

Paquetage TOOLS - Outils de débogage

Version 4.0.0-stable-x86_64-r60603

Frank Meyer
courriel: frank@fli4l.de

L'équipe fli4l
courriel: team@fli4l.de

7 mars 2022

Table des matières

1	Documentation du packaging TOOLS	3
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	8
1.1.3	Outils pour gérer les fichiers	10
1.1.4	Outils pour les développeurs	10
	Table des figures	12
	Liste des tableaux	13
	Index	14

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_BMON Moniteur de bande passante

Le programme *bmon* est un outil de surveillance et de débogage, il collecte les statistiques liées au réseau et les affiche de manière conviviale. Il propose différentes méthodes de sortie, notamment une interface utilisateur avec des curseurs interactifs et une sortie texte programmable avec l'utilisation de scripts.

OPT_CURL Outil pour le transfert de données

Le programme cURL permet le transfert de données depuis ou vers un serveur il supporte un certain nombre de protocoles. Il comprend entre autres les protocoles FTP, HTTP(S), SCP, SFTP et TFTP.

Le programme fournit également une authentification de l'utilisateur, le transfert de données via un proxy, le téléchargement FTP, des requêtes POST pour HTTP, une connexion SSL, les cookies, la reprise du transfert de fichiers interrompus, et encore plus.

Important: *Pour établir une connexion TLS, les certificats X.509 doivent être installés à la racine de Mozilla, pour cela vous devez activer la variable CERT_X509_MOZILLA='yes' dans le paquetage CERT.*

OPT_DIG Couteau suisse pour DNS

Le programme dig vous permet d'effectuer différentes requêtes DNS.

OPT_FTP Pour utiliser 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 cette 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 le programme iftop, vous pouvez créer une liste de toutes les connexions réseaux, il affichera le débit direct de fli4l.

Le programme iftop démarre, après avoir installé 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 le programme 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. Voici la commande du programme sur le serveur.

```
fli4l-server 4.0.0-stable-x86_64-r60603~# 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. Voici la commande sur le client avec l'adresse IP du serveur.

```
fli4l-client 4.0.0-stable-x86_64-r60603~# 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_LNSTAT Information sur les réseaux avancés

Si vous instalez l'OPT lnstat, avec cet outil vous pouvez éditer les informations du réseau dans le dossier `/proc/net/stat`. Le nom signifie "Linux Network Statistics" (ou Statistique du Réseau Linux). La commande lnstat permet un affichage de manière conviviale des informations du réseau.

Exemple du manuel des commandes de lnstat :

```
# lnstat -d  
    Get a list of supported statistics files.  
# lnstat -k arp_cache:entries,rt_cache:in_hit,arp_cache:destroys  
    Select the specified files and keys.  
# lnstat -i 10  
    Use an interval of 10 seconds.  
# lnstat -f ip_contrack  
    Use only the specified file for statistics.
```

```
# lostat -s 0
    Do not print a header at all.
# lostat -s 20
    Print a header at start and every 20 lines.
# lostat -c -1 -i 1 -f rt_cache -k entries,in_hit,in_slow_tot
    Display statistics for keys entries, in_hit and in_slow_tot of field
    rt_cache every second.
```

OPT_NETCAT Pour le transfert de données, basé sur un serveur TCP

OPT_NETIO Mesure les performances du réseau

Le programme "netio" est similaire à "iperf" il mesure la performance du réseau. Vous devez démarrer le programme sur les deux système à tester. Sur le serveur la commande sera **netio -s -t** (pour l'échange de données via le TCP) ou **netio -s -u** (pour l'échange de données via l'UDP). Sur le client la commande sera **netio <serveur> -t** ou **netio <serveur> -u** le processus client démarre, il contacte ensuite le serveur et effectue les mesures.

OPT_NGREP Grep peut être utilisé directement sur le périphérique réseau

OPT_NMAP Scanner de ports

Vous pouvez utiliser Nmap pour scanner les ports ouverts sur un système d'exploitation. Le programme fournit aussi des informations supplémentaires, par ex. les adresses MAC ou la version du système d'exploitation utilisé.

OPT_NTTCP Pour tester le réseau

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

On lance le serveur avec la commande **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 la commande **nttcp -t <Adresse IP du Serveur>**

Démarrer le serveur avec nttcp comme ceci :

```
fli4l-server 4.0.0-stable-x86_64-r60603~# 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 le client avec nttcp comme ceci :

```
fli4l-client 4.0.0-stable-x86_64-r60603~# 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 outil, il surveillera les changements du tableau de routage. L'utilisation initial est : le débogage

OPT_SOCAT Le programme "socat" est une version plus ou moins améliorée du [programme "netcat"](#) (Page 5) 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 utilisées : 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 le programme tcpdump on peut observer en détail le trafic du réseau et d'analyser les paquets. Pour en savoir plus, faite une recherche par ex. sur Google ou avec la commande «tcpdump man».

```
tcpdump <paramètre>
```

OPT_TRACEPATH Détermine le PMTU

Le programme `tracpath` peut soi-disant déterminer le MTU. C'est la taille maximale du paquet que le routeur flie utilise pour le transfère de l'hôte vers le destinataire. Les plus grands paquets seront fragmentés pour (l'IPv4) et rejetés pour (l'IPv6). En règle générale, le Kernel Linux détermine le Path MTU correctement (mots-clés "Path MTU Discovery"). Cependant, il est parfois utile de savoir utiliser le Path MTU pour trouver les problèmes dans le réseau.

Un exemple pour l'IPv4 :

```
sandbox 4.0.0-r46077M # tracepath -4 fli4l.de
1?: [LOCALHOST] pmtu 1500
1: fritz.box 0.703ms
1: fritz.box 0.588ms
2: a89-182-53-190.net-http.de 0.702ms pmtu 1492
2: a81-14-248-243.net-http.de 33.692ms
3: a81-14-249-82.net-http.de 32.089ms asymm 4
4: xe-4-1-2.edge4.Berlin1.Level3.net 35.936ms
5: SYSELEVEN-G.edge4.Berlin1.Level3.net 74.944ms asymm 8
6: ecix.dus.octalus.in-berlin.de 49.693ms asymm 7
7: virtualhost.in-berlin.de 50.269ms reached
Resume: pmtu 1492 hops 7 back 57
```

Ci-dessus, une connexion Internet DSL via une Fritz !Box de AVM et le B-RAS du fournisseur d'accès Internet http. En Allemagne une connexion DSL utilise le protocole (PPPoE) via PPP sur Ethernet, le path MTU est de 1492 octets.

Un exemple pour l'IPv6 :

```
sandbox 4.0.0-r46077M # tracepath -6 fli4l.de
1?: [LOCALHOST] 0.046ms pmtu 1280
1: gw-1362.ham-01.de.sixxs.net 43.586ms
1: gw-1362.ham-01.de.sixxs.net 42.832ms
2: 2001:6f8:862:1::c2e9:c729 43.565ms asymm 1
3: 2001:6f8:862:1::c2e9:c72c 44.313ms asymm 2
4: 30gigabitethernet4-3.core1.fra1.he.net 64.501ms asymm 6
5: no reply
6: virtualhost.in-berlin.de 65.949ms reached
Resume: pmtu 1280 hops 6 back 56
```

Ci-dessus, une connexion Internet via un tunnel 6in4 de SixXS. Dans la configuration du tunnel le MTU est fixé à 1280 octets. C'est le plus petit MTU possible pour une connexion IPv6, il ne sera plus réduit nulle part ailleurs entre l'hôte et le destinataire.

OPT_DHCPDUMP Pour analyser les paquets DHCP

Avec le programme `dhcpcdump` on peut analysé les paquets DHCP en détail. Le programme est basé sur le programme `tcpdump`, la sortie générée des paquets est plus facilement lisible.

Utilisation :

```
dhcpcdump -i interface [-h expression régulière]
```

Vous démarrez le programme par exemple avec la commande suivante :

```
dhcpcdump -i eth0
```

Si vous le souhaitez, vous pouvez également filtrer directement une adresse MAC spécifique, en utilisant une expression régulière. La commande ressemble à ceci :

```
dhcpcdump -i eth0 -h ^00:a1:c4
```

La réponse pourrait alors, par exemple ressembler à ceci :

```

TIME: 15:45:02.084272
  IP: 0.0.0.0.68 (0:c0:4f:82:ac:7f) > 255.255.255.255.67 (ff:ff:ff:ff:ff:ff)
  OP: 1 (BOOTPREQUEST)
HTYPE: 1 (Ethernet)
HLEN: 6
HOPS: 0
  XID: 28f61b03
SECS: 0
FLAGS: 0
CIADDR: 0.0.0.0
YIADDR: 0.0.0.0
SIADDR: 0.0.0.0
GIADDR: 0.0.0.0
CHADDR: 00:c0:4f:82:ac:7f:00:00:00:00:00:00:00:00:00:00
  SNAME: .
  FNAME: .
OPTION: 53 ( 1) DHCP message type          3 (DHCPREQUEST)
OPTION: 54 ( 4) Server identifier          130.139.64.101
OPTION: 50 ( 4) Request IP address         130.139.64.143
OPTION: 55 ( 7) Parameter Request List    1 (Subnet mask)
                                           3 (Routers)
                                           58 (T1)
                                           59 (T2)

```

OPT_WGET Client http/ftp

Avec le programme `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 là placer sur sont propre serveur web avec un lien. Par exemple sur le site de Freenet, Steffen Peiser a décrit les commande dans ce mini HOWTO.

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

Important: *Pour établir une connexion TLS, les certificats X.509 doivent être installés à la racines de Mozilla, pour cela vous devez activer la variable `CERT_X509_MOZILLA='yes'` dans le paquetage `CERT`.*

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 4.0.0-stable-x86_64-r60603 # 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 Ce script 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 I²C

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 (voir <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 pour les cartes wifi Atheros, par exemple ath5k. Ceux-la comprennent, le chipset utilisé ou les données d'étalonnage.

OPT_FLASHROM Outil pour flasher le chipset

Grâce à cet outil, vous pouvez installer un nouveau BIOS ou un nouveau firmware par exemple, sur votre carte PC Engines. Pour plus de détails, vous pouvez aller sur le site <http://www.flashrom.org>.

1.1.3 Outils pour gérer les fichiers

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, dans le 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, ce programme 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 les développeurs

OPT_OPENSSL Avec l'outil OpenSSL vous pouvez mesurer la vitesse de chiffrement d'encodage et l'algorithme cryptographique.

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

OPT_STRACE Pour le débogage

Avec le programme strace, vous pouvez surveiller les appel systèmes, pour voir le déroulement d'un programme.

```
strace <programme>
```

OPT_REAVER Attaque du code PIN WPS par brute force sur le wifi

Cette outil teste tous les codes PIN WPS 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 d'un programme et mettre en évidence les fuites mémoires.

Table des figures

Liste des tableaux

Index

FTP_PF_ENABLE_ACTIVE, [3](#)

HW_DETECT_AT_BOOTTIME, [9](#)

OPT_ATH_INFO, [9](#)

OPT_BMON, [3](#)

OPT_CURL, [3](#)

OPT_DHCPDUMP, [7](#)

OPT_DIG, [3](#)

OPT_E3, [10](#)

OPT_FLASHROM, [9](#)

OPT_FTP, [3](#)

OPT_HW_DETECT, [9](#)

OPT_I2CTOOLS, [9](#)

OPT_IFTOP, [4](#)

OPT_IMONC, [4](#)

OPT_IPERF, [4](#)

OPT_IWLEEPROM, [9](#)

OPT_LNSTAT, [4](#)

OPT_LSPCI, [9](#)

OPT_MTOOLS, [10](#)

OPT_NETCAT, [5](#)

OPT_NETIO, [5](#)

OPT_NGREP, [5](#)

OPT_NMAP, [5](#)

OPT_NTTCP, [5](#)

OPT_OPENSSL, [10](#)

OPT_REAVER, [10](#)

OPT_RTMON, [6](#)

OPT_SHRED, [10](#)

OPT_SOCAT, [6](#)

OPT_STRACE, [10](#)

OPT_TCPDUMP, [6](#)

OPT_TRACEPATH, [6](#)

OPT_VALGRIND, [10](#)

OPT_WGET, [8](#)

OPT_YTREE, [10](#)