

# **Paquetage TOOLS - Outils de débogage**

## **Version 3.10.4**

Frank Meyer  
courriel: [frank@fli4l.de](mailto:frank@fli4l.de)

L'équipe fli4l  
courriel: [team@fli4l.de](mailto:team@fli4l.de)

25 octobre 2015

# Table des matières

<b>1</b>	<b>Documentation du packaging TOOLS</b>	<b>3</b>
1.1	TOOLS - Outils supplémentaires pour le débogage . . . . .	3
1.1.1	Outils pour le réseau . . . . .	3
1.1.2	Outils pour la détection du matériel . . . . .	5
1.1.3	Outils pour fichier . . . . .	7
1.1.4	Outils pour développeur . . . . .	7
	<b>Table des figures</b>	<b>9</b>
	<b>Liste des tableaux</b>	<b>10</b>
	<b>Index</b>	<b>11</b>

# 1 Documentation du paquetage TOOLS

## 1.1 TOOLS - Outils supplémentaires pour le débogage

Le paquetage Tools fournit un certain nombre de programmes Unix, pour l'administration et aussi pour le débogage. D'autres programmes sont intégrés comme wget, par ex. pour intercepter la première page (publicité) de certains fournisseurs d'accès. Si vous indiquez 'yes' le programme choisi sera installé dans le routeur fli4l. Le paramètre par défaut est 'no'. Voici une brève présentation des programmes, sur la façon de les utiliser, s.v.p. utiliser la commande man pour avoir plus d'informations sur les commandes des programmes de la distribution Unix/Linux, ou voir le site : <http://www.linuxmanpages.com>

### 1.1.1 Outils pour le réseau

#### **OPT\_DIG** Couteau suisse pour DNS

La commande dig permet d'effectuer différentes requêtes DNS.

#### **OPT\_FTP** Pour l'utilisation d'un client-FTP

Avec le programme ftp, vous pouvez mettre en place une connexion FTP en utilisant un serveur FTP et transmettre des fichiers entre le serveur FTP et le routeur.

**FTP\_PF\_ENABLE\_ACTIVE** Si vous activez la variable `FTP_PF_ENABLE_ACTIVE='yes'` une règle sera ajoutée dans le filtrage de paquets pour le routeur elle permettra d'initialiser un FTP actif. Si la variable `FTP_PF_ENABLE_ACTIVE='no'` est désactivée, vous pouvez créer cette règle manuellement dans `PF_OUTPUT_%` elle sera ajoutée à la liste. Vous pouvez trouver un exemple dans la [section](#) (Page ??).

Un FTP passif est toujours possible, il n'est pas nécessaire d'utiliser cette variable ni de créer une règle de filtrage de paquets.

#### **OPT\_IFTOP** Pour la surveillance du réseau

Avec la commande iftop, vous avez une liste de toutes les connexions réseaux et il affichera le débit direct de fli4l.

La commande iftop démarre, après avoir installé la commande iftop sur le routeur fli4l.

#### **OPT\_IMONC** Pour la gestion du programme imond par ligne de commande

Ce programme est utilisé pour le contrôle de fli4l par ligne de commande sur le routeur, afin de gérer imond.

#### **OPT\_IPERF** Pour mesurer la performance du réseau

Avec la commande iperf, vous pouvez effectuer des mesures sur la performance du réseau. En outre, la commande doit être lancée sur les deux systèmes serveur/client pour le test. Paramètre de la commande sur le serveur.

```
fli4l-server 3.10.4~# iperf -s
```

```
-----  
Server listening on TCP port 5001
```

TCP window size: 85.3 KByte (default)

-----

Ensuite pour démarrer, le serveur attend une connexion du client. Paramètre de la commande sur le client avec l'adresse-IP du serveur.

```
fli4l-client 3.10.4~# iperf -c 1.2.3.4
```

-----

```
Client connecting to 1.2.3.4, TCP port 5001
TCP window size: 16.0 KByte (default)
```

-----

```
[ 3] local 1.2.3.5 port 50311 connected with 1.2.3.4 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3]  0.0-10.0 sec    985 MBytes  826 Mbits/sec
```

Les mesures de performances démarre immédiatement et affiche les premiers résultats. iperf utilise un certain nombre d'options. Pour plus de détails sur ces options, visiter s'il vous plaît la page d'accueil du lien <http://iperf.sourceforge.net/>.

**OPT\_NETCAT** Pour le transfert de données, basé sur un serveur TCP

**OPT\_NGREP** Grep peut être utilisé directement sur le périphérique réseau.

**OPT\_NTTCP** Pour tester le réseau

Avec la commande NTTCP, on peut tester la vitesse du réseau. Pour ce faire, on démarre d'un côté le serveur et de l'autre côté le client correspondant.

On lance le serveur en tapant `nttcp -i -v`. Puis, le serveur attend une demande de test du client. Maintenant pour tester la vitesse, on entre par exemple sur le client `nttcp -t <Adresse-IP du Serveur>`

Démarrer le serveur `nttcp` comme ceci :

```
fli4l-server 3.10.4~# nttcp -i -v
nttcp-l: nttcp, version 1.47
nttcp-l: running in inetd mode on port 5037 - ignoring options beside -v and -p
```

Test avec le client `nttcp` comme ceci :

```
fli4l-client 3.10.4~# nttcp -t 192.168.77.77
1~~8388608~~~~4.77~~~~0.06~~~~14.0713~~~1118.4811~~~~2048~~~~429.42~~~34133.3
1~~8388608~~~~4.81~~~~0.28~~~~13.9417~~~239.6745~~~~6971~~~1448.21~~~24896.4
```

Vous pouvez voir ci-dessous tous les paramètres `nttcp` :

Usage: `nttcp [local options] host [remote options]`

local/remote options are:

- t transmit data (default for local side)
- r receive data
- l# length of bufs written to network (default 4k)
- m use IP/multicasting for transmit (enforces -t -u)
- n# number of source bufs written to network (default 2048)
- u use UDP instead of TCP
- g#us gap in micro seconds between UDP packets (default 0s)
- d set SO\_DEBUG in sockopt
- D don't buffer TCP writes (sets TCP\_NODELAY socket option)

```
-w#      set the send buffer space to #kilobytes, which is
        dependent on the system - default is 16k
-T      print title line (default no)
-f      give own format of what and how to print
-c      compares each received buffer with expected value
-s      force stream pattern for UDP transmission
-S      give another initialisation for pattern generator
-p#     specify another service port
-i      behave as if started via inetd
-R#     calculate the getpid()/s rate from # getpid() calls
-v      more verbose output
-V      print version number and exit
-?      print this help
-N      remote number (internal use only)
default format is: %9b%8.2rt%8.2ct%12.4rbr%12.4cbr%8c%10.2rcr%10.1ccr
```

#### **OPT\_RTMON** Pour le débogage

Si vous installez cette util, il surveillera les changements du tableau de routage. Utilisation initial : le débogage

**OPT\_SOCAT** Le programme "socat" est une version plus ou moins améliorée du [programme "netcat"](#) (Page 4) avec plus de fonctionnalités. En utilisant "socat" vous pouvez non seulement établir ou accepter différents types de connexions réseau, mais aussi d'envoyer des données ou lire des données avec les Sockets UNIX, les périphériques, FIFO, et ainsi de suite. Au sujet des sources et des destinations particulières, *différent* types de connexions peuvent être utilisés : l'exemple suivant serait un serveur réseau qui écoute sur un port TCP et qui écrit les données reçues dans une mémoire FIFO local ou de lire les données dans la mémoire FIFO, puis de les transmettre via le réseau à un client. Vous pouvez aller sur le site <http://www.dest-unreach.org/socat/doc/socat.html> pour avoir plus d'exemples sur les applications et sur la documentation.

#### **OPT\_TCPDUMP** Pour le débogage réseau

Avec la commande tcpdump on peut observer des détails sur le trafic du réseau, et d'analyser en détail les paquets. Pour en savoir plus, faite une recherche par ex. sur Google ou avec le paramètre »tcpdump man«.

tcpdump <paramètre>

#### **OPT\_WGET** Client http/ftp

Avec la commande wget on peut télécharger des données sur un serveur Web avec un fichier batch de lancement, il travail en arrière-plan. Il est pratique (c'est pour cela qu'il est dans le paquetage-fli4l), on peut télécharger d'une manière simple la page web du fournisseur d'accès Internet et la placer sur son propre serveur web avec un lien. Par exemple sur le site de Freenet, Steffen Peiser a décrit les commandes dans ce mini-HOWTO.

Voir : <http://www.fli4l.de/fr/aide/guide-pratique/debutant/wget-und-freenet/>

### **1.1.2 Outils pour la détection du matériel**

En général, on ne sait jamais exactement le matériel qui est installé dans son propre routeur. Le matériel installé peut nous aider à configurer exactement le pilote de la carte réseau

ou du chipset-USB. Pour nous fournir la liste des périphériques et si possible des pilotes correspondants, nous avons le choix de visualiser ces informations, soit sur la console, juste après le démarrage (recommandé pour une première installation) ou plus facilement, par l'intermédiaire de l'interface-Web de votre ordinateur. Vous pouvez voir ci-dessous un exemple des informations fournies, avec la commande :

```
fli4l 3.10.4 # cat /bootmsg.txt

#
# PCI Devices and drivers
#
Host bridge: Advanced Micro Devices [AMD] CS5536 [Geode companion] Host Bridge (rev 33)
Driver: 'unknown'
Entertainment encryption device: Advanced Micro Devices [AMD] Geode LX AES Security Block
Driver: 'geode_rng'
Ethernet controller: VIA Technologies, Inc. VT6105M [Rhine-III] (rev 96)
Driver: 'via_rhine'
Ethernet controller: VIA Technologies, Inc. VT6105M [Rhine-III] (rev 96)
Driver: 'via_rhine'
Ethernet controller: VIA Technologies, Inc. VT6105M [Rhine-III] (rev 96)
Driver: 'via_rhine'
Ethernet controller: Atheros Communications, Inc. AR5413 802.11abg NIC (rev 01)
Driver: 'unknown'
ISA bridge: Advanced Micro Devices [AMD] CS5536 [Geode companion] ISA (rev 03)
Driver: 'unknown'
IDE interface: Advanced Micro Devices [AMD] CS5536 [Geode companion] IDE (rev 01)
Driver: 'amd74xx'
USB Controller: Advanced Micro Devices [AMD] CS5536 [Geode companion] OHC (rev 02)
Driver: 'ohci_hcd'
USB Controller: Advanced Micro Devices [AMD] CS5536 [Geode companion] EHC (rev 02)
Driver: 'ehci_hcd'
```

Vous pouvez voir que 3 cartes réseaux identiques sont installées, gérées par le pilote 'via\_rhine' et une carte wifi Atheros, gérée par le pilote madwifi (le nom n'est pas encore résolu).

**OPT\_HW\_DETECT** Cette variable s'occupe de vérifier les fichiers installés dans le routeur par rapport aux matériels identifiés. On peut alors voir le résultat sur la console après le boot, si vous mettez la variable `HW_DETECT_AT_BOOTTIME` sur 'yes' vous pouvez voir les informations sur l'Interface Web, bien entendu vous devez placer la variable [OPT\\_HTTPD](#) (Page ??) sur 'yes'. Sur l'interface Web, vous pourrez naturellement voir le contenu du fichier '/bootmsg.txt', si vous avez un accès réseau qui fonctionne.

**HW\_DETECT\_AT\_BOOTTIME** Cette variable lance la détection du matériel lors du boot. La détection fonctionne en tâche de fond (cela prend un peu de temps), le résultat sera visible sur la console, puis sera écrit dans le fichier '/bootmsg.txt'.

**OPT\_LSPCI** Pour lister tous les périphériques PCI

**OPT\_I2CTOOLS** Outils pour accéder au bus I<sup>2</sup>C.

**OPT\_IWLEEPROM** Outil pour accéder à l'EEPROM des cartes WLAN (ou cartes wifi) Intel et Atheros.

Nécessaire par exemple pour reprogrammer le domaine réglementaire de la carte ath9k (see <http://blog.asiantuntijakaveri.fi/2014/08/one-of-my-atheros-ar9280-minipcie-cards.html>).

**OPT\_ATH\_INFO** Outil pour accéder à l'EEPROM Intel et des cartes WLAN Atheros.

Cet outil peut extraire les informations détaillées du matériel utilisé et des cartes wifi Atheros, par exemple ath5k. Ceux-la comprennent, le chipset utilisé ou les données d'étalonnage.

### 1.1.3 Outils pour fichier

**OPT\_E3** Éditeur de texte pour fli4l

Il s'agit d'un éditeur de texte de très petite taille, écrit en assembleur. Vous avez à disposition différents modes d'éditeurs, comme d'autre éditeur plus ("grand"). Pour choisir l'un des mode, il suffit d'utiliser la bonne commande de E3 pour démarrer. On obtient un rapide aperçu des raccourcis clavier avec le paramètre man, si vous lancez E3 sans le paramètre man, vous pouvez appuyer sur Alt+H (sauf dans le mode-VI, en mode-CMD à la place de man il faut saisir ":h"). Notez également que le caractère (^) est représenté par la touche "Ctrl".

Commande	Mode
e3 / e3ws	WordStar, JOE
e3vi	VI, VIM
e3em	Emacs
e3pi	Pico
e3ne	NEdit

**OPT\_MTOOLS** Avec mtools nous mettons à disposition une série de commandes (pour la copie, le formatage, etc.) similaire aux commandes DOS, ces commandes serviront à la gestion des données sur des supports DOS.

Vous trouverez dans le lien ci-dessous la documentation de mtools et les syntaxes des paramètres de commandes de chaque programme :

<http://www.gnu.org/software/mtools/manual/mtools.html>

**OPT\_SHRED** Pour effacer un fichier

Si vous installez *shred* sur le routeur, cette commande effacera définitivement les blocs de données.

**OPT\_YTREE** Gestionnaire de fichier

Si vous installez Ytree sur le routeur, vous aurez un gestionnaire de fichier sur votre routeur fli4l.

### 1.1.4 Outils pour développeur

**OPT\_OPENSSL** Avec l'outil OpenSSL vous pouvez mesurer la vitesse des algorithmes d'encodage et le chiffrement du crypto.

```
openssl speed -evp des -elapsed openssl speed -evp des3 -elapsed openssl speed -evp aes128 -elapsed
```

**OPT\_STRACE** Pour le débogage

Avec la commande strace, vous pouvez surveiller les appel systèmes, pour voir le déroulement des programmes.

```
strace <programme>
```

**OPT\_REAVER** Attaque du code PIN WPS par Brute force sur le Wifi

Cette outil teste tous les codes PIN WPS possibles pour déterminer la vulnérabilité du mot de passe WPA sur votre routeur. Si vous voulez plus de détail pour l'utilisation par ligne de commande de reaver, lire la documentation sur le site <http://code.google.com/p/reaver-wps/>

**OPT\_VALGRIND** Pour le débogage de programme

Si vous installez Valgrind sur le routeur, vous pouvez débusquer les failles dans un programme et mettre en évidence des fuites mémoires.



## Table des figures

## Liste des tableaux

# Index

FTP\_PF\_ENABLE\_ACTIVE, [3](#)

HW\_DETECT\_AT\_BOOTTIME, [6](#)

OPT\_ATH\_INFO, [6](#)

OPT\_DIG, [3](#)

OPT\_E3, [7](#)

OPT\_FTP, [3](#)

OPT\_HW\_DETECT, [6](#)

OPT\_I2CTOOLS, [6](#)

OPT\_IFTOP, [3](#)

OPT\_IMONC, [3](#)

OPT\_IPERF, [3](#)

OPT\_IWLEEPROM, [6](#)

OPT\_LSPCI, [6](#)

OPT\_MTOOLS, [7](#)

OPT\_NETCAT, [4](#)

OPT\_NGREP, [4](#)

OPT\_NTTCP, [4](#)

OPT\_OPENSSL, [7](#)

OPT\_REAVER, [7](#)

OPT\_RTMON, [5](#)

OPT\_SHRED, [7](#)

OPT\_SOCAT, [5](#)

OPT\_STRACE, [7](#)

OPT\_TCPDUMP, [5](#)

OPT\_VALGRIND, [8](#)

OPT\_WGET, [5](#)

OPT\_YTREE, [7](#)