiviert die Unterstützung für Fritz!DSL. Standard-Einstellung:
OPT_FRITZDSL='no'.

FRITZDSL_TYPE Es gibt verschiedene Frititz!-Karten, ueber die eine DSL-Anbindung erfolgen kann. Die verwendete Karte wird ueber FRITZDSL_TYPE eingestellt, wobei die in Tabelle 1.3 aufgefuehrten Typen zur Verfuegung stehen.

Standard-Einstellung:

```
FRITZDSL_TYPE='fcdsl'
```

FRITZDSL_PROVIDER Mit dieser Option wird der Typ der Gegenstelle eingestellt. Moegliche Optionen sind:

Kartentyp	Verwendung
fcdsl	Fritz!Card DSL
fcdsl2	Fritz!Card DSLv2
fcdslsl	Fritz!Card DSL SL
fcdslusb	Fritz!Card DSL USB
fcdslslusb	Fritz!Card DSL SL USB

Tabelle 1.3: Fritz-Karten

U-R2, ECI, Siemens, Netcologne, oldArcor, Switzerland, Belgium, Austria1, Austria2, Austria3, Austria4

In Deutschland handelt es sich fast immer um UR-2. Siemens und ECI kommen nur bei sehr alten Anschlüssen zum Einsatz.

Für Schweiz und Belgien sollten die Optionen selbsterklärend sein und in Österreich heisst es ausprobieren.

Sollte jemand für Österreich eine bessere Beschriftung der Optionen haben möchte er diese bitte mitteilen.

Standard-Einstellung:

```
FRITZDSL_PROVIDER='U-R2'
```

FRITZDSL_MTU FRITZDSL_MRU Mit diesen optionalen Variablen können die sog. **MTU** (maximum transmission unit) und die **MRU** (maximum receive unit) eingestellt werden. Optional bedeutet, die Variable muß nicht in der Konfigurationsdatei stehen, sie ist bei Bedarf durch den Benutzer einzufügen!
Normal beträgt die MTU und die MRU jeweils 1492. Diese Einstellung sollte nur in Sonderfällen geändert werden!

1.1.5 OPT_PPTP - DSL über PPTP in Österreich/Holland (EXPERIMENTAL)

In Österreich (und anderen europäischen Ländern) wird statt PPPoE das PPTP-Protokoll verwendet. Auch hier wird eine separate Ethernet-Karte an ein PPTP-Modem angeschlossen.

Ab Version 2.0 ist der Zugang über PPTP als Circuit realisiert - mit freundlicher Unterstützung von Rudolf Hämmerle (*email: rudolf.haemmerle@aon.at*).

Bei PPTP werden zwei Karten verwendet. Hierbei sollte man die erste Karte für den Anschluß des LANs verwenden und die zweite für die Verbindung zum DSL-Modem.

Nur die erste Karte darf mit IP-Adressen belegt werden.

Das heißt:

```
IP_NET_N='1'           # Nur *eine* Karte mit IP-Adresse!
IP_NET_1xxx='...'      # Die üblichen Parameter
```

Bei PPTP.ETH gibt man 'eth1' für die zweite Ethernetkarte an und definiert *keine* IP.NET.2-xxx-Variablen.

OPT_PPTP Aktiviert die Unterstützung für PPTP. Standard-Einstellung: OPT_PPTP='no'.

PPTP_ETH Name des Ethernet-Interfaces

'eth0' 1. Ethernet-Karte
'eth1' 2. Ethernet-Karte
... ..

Standard-Einstellung: PPTP_ETH='eth1'

PPTP_MODEM_TYPE Es gibt verschiedene PPTP-Modemtypen, über die eine pptp-Anbindung erfolgen kann. Das verwendete Modem wird über PPTP_MODEM_TYPE eingestellt, wobei die in Tabelle 1.4 aufgeführten Typen zur Verfügung stehen.

Modemtyp	Verwendung
bbaa	Österreich
bcba	Österreich
xdsl	Österreich, Inode xDSL@home
mxstream	Holland

Tabelle 1.4: PPTP-Modemtypen

Standard-Einstellung:

PPTP_MODEM_TYPE='bcba'

Inode xDSL@home

Implementiert wurde die Unterstützung von Inode xDSL@home in Anlehnung an das im Supportbereich von Inode beschriebene³.

Es gibt momentan evtl. noch Probleme mit dem Erneuern der Lease für das Interface (die IP für das Interface wird über dhcp zugeteilt und muß regelmäßig neu angefordert werden) und das Auflegen und wieder Einwählen per imonc funktioniert noch nicht so richtig. Hier ist Hilfe in Form von Patches oder zur Verfügung stellung als Testkaninchen willkommen.

Bei xsdl gibt es zwei weitere Optionen für pptp:

PPTP_CLIENT_REORDER_TO Der pptp-client, der für xsdl genutzt wird, muß unter Umständen Pakete zwischenspeichern und umordnen. Normalerweise wartet er 0,3s auf ein ausstehendes Paket. Mit dieser Variabel kann man das Timeout zwischen 0.00 (gar nicht puffern) und 10.00 variieren. Die Zeiten müssen immer mit zwei Nachkommastellen angegeben werden.

³Siehe http://www6.inode.at/support/internetzugang/xdsl.home/konfiguration_ethernet_linux.html

PPTP_CLIENT_LOGLEVEL Hier kann angegeben werden, wieviel Debug-Ausgaben der pptp-client produziert. Möglich sind 0 (wenig), 1 (default) und 2 (viel).

1.1.6 OPT_POESTATUS - PPPoE-Status-Monitor auf fli4l-Console

Auch diesen PPPoE-Status-Monitor für DSL-Verbindungen hat Thorsten Pohlmann entwickelt.

Bei OPT_POESTATUS='yes' kann auf der 3. fli4l-Console jederzeit der DSL-Status eingesehen werden. Auf die 3. Console kann mit der Tastenkombination ALT-F3 gewechselt werden, zurück auf die 1. Console mit ALT-F1.

1.1.7 OPT_PFC - Packet-Filter-Compiler

Installiert den Packet-Filter-Compiler auf dem Router, siehe Anhang PPPD und Active Filter (Seite 13).

PFC benötigt man nur, wenn man am von fli4l verwendeten Filterausdruck etwas ändern will. Will man das nicht, kann man diese Option auf no lassen.

A Anhang zum DSL-Paket

A.1 PPPD und Active Filter

Für `fi4l` setzen wir den im Link angegebenen Ausdruck ein:

```
'outbound and not icmp[0] != 8 and not tcp[13] & 4 != 0'
```

und erreichen damit, daß grundsätzlich nur vom lokalen Netz ins Internet gesendete Pakete die Verbindung offen halten, mit ein paar Ausnahmen:

- *TCP-RST*: Antworten auf abgelehnte Verbindungswünsche von außen setzen den Timeout nicht zurück,
- *ICMP*: gesendete ICMP-Nachrichten setzen den Timeout ebenfalls nicht zurück, es sei denn, es wird ein Echo-Request gesendet.

Dieser Ausdruck wird normalerweise vom PPPD in einen vom Kernel verwendbaren Paket-Filter umgesetzt. Dieser würde in diesem Beispiel wie folgt aussehen:

```
#
# Expression: outbound and not icmp[0] != 8 and not tcp[13] & 4 != 0
#
(000) ldb      [0]
(001) jeq      #0x0          jt 17   jf 2
(002) ldh      [2]
(003) jeq      #0x21          jt 4    jf 18
(004) ldb      [13]
(005) jeq      #0x1          jt 6     jf 11
(006) ldh      [10]
(007) jset     #0x1fff          jt 18   jf 8
(008) ldx      4*([4]&0xf)
(009) ldb      [x + 4]
(010) jeq      #0x8          jt 18    jf 17
(011) jeq      #0x6          jt 12    jf 18
(012) ldh      [10]
(013) jset     #0x1fff          jt 18    jf 14
(014) ldx      4*([4]&0xf)
(015) ldb      [x + 17]
(016) jset     #0x4          jt 17    jf 18
(017) ret      #0
(018) ret      #4
```

A Anhang zum DSL-Paket

Dieser Code muß generiert werden und dazu enthält der pppd normalerweise einen kleinen Compiler, der Ausdrücke in den entsprechenden Code für den Paketfilter des Kerns umwandelt. Dieser Code würde denn pppd-Daemon jedoch um 100Kbyte vergrößern und damit sinnlos Platz auf der Floppy verschwenden. Deshalb wurde beschlossen, den Compiler aus dem pppd rauszunehmen und separat zur Verfügung zu stellen (pfc). Der pppd liest dann lediglich den offline compilierten Code und übergibt ihn an den Kern. Der vorcompilierte Filterausdruck liegt unter `opt/etc/ppp/filter`.

Will man den von fli4l verwendeten Ausdruck verändern, muß man den Paketfiltercode neu generieren. Dazu benötigt man den packet filter compiler. Zum Compilieren setzt man entweder `OPT_PFC` auf `yes` und compiliert den Ausdruck auf dem Router oder man übersetzt das unter `src/base` liegende `pfc.c` auf einem Unix-System (dazu benötigt man `libpcap`) und übersetzt den Ausdruck dann auf diesem System. Das Ergebnis (eine Menge Zeilen mit jeweils 4 Zahlen) kopiert man dann nach `opt/etc/ppp/filter` und baut eine neue floppy.

Index

FRITZDSL_CHARGEINT, 5
FRITZDSL_DEBUG, 4
FRITZDSL_FILTER, 6
FRITZDSL_HUP_TIMEOUT, 5
FRITZDSL_MRU, 10
FRITZDSL_MTU, 10
FRITZDSL_NAME, 3
FRITZDSL_PASS, 4
FRITZDSL_PROVIDER, 9
FRITZDSL_TIMES, 5
FRITZDSL_TYPE, 9
FRITZDSL_USEPEERDNS, 3
FRITZDSL_USER, 4

OPT_FRITZDSL, 9
OPT_PPPOE, 7
OPT_PPPOE_CIRC, 9
OPT_PPPOE_CIRC_N, 9
OPT_PPPOE_CIRC_x_CHARGEINT, 9
OPT_PPPOE_CIRC_x_DEBUG, 9
OPT_PPPOE_CIRC_x_ETH, 9
OPT_PPPOE_CIRC_x_FILTER, 9
OPT_PPPOE_CIRC_x_HUP_TIMEOUT,
9
OPT_PPPOE_CIRC_x_MRU, 9
OPT_PPPOE_CIRC_x_MTU, 9
OPT_PPPOE_CIRC_x_NAME, 9
OPT_PPPOE_CIRC_x_PASS, 9
OPT_PPPOE_CIRC_x_TIMES, 9
OPT_PPPOE_CIRC_x_USEPEERDNS, 9
OPT_PPPOE_CIRC_x_USER, 9
OPT_PPTP, 11

PPPOE_CHARGEINT, 5
PPPOE_DEBUG, 4
PPPOE_ETH, 7
PPPOE_FILTER, 6
PPPOE_HUP_TIMEOUT, 5, 8
PPPOE_MRU, 9
PPPOE_MTU, 9
PPPOE_NAME, 3
PPPOE_PASS, 4
PPPOE_TIMES, 5
PPPOE_TYPE, 8
PPPOE_USEPEERDNS, 3
PPPOE_USER, 4
PPTP_CHARGEINT, 5
PPTP_CLIENT_LOGLEVEL, 11
PPTP_CLIENT_REORDER_TO, 11
PPTP_DEBUG, 4
PPTP_ETH, 11
PPTP_FILTER, 6
PPTP_HUP_TIMEOUT, 5
PPTP_MODEM_TYPE, 11
PPTP_NAME, 3
PPTP_PASS, 4
PPTP_TIMES, 5
PPTP_USEPEERDNS, 3
PPTP_USER, 4