

Paquetage DSL

Version 3.10.18

Frank Meyer
courriel: frank@fli41.de

L'équipe fli4l
courriel: team@fli41.de

15 septembre 2019

Table des matières

1. Documentation du paquetage DSL	3
1.1. DSL - DSL pour PPPoE, Fritz!DSL et PPTP	3
1.1.1. Variables de configuration générales	3
1.1.2. OPT_PPPOE - DSL avec PPPoE	6
1.1.3. OPT_PPPOE_CIRC - Plusieurs accès DSL avec PPPoE (Expérimental)	8
1.1.4. OPT_FRITZDSL - DSL avec carte Fritz!DSL	8
1.1.5. OPT_PPTP - DSL avec PPTP pour l'Autriche/les Pays-Bas (Expéri- mental)	9
1.1.6. OPT_POESTATUS - Moniteur pour l'état du PPPoE sur la console fli4l	11
A. Annexe du paquetage DSL	12
A.1. DSL - PPPD et filtre actif	12
Table des figures	13
Liste des tableaux	14
Index	15

1. Documentation du paquetage DSL

1.1. DSL - DSL pour PPPoE, Fritz !DSL et PPTP

fli4l supporte la connexion DSL avec trois variantes différentes :

- PPPoE (Modem-DSL externe, avec un raccordement Ethernet, il supporte le protocole pppoe)
- PPTP (Modem externe, avec un raccordement Ethernet, il supporte le protocole pptp)
- Fritz !DSL (DSL avec adaptateur-DSL de AVM)

On peut choisir une seule connexion DSL, l'exploitation de plusieurs connexions différentes n'est malheureusement pas encore possible.

La configuration de ces différentes variantes se ressemble, c'est pour cela que nous allons décrire d'abord les variables générales, ensuite nous évoquerons séparément les variables spécifiques avec leurs options. Imond administre l'accès-DSL comme un Circuit ¹, lorsque l'on active l'une des variante-DSL vous pouvez activer imond (voir `START_IMOND` (Page ??)).

1.1.1. Variables de configuration générales

Les paquetages utilisent les mêmes variables de configuration, ils se distinguent uniquement par leurs noms de paquetages placés en tête. Par ex. dans tous les paquets on demande le nom d'utilisateur, la variable s'appelle selon le paquetage `PPPOE_USER`, `PPTP_USER` ou `FRITZDSL_USER`. Par la suite, les variables seront décrites sans leur préfixe, le préfixe manquant sera remplacé par une étoile donc dans l'exemple concret la partie manquante est `PPPOE` (cela est valable pour tous les autres préfixes).

* **_NAME** Dans cette variable nous devons donner un nom pour le circuit - un maximum de 15 caractères. il sera affiché dans le Client-imonc. Le nom ne doit pas contenir d'espaces.

Exemple : `PPPOE_NAME='DSL'`

* **_USEPEERDNS** On détermine ici si les fournisseurs d'accès Internet utilisent un serveur de nom (ou DNS), nous devons enregistrer ce serveur de nom dans notre réseau local pour la durée de connexion Internet.

Logiquement cette option est uniquement utilisée pour la connexion au fournisseur d'accès Internet. En même temps, presque tous les Fournisseurs supportent ce type de transfert. Vous devez enregistrer les adresses-IP du serveur de nom (ou DNS) fournies par votre FAI dans le fichier `base.txt` dans la variable `DNS_FORWARDERS` et vous devez supprimer le serveur de nom qui est configuré sur votre PC du réseau local. Ensuite vous devez mettre à la place, l'adresse IP de votre routeur. Avec ce réglage la résolution des noms ne se perd pas dans le cache du serveur de nom.

1. Malheureusement en ce moment un seul Circuit-DSL est possible – Si vous voulez utiliser plusieurs Circuits, il faut construire plusieurs média

Cette option offre l'avantage de toujours pouvoir travailler avec un serveur de nom le plus proche, dans la mesure où le fournisseur d'accès à une adresse IP correcte - ainsi, la résolution de nom sera plus rapide.

En cas d'une défaillance d'un serveur de DNS du fournisseur d'accès, en règle général, on pourra corriger plus rapidement la transmission des adresses des serveurs DNS du fournisseur d'accès.

Malgré tout, il est nécessaire d'indiquer un serveur de nom valide dans la variable `DNS_FORWARDERS` du fichier `base.txt` pour se connecter, autrement lors de la première connexion Internet la demande ne pourra pas être résolue correctement. En outre, la configuration originale du serveur de nom local est restaurée à la fin de la connexion.

Configuration standard : `*_USEPEERDNS='yes'`

- * **_DEBUG** Pour avoir des informations de débogages supplémentaires par exemple sur `pppd`, vous devez activer la variable `*_DEBUG` régler celle-ci sur `'yes'`. Ces informations supplémentaires sur `pppd` seront enregistrées avec interface `syslogd`

IMPORTANT : pour que le démon `syslogd` fonctionne, il faut activer la variable `OPT_SYSLOGD` régler celle-ci sur `'yes'`

- * **_USER, *_PASS** Dans ces variables on indique les données du Fournisseur d'accès. `*_USER` l'identification de l'utilisateur, `*_PASS` le mot de passe.

ATTENTION : pour un accès au FAI T-Online (pour l'Allemagne) il est à noter :

Le nom d'utilisateur `AAAAAAAAAAAAATTTTTT#MMMM` est composé, du numéro co-utilisateur, puis du numéro T-online de 12 chiffres et de l'identification. Le dernier chiffre du numéro T-Online doit se terminer par `'#'` si le numéro de T-Online ne comporte pas les 12 chiffres.

Avec ça si cela ne fonctionne pas ! (évidemment cela peut provenir du centrale téléphonique), le caractère `'#'` doit être placé entre le numéro T-Online et l'identification.

Évidemment si le (numéro T-Online comporte les 12 chiffres) il n'y a pas besoin de mettre le caractère `'#'`.

Le nom d'utilisateur de T-Online doit se terminer par `'@t-online.de'`

Exemple :

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

Des infos sur la configuration des autres fournisseurs d'accès se trouvent dans la FAQ :

— http://extern.fli4l.de/fli4l_faqengine/faq.php?list=category&catnr=3&prog=1

- * **_HUP_TIMEOUT** On peut paramétrer dans cette variable le Timeout (ou le temps d'arrêt) en seconde, s'il y a aucune transmission sur le circuit DSL (ou Internet), la connexion se coupera. Vous disposez des paramètres, pour régler le délai `'0'` ou aucun délai `'never'` - Si vous sélectionnez `'never'` le routeur a en plus une nouvelle contrainte, c'est de raccrocher immédiatement. Les modifications dans mode Dial ne sont pas possibles - Le réglage doit être sur `'auto'` et doit y rester. Le paramètre `'never'` est uniquement utilisé pour PPPOE et FRITZDSL.

- * **_CHARGEINT** On utilise cette variable pour placer une unité de temps : on paramètre ici le coût par unité téléphonique en seconde. Pour calculer le prix total des communications. En Allemagne les FAI les plus gros, facture l'unité Tél. exactement à la minute, le paramètre correct dans la variable est donc `'60'`. Certain FAI facture l'unité exactement en seconde dans ce cas la variable `*_CHARGEINT` sera de `'1'`.

1. Documentation du paquetage DSL

Malheureusement, les unités de temps DSL ne sont pas exploitées pleinement, comme avec ISND. c'est après le temps paramétré ici *_HUP_TIMEOUT que la transmission se coupe.

C'est pour cette raison que la variable *_CHARGEINT sert uniquement à calculer le prix des communications

***_TIMES** Dans cette variable on paramètre temps d'activation et d'arrêt de la connexion, ainsi que le prix de l'unité Tél. Il est possible d'activer des circuits 'Default-Route' différents et aussi l'utilisation de (Least-Cost-Routing). Le contrôle du routage s'effectue avec le démon (ou programme) imond.

Structure de la variable :

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

Il y a dans chaque times-?-info 4 sous-paramètres - ces sous-paramètres sont séparés par deux points (':').

1. Sous-paramètre : W1-W2

On indique ici les périodes des jours ouvrables, par ex. Mo-Fr ou Sa-Su, il est possible de décrire les jours en Anglais ou en Allemand. Si l'on paramètre un seul jour, il sera écrit W1-W1 par ex. Su-Su.

2. Sous-paramètre : hh-hh

On indique ici la période horaire, par ex. 09-18 ou 18-09. De 18-09 est synonyme à 18-24 plus 00-09. de 00-24 correspond à toute une journée.

3. Sous-paramètre : Charge

On indique ici le prix par minute de connexion ou par unité téléphonique en euro, par ex. 0.032 correspond à 3.2 Centimes par minute. Les unités téléphonique, sont calculées en tenant compte du temps de conversion pour un coût réel, et seront alors affichées dans le client-imonc.

4. Sous-paramètre : LC-Default-Route

Le contenu de ce sous-paramètre peut être Y ou N. cela signifie :

Y : Autorise la plage horaire et LC-Routing (ou calcul des frais) avec Default-Route (ou routage par défaut).

N : Autorise la plage horaire et le calcul des frais Tél automatiquement avec LC-Routing, il n'est pas utilisé pour autre chose.

Exemple (lire cet exemple comme une seule ligne) :

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

Important: les paramètres de la variable *_TIMES doit couvrir toute la semaine. si ce n'est pas le cas, aucune connexion valide ne peut se produire.

Important: Si vous avez placé le paramètre ("Y") pour LC-Default-Route-Circuits et que vous n'avez pas réglé la semaine complète, Il aura des interruptions dans la période de la semaine avec Default-Route. Alors il sera impossible de surfer sur Internet pendant ces périodes !

Encore un exemple simple :

PPPOE_TIMES='Mo-Su:00-24:0:0:Y'

Cette exemple est pour ceux qui utilise un Flatrate (ou forfait d'accès internet illimité). Encore une dernière remarque pour le LC-Routing : *Les jours fériés sont traités comme un dimanche.*

- * **_FILTER** fli4l se coupe automatiquement, si aucun transfert de donné ne se fait sur l'interface pppoe pendant le temps paramétré dans timeout. Malheureusement, l'interface, évalue également les transferts de données, qui viennent de l'extérieur, par exemple, des tentatives de connexion avec des Clients-P2P comme eDonkey. A présent on est contacté en permanence par d'autres clients externes, il peut arriver que le routeur fli4l ne se coupe jamais.

Avoir la possibilité de filtrer le trafic et de couper la connexion, même si quelqu'un tente de se connecter.

Si on paramètre la variable *_FILTER sur yes. seul le trafic produit par sa propre machine est générée, le trafic qui vient de l'extérieur est ignoré complètement. Lorsque le trafic entrant est généré, le routeur se trouvant entre Internet et les ordinateurs réagit, et rejette celui-ci par ex. une demande de connexion, de plus les quelques paquets sortant seront ignorés. On peut voir le fonctionnement exact ici :

— <http://www.fli4l.de/hilfe/howtos/basteleien/hangup-problem-loesen/> et

— <http://web.archive.org/web/20061107225118/http://www.linux-bayreuth.de/dcforum/DCForumID2/46.html>.

Une description plus précise se trouve dans l'annexe sur l'expression et l'intégration des codes de filtrages - C'est intéressant, uniquement si vous voulez effectuer des modifications voir à la fin de la doc.

- * **_MTU *_MRU** Ces variables sont optionnelles, ont peut paramétrer le **MTU** (maximum transmission unit) et le **MRU** (maximum receive unit). Optionnelle signifie que les variables ne sont peut être pas dans le fichier de configuration, elles sont à insérer par l'utilisateur si besoin !

normalement le MTU et le MRU sont réglés à 1492. Ce réglage doit être modifier uniquement dans des cas exceptionnels ! Ces variables ne sont pas utilisées pour OPT _PPTP.

- * **_NF_MSS** Avec certains fournisseurs d'accès les effets suivants se produisent :

— Le navigateur Web reçoit un lien, mais après plus rien ne se passe.

— Fonctionne avec des petits Mail, mais pas avec des plus gros Mail.

— Avec la fonction ssh, coupure du scp après avoir établi une connexion.

Afin d'éviter ces problèmes, configurez fli4l avec le MTU par défaut. Dans certains cas, ce n'est pas suffisant, c'est pourquoi fli4l permet explicitement de composer avec MSS (message segment size) et d'indiquer une valeur fournie par le fournisseur d'accès. Si le FAI ne fournie rien, vous pouvez essayer 1412 c'est une bonne valeur de départ ; la valeur est de 40 octets de moins par rapport à la valeur MTU ($mss = mtu - 40$). Cette variable est optionnelle, ce qui signifie, que la variable n'est peut être pas dans le fichier de configuration, elle est à insérer par l'utilisateur en cas de besoin ! Ces variables ne sont pas utilisées pour OPT _PPTP.

1.1.2. OPT_PPPOE - DSL avec PPPoE

En règle générale, pour la communication via ADSL, les paquets PPPoE sont nécessaires, parce que les fournisseurs d'accès ne fournissent pas de bon routeur, mais simplement un mo-

dem DSL. Entre le routeur-fli4l et le modem fournie, on utilise le protocole PPP, en particulier sur le réseau ethernet.

Une ou deux cartes ethernet peuvent être installées dans votre routeur-fli4l, pour obtenir dans le cas échéant :

- Une seule carte avec IP pour le LAN et le protocole PPP pour le modem-DSL
- Deux cartes : une avec IP pour le LAN, et l'autre avec PPP pour le modem-DSL

La meilleure option est la solution avec les deux cartes Ethernet. Parce que les deux protocoles - IP et PPPoE - sont séparés l'un de l'autre.

Mais la méthode avec une carte Ethernet fonctionne aussi. Dans ce cas, le Modem-DSL-T est simplement raccordé au Hub ou switch du réseau. Vous pouvez avoir éventuellement une légère perte de transmission lors du débit maximum.

Si vous avez des problèmes de communication entre le modem et la carte réseau, vous pouvez essayer de ralentir la vitesse de la carte réseau, en passant éventuellement en mode Half-Duplex. Toutes les cartes réseaux PCI peuvent être configurées pour fonctionner dans les différents modes, mais seulement quelques cartes ISA. Soit vous utilisez le programme ethtool qui est dans le paquetage `advanced_networking`, soit vous créez un support de démarrage sous DOS avec les outils de configuration de la carte. Démarrer fli4l avec ce support et exécuter l'outil de configuration de la carte pour choisir et enregistrer le mode de fonctionnement plus lent pour la carte. Le programme de configuration et le pilote sont généralement fournis sur une disquette à l'achat de la carte, ils peuvent aussi être téléchargeables depuis le site Web du fabricant. Éventuellement, vous pouvez rechercher et trouver des informations sur les cartes dans wiki :

- <https://ssl.networks.org/wiki/display/f/Netzwerkkarten>

Si vous utilisez deux cartes, la première sera pour le LAN (ou réseau local) et la seconde pour la connexion au modem DSL.

Seul la première carte, doit être paramétrée avec une adresse-IP.

En d'autres termes :

```
IP_NET_N='1'                # Seule *Une* carte avec l'adresse IP~!
IP_NET_1xxx='...'           # Les paramètres habituels
```

Dans la variable `PPPOE_ETH` on indique 'eth1' et pour la deuxième carte Ethernet on définit *aucun* paramètre dans la variable `IP_NET_2-xxx`.

OPT_PPPOE Activé pour la prise en charge PPPoE. Configuration Standard :
`OPT_PPPOE='no'`.

PPPOE_ETH Nom des Interfaces-Ethernet

'eth0' 1. Carte-Ethernet

'eth1' 2. Carte-Ethernet

... ..

Configuration Standard : `PPPOE_ETH='eth1'`

PPPOE_TYPE *PPPOE* sert à transférer les paquets-PPP directement sur la ligne ethernet.

C. à d. dans la première étape les paquets-PPP de données sont transmis par le Démon-PPP, puis dans la deuxième étape ils sont transformés en paquets-pppoe et envoyés sur Ethernet pour arriver sur le modem-DSL. Dans la deuxième étape les paquets sont empaquetés par le Démon-pppoe ou par le Kernel. Au moyen de la variable `PPPOE_TYPE` on définit la manière d'empaquetage des paquets-pppoe.

Valeur	Description
async	Les paquets sont créés avec Démon-pppoe ; La communication entre <i>pppd</i> et <i>pppoed</i> est asynchrone.
sync	Les paquets sont créés avec Démon-pppoe ; La communication entre <i>pppd</i> et <i>pppoed</i> est synchrone. Cela conduit à une plus grande efficacité pour la communication, et une moindre charge du processeur.
in_kernel	Les paquets-ppp sont directement transformés par le Kernel-Linux, en paquet-pppoe. Il est alors inutile de communiquer avec le deuxième démon, donc une économie sur la transformation des paquets, et une moindre charge du processeur.

TABLE 1.1. – Type de création de paquet-pppoe

Un utilisateur a fait une comparaison des différentes variantes avec un ordinateur Fujitsu Siemens PCD-H P75, voici le tableau 1.2 Présentation des résultats².

fli4l	NIC	Bande passante (en aval)	Charge du CPU
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%

TABLE 1.2. – Bande passante et charge CPU pour pppoe

PPPOE_HUP_TIMEOUT Si on utilise `in_kernel` pour le Type-PPPoE et `dialmode auto`, on peut indiquer 'never' dans Timeout (ou temps d'arrêt). Le routeur ne raccrochera plus et se reconnectera automatiquement après une déconnexion au FAI, Toute modification ultérieure dans `dialmodes` ne sera plus possible.

1.1.3. OPT_PPPOE_CIRC - Plusieurs accès DSL avec PPPoE (Expérimental)

Si l'on veut gérer plusieurs accès DSL, on peut le faire avec la variable `OPT_PPPOE_CIRC`. Si on place cette variable sur `yes`, on peut définir plusieurs Circuits-PPPOE. On détermine le nombre dans la variable `PPPOE_CIRC_N`, les options sont identiques aux variables `OPT_PPPOE` précédente, Il faut simplement rajouter `CIRC_x`, par exemple `PPPOE_CIRC_x_NAME` au lieu de `PPPOE_NAME`.

1.1.4. OPT_FRITZDSL - DSL avec carte Fritz !DSL

Ici la connexion Internet est activé pour une carte Fritz !DSL. On utilise Fritz !Card DSL de AVM pour se connecter à Internet. Les pilotes de ces cartes ne sont pas sous licence GPL, c'est pour cela qu'il ne sont pas livrés avec le paquetage DSL. Il est nécessaire de télécharger ces pilotes à cette adresse <http://www.fli4l.de/fr/telechargement/version-stable/pilote-avm/> et de les décompresser dans le répertoire `fli4l`.

2. Les chiffres ont été prélevés d'un message des newsgroups `spline.fli4l` et n'a été objet d'aucun examen. L'article du Message portait ID <caf9fk\$ala\$1@bla.spline.inf.fu-berlin.de>.

Ces pilotes sont trop volumineux pour une disquette, il est absolument nécessaire d'installer fli4l sur un disque dur, si l'on veut utiliser ces pilotes.

la réalisation et la prise en charge des circuits pour les cartes Fritz!Card DSL a été rendue possible avec le soutien amical de Stefan Uterhardt (courriel: zer0@onlinehome.de).

OPT_FRITZDSL Activez cette variable pour la prise en charge des Fritz!Card DSL. Configuration Standard : `OPT_FRITZDSL='no'`.

FRITZDSL_TYPE Il existe plusieurs cartes Fritz!, avec lesquelles une connexion-DSL peut s'effectuer. On paramètre le type de carte dans la variable `FRITZDSL_TYPE`, les différents types disponible se trouvent dans le tableau 1.3.

Type de carte	Application
fcdsl	Fritz!Card DSL
fcdsl2	Fritz!Card DSLv2
fcdslsl	Fritz!Card DSL SL
fcdslusb	Fritz!Card DSL USB
fcdslslusb	Fritz!Card DSL SL USB
fcdslusb2	Fritz!Card DSL USBv2

TABLE 1.3. – Cartes-Fritz

Configuration Standard :

```
FRITZDSL_TYPE='fcdsl'
```

FRITZDSL_PROVIDER On règle avec cette variable le type de serveur. Les options possibles sont :

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

En Allemagne il s'agit presque toujours de UR-2. Siemens et ECI sont uniquement utilisées pour d'ancien connexion.

Pour la Suisse et la Belgique Les options sont très explicites et pour l'Autriche, il faut essayer.

Si quelqu'un a une meilleure option possible pour l'Autriche, vous pouvez la communiquer merci.

Configuration Standard :

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

1.1.5. OPT_PPTP - DSL avec PPTP pour l'Autriche/les Pays-Bas (Expérimental)

En Autriche (et dans d'autres pays européens), au lieu utiliser PPPoE ils utilisent Protocole-PPTP. Là aussi, une carte Ethernet est connectée au Modem-PPTP.

C'est à partir de la version 2.0 que l'accès au circuit avec le protocole PPTP a été réalisé - Avec le soutien amical de Rudolf Hämmelerle (courriel: rudolf.haemmerle@aon.at).

Deux cartes Ethernet sont utilisées pour la connexion PPTP. Cela devrait être la première carte pour le raccordement au réseau local (ou LAN), et la seconde pour la connexion au modem-DSL.

Seul la première carte, est paramétrée avec une adresse-IP.
En d'autres termes :

```
IP_NET_N='1'           # Seule *Une* carte avec l'adresse IP~!
IP_NET_1xxx='...'      # Les paramètres habituels
```

Dans la variable PPTP_ETH on indique 'eth1' et pour la deuxième carte Ethernet on définit *aucun* paramètre dans la variable IP_NET_2-xxx.

OPT_PPTP Activé pour la prise en charge avec une connexion PPTP. Configuration Standard : OPT_PPTP='no'.

PPTP_ETH Nom des Interfaces-Ethernet

'eth0' 1. Carte-Ethernet

'eth1' 2. Carte-Ethernet

... ..

Configuration Standard : PPTP_ETH='eth1'

PPTP_MODEM_TYPE Il existe différents types de modems PPTP, pour effectuer une connexion-pptp. Le type de modem est paramétré dans la variable PPTP_MODEM_TYPE, dans le tableau 1.4 sont énumérés les différents types pour les internautes.

Type-Modem	Utilisation
bbaa	Autriche
bcaa	Autriche
xdsl	Autriche, Inode xDSL@home
mxstream	les Pays-Bas

TABLE 1.4. – Type de Modem-PPTP

Configuration Standard :

PPTP_MODEM_TYPE='bcaa'

Inode xDSL@home

Le déploiement, et la mise en place pour la prise en charge de Inode xDSL@home est décrits dans l'assistance Inode³.

Pour l'instant, il y a peut être encore des problèmes avec le renouvellement du bail avec l'interface dhcp (L'adresse IP de l'interface dhcp qui est attribué automatiquement doit être renouvelée régulièrement) et la coupure de l'accès du circuit avec imonc, cela ne fonctionne pas toujours correctement. Ici toute aide par patch ou autre dispositif est le bien venu, les utilisateurs peuvent aussi donner leurs avis.

Avec xdsl, il y a deux autres options pour pptp :

PPTP_CLIENT_REORDER_TO Le client-pptp, qui utilise xdsl, peut régler les paquets entre la mémoire tampon et son PC. Normalement, le paquet attend 0,3 s avant d'être envoyé au PC. Grâce à cette variable, on peut faire varier le temps de 0,00 (pas de mémoire tampon) à 10.00. Le temps doit toujours être indiqué entre deux postes.

PPTP_CLIENT_LOGLEVEL Dans cette variable on indique, le nombre d'enregistrement possible que produit le client-pptp pour Debug. 0 (peu), 1 (défaut) et 2 (beaucoup).

3. Voir http://www6.inode.at/support/internetzugang/xdsl_home/konfiguration_ethernet_linux.html

1.1.6. OPT_POESTATUS - Moniteur pour l'état du PPPoE sur la console fli4l

Thorsten Pohlmann a développé un moniteur pour l'état du PPPoE avec les connexions DSL.

Si vous paramétrez la variable OPT_POESTATUS='yes' vous pouvez consulter le l'état de la DSL sur le 3ème écran fli4l à tout moment. Vous passez cette écran avec la combinaison de touches ALT-F3, vous pouvez revenir en arrière. 1. Ecran fli4l avec ALT-F1.

A. Annexe du paquetage DSL

A.1. DSL - PPPD et filtre actif

Nous mettons à disposition dans fli4l l'expression suivant :

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

Pour obtenir, en partant du principe, d'envoyer uniquement les paquets du réseau local à Internet, et de garder la connexion ouverte, avec quelques exceptions :

- *TCP-RST* : Réponses aux demandes de connexion qui ont été refusées venant de l'extérieur, et ne pas mettre de Timeout (ou temps d'arrêt) derrière
- *ICMP* : Pour l'envoi de messages ICMP ne pas mettre également de Timeout derrière, à moins que vous envoyez une écho-requête.

Cette expression est généralement réalisée en PPPD, dans le filtrage de paquet par le Kernel. Cela ressemble à l'exemple ci-dessous :

```
#
# 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
```

Table des figures

Liste des tableaux

1.1.	Type de création de paquet-pppoe	8
1.2.	Bande passante et charge CPU pour pppoe	8
1.3.	Cartes-Fritz	9
1.4.	Type de Modem-PPTP	10

Index

FRITZDSL_CHARGEINT, [4](#)
FRITZDSL_DEBUG, [4](#)
FRITZDSL_FILTER, [6](#)
FRITZDSL_HUP_TIMEOUT, [4](#)
FRITZDSL_MRU, [6](#)
FRITZDSL_MTU, [6](#)
FRITZDSL_NAME, [3](#)
FRITZDSL_NF_MSS, [6](#)
FRITZDSL_PASS, [4](#)
FRITZDSL_PROVIDER, [9](#)
FRITZDSL_TIMES, [5](#)
FRITZDSL_TYPE, [9](#)
FRITZDSL_USEPEERDNS, [3](#)
FRITZDSL_USER, [4](#)

OPT_FRITZDSL, [9](#)
OPT_POESTATUS, [11](#)
OPT_PPPOE, [7](#)
OPT_PPPOE_CIRC, [8](#)
OPT_PPTP, [10](#)

PPPOE_CHARGEINT, [4](#)
PPPOE_CIRC_N, [8](#)
PPPOE_CIRC_x_CHARGEINT, [8](#)
PPPOE_CIRC_x_DEBUG, [8](#)
PPPOE_CIRC_x_ETH, [8](#)
PPPOE_CIRC_x_FILTER, [8](#)
PPPOE_CIRC_x_HUP_TIMEOUT, [8](#)
PPPOE_CIRC_x_MRU, [8](#)
PPPOE_CIRC_x_MTU, [8](#)
PPPOE_CIRC_x_NAME, [8](#)
PPPOE_CIRC_x_PASS, [8](#)
PPPOE_CIRC_x_TIMES, [8](#)
PPPOE_CIRC_x_TYPE, [8](#)
PPPOE_CIRC_x_USEPEERDNS, [8](#)
PPPOE_CIRC_x_USER, [8](#)
PPPOE_DEBUG, [4](#)
PPPOE_ETH, [7](#)

PPPOE_FILTER, [6](#)
PPPOE_HUP_TIMEOUT, [4](#), [8](#)
PPPOE_MRU, [6](#)
PPPOE_MTU, [6](#)
PPPOE_NAME, [3](#)
PPPOE_NF_MSS, [6](#)
PPPOE_PASS, [4](#)
PPPOE_TIMES, [5](#)
PPPOE_TYPE, [7](#)
PPPOE_USEPEERDNS, [3](#)
PPPOE_USER, [4](#)
PPTP_CHARGEINT, [4](#)
PPTP_CLIENT_LOGLEVEL, [10](#)
PPTP_CLIENT_REORDER_TO, [10](#)
PPTP_DEBUG, [4](#)
PPTP_ETH, [10](#)
PPTP_FILTER, [6](#)
PPTP_HUP_TIMEOUT, [4](#)
PPTP_MODEM_TYPE, [10](#)
PPTP_NAME, [3](#)
PPTP_PASS, [4](#)
PPTP_TIMES, [5](#)
PPTP_USEPEERDNS, [3](#)
PPTP_USER, [4](#)