

Paket UMTS

Version 3.10.4

Gerd Walter
E-Mail: fli4l@hgwb.de

Das fli4l-Team
E-Mail: team@fli4l.de

25. Oktober 2015

Inhaltsverzeichnis

1. Dokumentation des Paketes UMTS	3
1.1. UMTS - Anbindung mittels UMTS an das Internet	3
1.1.1. Konfiguration	3
1.1.2. Beispielkonfiguration für RRDTOOL	6
A. Anhang zum Paket UMTS	7
A.1. UMTS	7
A.1.1. Unterstützte Hardware	7
A.1.2. Modemschnittstelle nicht aktiviert	7
Abbildungsverzeichnis	9
Tabellenverzeichnis	10
Index	11

1. Dokumentation des Paketes UMTS

1.1. UMTS - Anbindung mittels UMTS an das Internet

Anbindung eines fli4l mittels UMTS an das Internet. Für den Betrieb sind unter anderem auch weitere optionale Pakete erforderlich.

1.1.1. Konfiguration

OPT_UMTS Standard-Einstellung: `OPT_UMTS='no'`

'yes' aktiviert das Paket.

UMTS_DEBUG Standard-Einstellung: `UMTS_DEBUG='no'`

Soll pppd zusätzliche Debug-Informationen ausgeben, muss man `UMTS_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.

UMTS_PIN Standard-Einstellung: `UMTS_PIN='disabled'`

Pin für die SIM-Karte

Erlaubt sind eine 4stellige Nummer oder das Wort 'disabled'

UMTS_DIALOUT Standard-Einstellung: `UMTS_DIALOUT='*99***1#'`

Wählparameter zum Herstellen der Verbindung

UMTS_GPRS_UMTS Standard-Einstellung: `UMTS_GPRS_UMTS='both'`

Welche Übertragungsart soll genutzt werden

Erlaubte Werte (both, gprs, umts)

UMTS_APN Standard-Einstellung: `UMTS_APN='web.vodafone.de'`

UMTS_USER Standard-Einstellung: `UMTS_USER='anonymer'`

UMTS_PASSWD Standard-Einstellung: `UMTS_PASSWD='surfer'`

Hier werden die für die Einwahl nötigen Daten angegeben.

Es sind Benutzerkennung und Passwort für den jeweils benutzten Provider anzugeben.

`UMTS_USER` enthält die Benutzerkennung, `UMTS_PASSWD` das Passwort.

Für einige deutsche Netzbetreiber/Provider lauten die APNs (Einwahlknoten)

- <http://www.teltarif.de/mobilfunk/internet/einrichtung.html>

Einwahldaten einiger deutscher Netzbetreiber/Provider

Anbieter	APN	Benutzername	Passwort
T-Mobile	internet.t-mobile	beliebig	beliebig
Vodafone	web.vodafone.de	beliebig	beliebig
E-Plus	internet.eplus.de	eplus	gprs
O2 (Vertragskunden)	internet	beliebig	beliebig
O2 (Prepaid-Kunden)	pinternet.interkom.de	beliebig	beliebig
Alice	internet.partner1	beliebig	beliebig

UMTS_NAME Standard-Einstellung: UMTS_NAME='UMTS'

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.

UMTS_HUP_TIMEOUT Standard-Einstellung: UMTS_TIMEOUT='600'

Hier kann die Zeit in Sekunden angegeben werden, nach welcher die Verbindung beendet werden soll, wenn nichts mehr über die UMTS-Verbindung läuft. Dabei steht ein Timeout von '0' für kein Timeout.

UMTS_TIMES Standard-Einstellung: UMTS_TIMES='Mo-Su:00-24:0.0:Y'

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.

UMTS_CHARGEINT Standard-Einstellung: UMTS_CHARGEINT='60'

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

UMTS_USEPEERDNS Standard-Einstellung: UMTS_USEPEERDNS='yes'

Soll der DNS des Providers verwendet werden.

UMTS_FILTER Standard-Einstellung: UMTS_FILTER='yes'

fli4l legt automatisch auf, wenn während der über hangup timeout angegebenen Zeit keine Daten über das ppp0-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 UMTS-Verbindung nie beendet.

Hier hilft die Option UMTS_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.

UMTS_ADAPTER (optional)

Hier wird eingetragen ob es sich um eine PCMCIA-Karte, einen USB-Adapter oder um ein per USB-Kabel angeschlossenes Telefon handelt.

Bei nichtvorhandensein der Variable werden nur die benötigten Dateien für einen USB-Adapter kopiert.

Erlaubte Werte: (pcmcia,usbstick,usbphone)

Alle folgenden Variablen sind optional und nur notwendig wenn die automatische Erkennung versagt.

UMTS_IDVENDOR (optional) UMTS_IDVENDOR='xxxx'

Hersteller ID nach Einschalten des Adapters

UMTS_IDDEVICE (optional) UMTS_IDDEVICE='xxxx'

Produkt ID nach Einschalten des Adapters

Angabe der folgenden beiden Parameter nur notwendig, wenn sich eine ID ändert nach der Initialisierung

UMTS_IDVENDOR2 (optional) UMTS_IDVENDOR2='xxxx'

Hersteller ID nach Initialisierung des Adapters

UMTS_IDDEVICE2 (optional) UMTS_IDDEVICE2='xxxx'

Produkt ID nach Initialisierung des Adapters

UMTS_DRV (optional) UMTS_DRV='xxxx'

Treiber zum Ansteuern Adapters, wenn nicht angegeben wird 'usbserial' genommen

UMTS_SWITCH (optional) UMTS_SWITCH='-v 0x0af0 -p 0x6971 -M 555...000 -s 10'

Parameter für usb-modeswitch zum Initialisieren des Modems. (siehe Website usb-modeswitch)
Es sollten bis auf wenige Ausnahmen alle auf der Website genannten Modems automatisch erkannt werden.

- http://www.draisberghof.de/usb_modeswitch/

UMTS_DEV (optional)

Bei Problemen kann hier die Datenschnittstelle für den pppd angegeben werden. Für die Adapter sind das meist folgende:

ttyUSB0 für usbstick
ttyS2 für pcmcia
ttyACM0 für usbphone

UMTS_CTRL (optional)

Einige Adapter haben mehrere Schnittstellen, über die das Modem gesteuert wird. Ist nur eine vorhanden können Statusinformationen nur im 'Offline'-Zustand ausgelesen werden. Bei einer Option Fusion UMTS Quad lautet die Schnittstelle zB: ttyUSB2.

1.1.2. Beispielkonfiguration für RRDTOOL

Je nach Hardware ist es möglich, über [OPT_HTTPD](#) (Seite ??) die Signalstärke und Bitfehler anzeigen zu lassen. Ausserdem kann der Verlauf der Signalstärke bzw. Bitfehlerrate mittels `OPT_RRDTOOL` aufgezeichnet werden. Bei mancher Hardware ist es nicht so sinnvoll, da Statusinformationen nur während des Offline-Zustandes ausgelesen werden können. Als Quelle für RRD ist dabei 'umts' anzugeben.

Beispielkonfig für rrdtool:

```
RRDTOOL_x_SOURCE='umts'  
RRDTOOL_x_LABEL='UMTS Status'  
RRDTOOL_x_OPTIONS_N='2'  
RRDTOOL_x_OPTIONS_1='signal'  
RRDTOOL_x_OPTIONS_1_LABEL='Signalstärke'  
RRDTOOL_x_OPTIONS_2='error'  
RRDTOOL_x_OPTIONS_2_LABEL='Bitfehler'
```

A. Anhang zum Paket UMTS

A.1. UMTS

A.1.1. Unterstützte Hardware

Dieses Paket unterstützt folgende UMTS-Hardware:

Für den Betrieb sind unter anderem auch weitere Pakete erforderlich.

Für USB-Adapter muß das USB-Paket aktiviert werden.

OPT_USB='yes'

Hardware:	getestet	zusätzliche Pakete
Novatel Adapter:		
Merlin U530	ja	PCMCIA, TOOLS (serial)
Merlin U630	nein	PCMCIA, TOOLS (serial)
MC950D	ja	USB
OPTION Adapter:		
3G Datacard	nein	PCMCIA, USB
GT 3G Quad	ja	PCMCIA, USB
GT Fusion	nein	PCMCIA, USB
GT MAX HSUPA GX0301	ja	PCMCIA, USB
bei den vier Cardbusadaptern ist PCMCIA_PCIC='yenta_socket' zu setzen		
Icon 225 (GI0225)	ja	USB
Huawei Adapter:		
E220, E230, E270	ja	USB
E510	ja	USB
E800	ja	USB
K3520	ja	USB
ZTE Adapter:		
MF110	ja	USB
MF190	ja	USB

A.1.2. Modemschnittstelle nicht aktiviert

Bei einigen OPTION UMTS Sticks kann es vorkommen, das die Modemschnittstelle nicht aktiviert ist, welche aber für den pppd benötigt wird.

Beispiel anhand des GIO225 Adapter

Kontrolle mittels:

```
grep "" /sys/bus/usb/devices/*/tty*/hsotype
```

Die Ausgabe sollte etwa so aussehen:

```
/sys/bus/usb/devices/2-1:1.0/tty/ttyHS0/hsotype:Control
/sys/bus/usb/devices/2-1:1.0/tty/ttyHS1/hsotype:Application
/sys/bus/usb/devices/2-1:1.1/tty/ttyHS2/hsotype:Diagnostic
```

Hier fehlt die Ausgabe hsotype:Modem.

Jetzt kann man mit folgenden Befehl die Interface Konfiguration kontrollieren:

```
chat -e -t 1 '' "AT_OIFC?" OK >/dev/ttyHS0 </dev/ttyHS0
```

Die Ausgabe sollte folgendermassen aussehen:

```
AT_OIFC?
_OIFC: 3,1,1,0
```

OK

Sollte dort folgendes stehen:

```
AT_OIFC?
_OIFC: 2,1,1,0
```

OK

kann man die Modemschnittstelle mit folgenden Befehl aktivieren:

```
chat -e -t 1 '' "AT_OIFC=3,1,1,0" OK >/dev/ttyHS0 </dev/ttyHS0
```

Danach den Adapter noch einmal abziehen und neu anstecken.

Jetzt sollte mittels:

```
grep "" /sys/bus/usb/devices/*/tty*/hsotype
```

auch ein Modemeintrag vorhanden sein.

```
/sys/bus/usb/devices/2-1:1.0/tty/ttyHS0/hsotype:Control
/sys/bus/usb/devices/2-1:1.0/tty/ttyHS1/hsotype:Application
/sys/bus/usb/devices/2-1:1.1/tty/ttyHS2/hsotype:Diagnostic
/sys/bus/usb/devices/2-1:1.2/tty/ttyHS3/hsotype:Modem
```


Abbildungsverzeichnis

Tabellenverzeichnis

Index

OPT_UMTS, [3](#)

UMTS_ADAPTER, [4](#)

UMTS_APN, [3](#)

UMTS_CHARGEINT, [4](#)

UMTS_CTRL, [5](#)

UMTS_DEBUG, [3](#)

UMTS_DEV, [5](#)

UMTS_DIALOUT, [3](#)

UMTS_DRV, [5](#)

UMTS_FILTER, [4](#)

UMTS_GPRS_UMTS, [3](#)

UMTS_HUP_TIMEOUT, [4](#)

UMTS_IDDEVICE, [5](#)

UMTS_IDDEVICE2, [5](#)

UMTS_IDVENDOR, [5](#)

UMTS_IDVENDOR2, [5](#)

UMTS_NAME, [3](#)

UMTS_PASSWD, [3](#)

UMTS_PIN, [3](#)

UMTS_SWITCH, [5](#)

UMTS_TIMES, [4](#)

UMTS_USEPEERDNS, [4](#)

UMTS_USER, [3](#)