

# Mit UMTS-Handy ins Internet

## Mit einem UMTS-Handy ins Internet

Wie wird es unter Linux eingerichtet und welche Erfahrungen wurden gemacht

Dipl.Ing. Christoph Stockmayer, Schwaig, [sto@stockmayer.de](mailto:sto@stockmayer.de)

### 1. Einleitung

In diesem Artikel wird beschrieben, wie man unter Linux mit einem UMTS-Handy ins Internet kommt. Außerdem werden Erfahrungen damit insbesondere mit dem Provider Mobilcom geschildert. Die Installation beruhte auf folgenden Versionen/Geräten, sollte aber ansonsten auf einer anderen Konstellation ähnlich funktionieren:

Linux: SuSE10.0

UMTS-Handy: Nokia N70 mit USB-Anschluss

Provider: Mobilcom (mit Vodafone)

### 2. Konfiguration

Das USB-Kabel wird am Linux-System als `/dev/ttyACM0` erkannt. Es handelt sich um eine normale serielle Schnittstelle, deren Eigenschaften mit `stty -a < /dev/ttyACM0` angeschaut (und auch gesetzt werden können). Im `yast` wird dies bei der Modemkonfiguration als „Nokia Mobile Phones Nokia N70“ erkannt und kann konfiguriert werden:

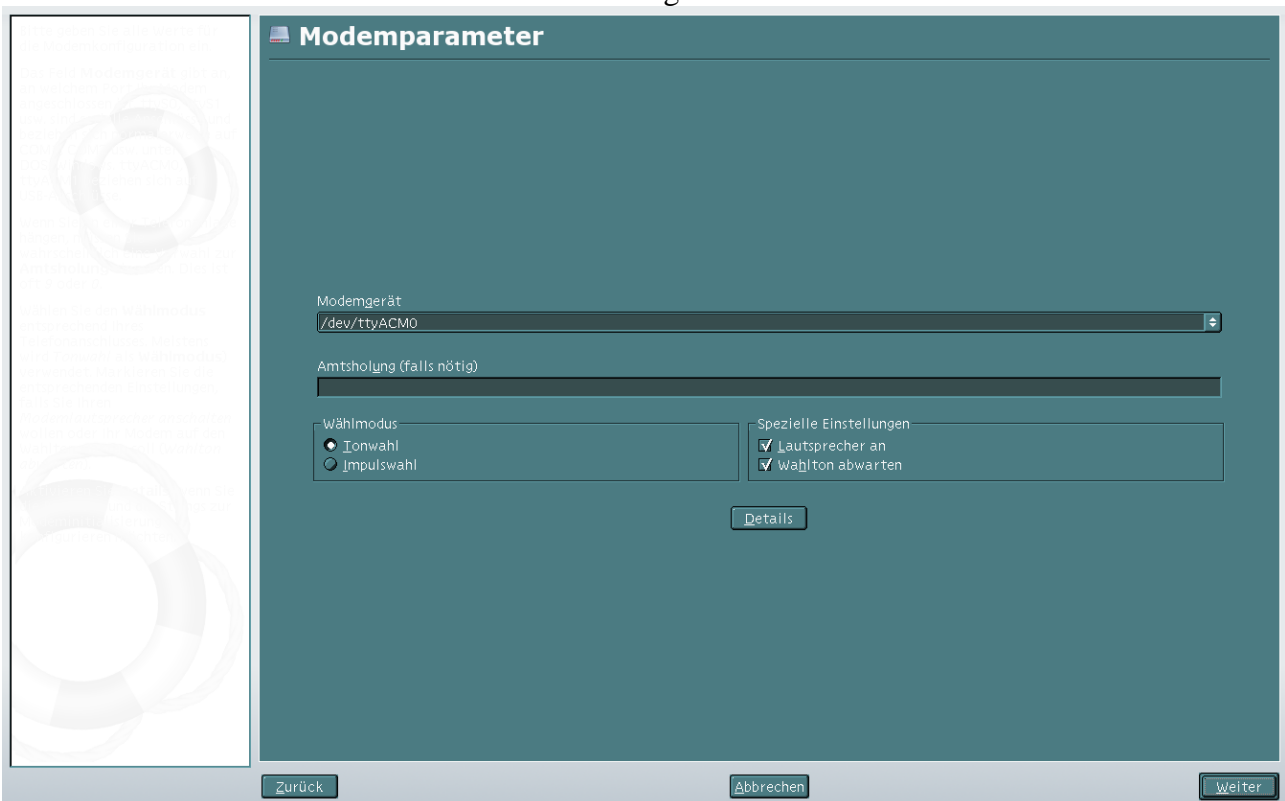


Abbildung 1: Eingabe der Schnittstelle

## Mit UMTS-Handy ins Internet

Unter Details finden sich die Modem-AT-Parameter, die einzustellen sind: ATZ setzt das Modem zurück, AT&S7=4&S0=0L1V1X4E1Q0 setzt spezielle Modemparameter und mit dem neuen UMTS-Parameter AT+CGDCONT=1, "IP", "web.vodafone.de" wird der Provider gesetzt.

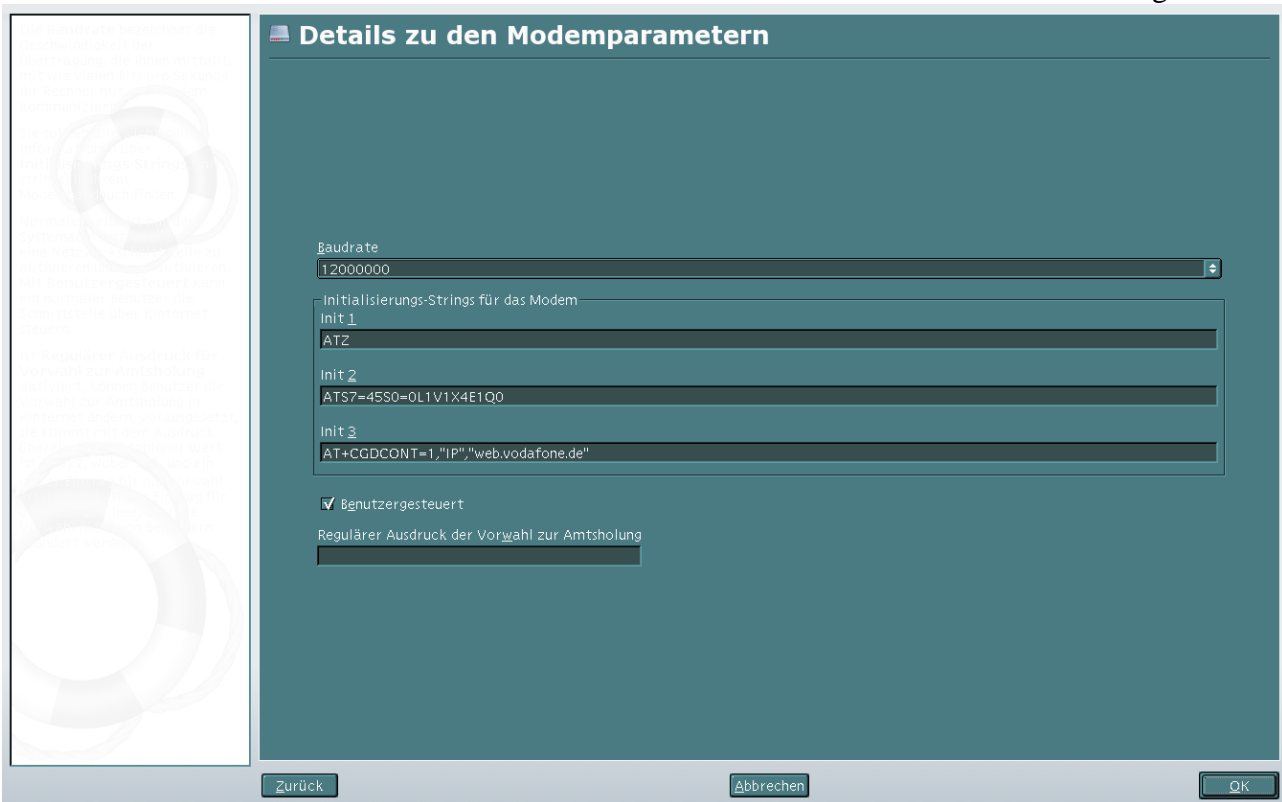


Abbildung 2: Setzen der Modemparameter

Beim Weitergehen muß die Einwahl konfiguriert werden. Dazu können momentan keine vordefinierten Provider verwendet werden – dazu ist UMTS zu neu. Aber man kann ja einen eigenen Provider einrichten: Die Telefonnummer lautet \*99\*\*\*1#, als Benutzer trägt man `vodafone` ein, das Passwort bleibt frei.

## Mit UMTS-Handy ins Internet

Provider-Parameter

Name für die Einwahl:

Name des Providers:

Telefonnummer:

Berechtigung:

Benutzername:

Passwort:

Immer Passwort abfragen

Abbildung 3: Telefonnummer

Ansonsten reichen die Voreinstellungen.

Zur Einwahl muß das Handy eingeschaltet sein (inklusive Eingabe der PIN). Nun kann z.B. im Programm `kinternet` dieses Modem mit seinem neuen Provider ausgewählt und die Internetverbindung aufgebaut werden. Im Kommando `route -n` (als Sysadmin root) sieht das dann so aus:

```
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
10.6.6.6         0.0.0.0        255.255.255.255 UH    0      0      0 modem0
193.97.237.0    0.0.0.0        255.255.255.0  U     0      0      0 eth0
127.0.0.0       0.0.0.0        255.0.0.0      U     0      0      0 lo
0.0.0.0         10.6.6.6       0.0.0.0        UG    0      0      0 modem0
```

Abbildung 4: Routing

Der Provider weist der Schnittstelle eine IP-Adresse zu und gibt Default-Route (Gateway) und Nameserver bekannt. Somit steht die Verbindung.

### 3. Start der Verbindung „von Hand“

Möchte man die Verbindung „von Hand“ oder aus einem Shell-Skript heraus starten, können folgende Befehlszeilen helfen, in denen eine ev. vorhandene Defaultroute gelöscht wird und der `pppd`-Dienst mit den entsprechenden Parametern gestartet wird:

```
#!/bin/sh
# ppp starten
TTY=/dev/ttyACM0
```

## Mit UMTS-Handy ins Internet

```
SPEED=460800
route del default
pppd $TTY $SPEED file /etc/ppp/options.handyumts
```

Abbildung 5: Einwahlskript

Die Optionen stehen dabei in `/etc/ppp/options.handyumts`:

```
460800
debug
kdebug 4
connect /etc/ppp/handyumts-chat
crtscts
local
nomagic
nodetach
novj
novjccomp
noauth
replacedefaultroute
defaultroute
ipcp-accept-local
usepeerdns
```

Abbildung 6: Datei `/etc/ppp/options.handyumts`

Und die Einwahlparameter in `/etc/ppp/handyumts-chat`:

```
exec chat
TIMEOUT 5 \
ECHO ON \
ABORT '\nBusy\r' \
ABORT '\nError\r' \
ABORT '\nNo answer\r' \
ABORT '\nNo carrier\r' \
ABORT '\nNo dialtone\r' \
ABORT '\nRinging\r\n\r\nRinging\r' \
'' \rAT \
TIMEOUT 12 \
SAY "Press CTRL-c to close the connection at any stage!" \
SAY "\ndefining PPP context...\n" \
OK ATZ \
OK ATS7=45 \
OK ATS0=0 \
OK ATL1 \
OK ATV1 \
OK ATX4 \
OK ATE1 \
OK ATQ0 \
OK AT+CSQ \
OK AT+CGDCONT=1, "IP", "web.vodafone.de" \
OK ATD*99***1# \
TIMEOUT 22 \
SAY "\nWaiting for connect...\n" \
CONNECT "" \
SAY "\nConnected." \
SAY "\nIf fail, try restart\n"
```

Abbildung 7: Datei `/etc/ppp/handyumts-chat`

## 4. Erfahrungen

## Mit UMTS-Handy ins Internet

Wenn man die Parameter kennt, geht es erstaunlich einfach, trotz der Aussagen von Mobilcom und Vodafone, daß es mit Linux nicht funktioniert! Leider war bei mir die Euphorie zu groß, um über finanzielle Konsequenzen nachzudenken: Weder beim Kauf des Handys noch im Gespräch mit den Call-Center-Mitarbeitern (beides Mobilcom) tauchte die Frage nach einem Tarif und einem Vertrag dafür auf. Zu diesem Zeitpunkt existierte nur der alte Telefonvertrag. Die Meinung, der Tarif einer UMTS-Karte sei analog anzuwenden, ist irrig. Das böse Erwachen kam mit der nächsten Handy-Rechnung, wo Mobilcom für ca. 120MByte Datenübertragung sage und schreibe netto EUR 1880.- verlangte (als Rechnungsposten „51 Datenverbindungen“). Nach 6wöchigem Nachbohren ohne Resonanz bemühte man sich bei Mobilcom mitzuteilen, es handle sich (da ohne Vertrag) um einen „surf-by-call-Tarif“, der mit EUR 0.16 pro 10kByte (+ einige Cent pro angefangene Stunde und Verbindung) zu Buche schlage. Ausführliche Recherchen auf den Tarifseiten von Mobilcom führten zu keinem solchen Tarif. Im Vergleich dazu kostet der „surf-option-volume(200)-Tarif“ 17.- EUR im Monat für 200MByte Datentransfer, das wäre ein Faktor von über 100 weniger! Nachfragen im Call Center führten zu sehr schroffen Aussagen wie: „Da gibt es keine Kulanz“ oder „das hat alles seine Richtigkeit“ oder „keine Ahnung, wie solch ein Tarif zustande kommt“ oder „Anfrage abgelehnt“. Ich bin gespannt, ob sich da noch was bewegt, die Korrespondenz mit „meinem Kundenberater“ (wer immer das ist), ist enorm mühsam ...

Also ganz wichtig: **Erst Vertrag abschließen bzw. sich nach dem Tarif erkundigen, bevor der Surf-Spaß beginnt!**

zur Person:

Christoph Stockmayer ist seit 25 Jahren freiberuflicher Trainer in den Gebieten Programmierung, C/C++/Java/Perl, OOA/OOD und im gesamten UNIX/Linux-Sektor. Er ist SuSE Certified Linux-Trainer und Lehrbeauftragter an der FH Nürnberg. Er betreut außerdem Programmier- und UNIX/Linux-Projekte.

