

Re: Actius MM10 modem

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.portable/2005-06/0069.html>

From: Jim (james_at_the-computer-shop.co.uk)

Date: 06/20/05

Date: Mon, 20 Jun 2005 08:07:44 GMT

jan.a@gmx.de wrote:

- > *I am trying to make the modem of the Sharp Actius MM10 notebook work*
- > *with Redhat Linux.*
- > *The modem driver seems to work, since up to CONNECT everything seems to*
- > *work:*
- > *ATZ*
- > *OK*
- > *ATMIL1*
- > *OK*
- > *ATDT1000*
- > *CONNECT 34666 V42bis*
- >
- > *Then however Kppp fails, telling me that "pppd exited with return value*
- > *16".*
- > *Did anyone succeed to get the modem working on this laptop, or any*
- > *ideas what is going wrong?*
- > *Some log messages are attached below.*
- > *Thanks in advance!!*
- >
- > ---
- >
- > *Jun 17 16:26:12 localhost pppd[4627]: pppd 2.4.2 started by root, uid 0*
- > *Jun 17 16:26:12 localhost pppd[4627]: Using interface ppp0*
- > *Jun 17 16:26:12 localhost pppd[4627]: Connect: ppp0 <--> /dev/ttyS1*
- > *Jun 17 16:26:30 localhost pppd[4627]: PAP authentication succeeded*
- > *Jun 17 16:26:30 localhost pppd[4627]: kernel does not support PPP*
- > *filtering*
- > *Jun 17 16:26:39 localhost pppd[4627]: Terminating on signal 15.*
- > *Jun 17 16:26:45 localhost pppd[4627]: Connection terminated.*
- > *Jun 17 16:26:45 localhost pppd[4627]: Exit.*
- >
- > ---
- >
- > *Opener: received SetSecret*
- > *Opener: received SetSecret*
- > *Opener: received OpenLock*
- >
- > *Opener: received OpenDevice*

```
> Opener: received ExecPPPDaemon
> In parent: pppd pid 2480
> Couldn't find interface ppp0: No such device
> Kernel supports ppp alright.
> Couldn't find interface ppp0: No such device
> Opener: received KillPPPDaemon
> In killpppd(): Sending SIGTERM to 2480
> It was pppd that died
> pppd exited with return value 16
> Sending 2027 a SIGUSR1
> Opener: received RemoveSecret
> Opener: received RemoveSecret
> Opener: received OpenResolv
> Opener: received OpenResolv
> Opener: received RemoveLock
> ALSA lib pcm_hw.c:494:(snd_pcm_hw_start) SNDRV_PCM_IOCTL_START failed:
> Broken pipe
>
```

Not knowing much about the German telecommunications system, I'll throw a few ideas at you.

Did you disable call waiting?

Did you disable originator-mask?

Is your modem configured for the German telecommunications system? (for some reason, most dialers/modem controllers default to the US system – in most countries, using out of bounds dialling mechanisms is illegal!)

=====

From the HP page on pppd:

pppd is a daemon process used in UNIX systems to manage connections to other hosts using PPP, the Point to Point Protocol, or SLIP, the Serial Line Internet Protocol. It uses the UNIX host's native serial ports. It communicates with the UNIX kernel's own TCP/IP implementation via the HP IP tunnel driver (see tun(4)).

Daemon Management Options

auto

Start in `autocall' mode and detach from the controlling terminal to run as a daemon. Initiate a connection in response to a packet specified in the `bringup' category in filter-file. Requires the remote address.

up

When used with auto, bring the link up immediately rather than waiting for traffic. If the link goes down, attempt to restart it (after the call retry delay timer expires) without waiting for an outbound packet.

dedicated

Treat the connection as a dedicated line rather than a demand-dial connection. This option tells pppd to never give up on the connection; that is, if the peer tries to shut down the link, go ahead and do so, but then immediately try to reestablish the connection. Similarly, when first trying to connect, pppd will not give up after sending a fixed number of Configure-Request messages. Hangup events (LQM failures, loss of Carrier Detect) will still cause the device to be closed, just as with dial-up connections, and the Systems file will then be checked for alternate entries. If none are available, the connection will be reestablished after the call retry delay timer expires. Use a short call retry delay timer on dedicated circuits; something like Any;5-30 should work well. Implies up.

nodetach

Don't detach from the controlling terminal in `autocall' mode. When used with log -, this can be useful for watching the progress of the PPP session.

log log-file
Append logging messages to log-file (default: /var/adm/pppd.log).

acct acct-file
Append session accounting messages to acct-file. If acct-file is the same as log-file, the session accounting messages will be interleaved with other logging information.

filter filter-file

Look in filter-file for packet filtering and link management information (default: /etc/ppp/Filter).

debug debug-level

Set the log file verbosity to the following debug-level and each debugging verbosity level also provides the information of all the lower-numbered levels.

0

Daemon start messages

1

Link status messages, calling attempts (the default)

2

Chat script processing, input framing errors

3

LCP, IPCP, PAP and CHAP negotiation

4

LQM status summaries

5

IP interface changes

6

IP message summaries

7

Full LQM reports

8

All PPP messages (without framing)

9

Characters read or written

10

Procedure call messages

11

Internal timers

`exec exec-cmd`

Run ``exec-cmd up addr args'` when the link comes up, and ``exec-cmd down addr args'` when it goes down. Addr is the IP address of the peer, and args is the list of arguments given to pppd.

`nonice`

Run at a normal user process priority, rather than using the `nice()` library routine to elevate pppd scheduling priority to `-10`.

Communications Options

`asynctmap asynct-map`

Set the desired Async Control Character Map to `asynct-map`, expressed in C-style hexadecimal notation (default `0xA0000`).

`noasynctmap`

Disable LCP Async Control Character Map negotiation.

`escape odd-character`

In addition to those characters specified in the PPP Async Control Character Map (which can include only `0x00` through `0x1F`), also apply the escaping algorithm when transmitting `odd-character`. The value of `odd-character` must be between `0x00` and `0xFF`, and cannot be any of `0x5E`, `0x7D` or `0x7E`.

`Odd-character` can be specified as a decimal number, in C-style hexadecimal notation, or as an ASCII character with optional ``^'` control-character notation. For example, the XON character could be

specified as 17, 0x11, or ^Q.

If a character specified with the escape argument, when transformed into its escaped form, would be the same as a character contained in the peer's negotiated Async Control Character Map, a warning will be printed in the log file and the character specified on the command line will not be escaped.

If a character specified with the escape argument, when transformed into its escaped form, would be the same as a character specified in another escape argument on the daemon's command line, pppd will print an error message and exit.

device

Communicate over the named device (default /dev/tty).

comm-speed

Set communications rate to comm-speed bits per second.

ignore-cd

Ignore the state of the CD (Carrier Detect, also called DCD, Data Carrier Detect) signal. This is useful for systems that don't support CD but want to run PPP over a dedicated line.

xonxoff

Set the line to use in-band (software) flow control, using the characters DC3 (^S, XOFF, ASCII 0x13) to stop the flow and DC1 (^Q, XON, ASCII 0x11) to resume. (The default is to use no flow control.) For an outbound connection, this may be specified either in Devices or on the pppd command line.

telnet

When used on an answering pppd command line, negotiate the telnet binary option and understand telnet escape processing. Not for use with device or auto.

Link Management Options

nooptions

Disable all LCP and IPCP options.

noaccomp

Disable HDLC Address and Control Field compression.

noproto comp

Disable LCP Protocol Field Compression.

slip

Use RFC 1055 SLIP packet framing rather than PPP packet framing. Disables all option negotiation, and implies noasyncmap, noipaddress, vjslots 16, novjcid, nomagic, nomru, and mru 1006. Implies vjcomp if

peer sends a header-compressed TCP packet.
extra-slip-end

When running in SLIP mode, prepend a SLIP packet framing character (0xC0) to each frame before transmission, even if this frame immediately follows the previous frame. By default, pppd transmits only one framing character between adjacent SLIP frames.

nomagic

Disable LCP Magic Number negotiation.

mru mru-size

Set LCP Maximum Receive Unit value to mru-size for negotiation. The default is 1500 for PPP and 1006 for SLIP.

nomru

Disable LCP Maximum Receive Unit negotiation, and use 1500 for our interface.

active

Begin LCP parameter negotiation immediately (the default).

passive

Do not send our first LCP packet until we receive an LCP packet from the peer.

timeout restart-time

Set the LCP, IPCP, CCP, PAP, and CHAP option negotiation restart timers to restart-time (default 3 seconds).

lqinterval time

Send Link-Quality-Reports or Echo-Requests every time seconds (default 10 seconds). If the peer responds with a Protocol-Reject, send LCP Echo-Requests every time seconds instead, and use the received LCP Echo-Replies for link status policy decisions.

lqthreshold min/per

Set a minimum standard for link quality by considering the connection to have failed if fewer than min out of the last per LQRs we sent have been responded to by the peer (default 1/5).

echolqm

Use LCP Echo-Requests rather than standard Link-Quality-Report messages for link quality assessment and policy decisions. The peer can override this if it actively tries to configure Link Quality Monitoring unless the nolqm parameter is also specified.

nolqm

Don't send or recognize Link-Quality-Report messages. If echolqm is also specified, Echo-Request messages will be used to detect link failures.
idle idle-time[/session-idle-time]

Shut down the link when idle-time seconds pass without receiving or transmitting a packet specified in the 'keepup' category in the filter file (default is to never consider the link idle).

If session-idle-time is specified and any TCP sessions are open, shut down the link when session-idle-time seconds pass without receiving or transmitting a packet.

max-configure tries

Set the PPP Max-Configure counter (the maximum number of Configure-Requests sent without a response) to tries .

max-terminate tries

Set the PPP Max-Terminate counter (the maximum number of Terminate-Requests sent without a response) to tries .

max-failure tries

Set the PPP Max-Failure counter (the maximum number of Configure-Naks sent without a positive response) to tries .

IP Options

local:remote

The address of this machine, followed by the expected address for the remote machine. Can be specified either as symbolic names or as literal IP addresses, if their addresses cannot be discovered locally without using the PPP link.

Both addresses are optional, but a colon by itself is not valid, and the remote address is required when running as a daemon in 'autocall' mode.

If only local: is specified when receiving an incoming call, the remote address will be discovered during IPCP IP-Address negotiations.

If either address is followed by a tilde character (~), or if the tilde appears alone, pppd accepts the IP address given by the peer during IPCP negotiations, whether for the local end or the peer's end of the link. (not available in SLIP mode)

Because SLIP cannot perform option negotiations, including IPCP, both addresses should normally be specified, and the tilde option is unavailable. To obtain a similar "feature", the peer must provide the IP address textually during the login process, and a new value must be obtained using the Systems file '\A' chat script feature (see

ppp.Systems(4).
netmask subnet-mask

Set the subnet mask of the interface to subnet-mask, expressed either in C-style hexadecimal (e.g. 0xfffff00) or in decimal dotted-quad notation (e.g. 255.255.255.0). The default subnet mask will be appropriate for the network (class A, B, or C), assuming no subnetting.
noipaddress

Disable IPCP IP-Address negotiation.
vjcomp

Enable RFC 1144 `VJ' Van Jacobson TCP header compression negotiation with 16 slots and slot ID compression (this is the default with PPP framing). `VJ' compression is enabled by default for async connections, and disabled by default for sync connections.
novjcomp

Disable RFC 1144 `VJ' Van Jacobson TCP header compression (this is the default with SLIP framing, until the peer sends a header-compressed TCP packet).
vjslots vj-slots

Set the number of VJ compression slots (min 3, max 256, default 16).
novjcid

Disable VJ compression slot ID compression (enabled by default).
rfc1172-vj

Backwards compatibility with older PPP implementations (4-byte VJ configuration option), but with the correct option negotiation value of 0x002d.
rfc1172-typo-vj

Backwards compatibility with older PPP implementations (4-byte VJ configuration option) that conform to the typographical error in RFC 1172 section 5.2 (Compression-Type value 0x0037).
rfc1172-addresses

Backwards compatibility with older PPP implementations that conform to RFC 1172 section 5.1 (IP-Addresses, IPCP configuration option 1) and not with the newer RFC 1332 (IP-Address, IPCP configuration option 3), but that respond with something besides a Configure-Reject when they receive an IPCP Configure-Request containing an option 3.

Authentication Options

requireauth

Require either PAP or CHAP authentication.

requirechap

Require CHAP authentication as described in RFC 1334.

requiremschap

Require MS-CHAP authentication.

requirepap

Require PAP authentication.

rechap interval

Demand that the peer re-authenticate itself (using CHAP) every interval seconds. If the peer fails the new challenge, the link is terminated.

name identifier

Provide the identifier used during PAP or CHAP negotiation. This option is necessary if the PPP peer requires authentication. The default value is the value returned by the gethostname(2) system call or the hostname(1) command.

MicroSoft Compatibility Options

ms-dns address

Set the MS DNS address to provide to the peer. First occurrence of this option on the command line sets the primary address; the second occurrence sets the secondary address.

ms-nbns address

Set the MS NBNS address to provide to the peer. First occurrence of this option on the command line sets the primary address; the second occurrence sets the secondary address.

Encryption Options

Encryption is not currently available in software exported from the USA. However, customer may contact sales@progressive-systems.com to obtain encryption functionality.

Link Compression Options

compress

Offer all supported link compression types (currently only Predictor-1) when negotiating. The default is to propose and accept no link compression type.

compress-pred1

Accept any supported compression type, but prefer Predictor type 1 compression.

nopred1

Never use Predictor-1 compression.

Re: Actius MM10 modem

LOG FILE

Status information is recorded in the log file (/var/adm/pppd.log by default) by each copy of pppd running on a single machine. Each line in the file consists of a message preceded by the date, the time, and the process ID number of the daemon writing the message. The quantity and verbosity of messages are controlled with the debug option and with the log filter (see ppp.Filter(4)).

Each packet that brings up the link (at debug level 1 or more), each packet that matches the log filter (at any debug level), or any packet when the debug level is 7 or more writes a one-line description of the packet to the log file. The first item of the message is the protocol (tcp, udp, icmp, or a numeric protocol value). For ICMP packets, the keyword icmp is followed by the ICMP message type and sub code, separated by slashes. After the protocol comes an IP address and optionally a TCP or UDP port number, followed by an arrow indicating whether the packet was sent (->) or received (<-), followed by another address and port number, followed by the length of the packet in bytes before VJ TCP header compression, followed by zero or more keywords. For transmitted packets, the first IP address is the source address, while for received packets, the first IP address is the destination address. Well known TCP and UDP port numbers will be replaced by the name returned by the getservbyport() library function. The keywords and their meanings are:

frag

The packet is a middle or later part of a fragmented IP frame.

syn

The packet has the TCP SYN bit set.

fin

The packet has the TCP FIN bit set.

bringup

The transmitted packet matches the bringup filter and is bringing up the link.

!keepup

the packet has been rejected by the keepup filter.

!pass

The packet has been rejected by the pass filter.

dial failed

The packet was dropped because pppd is waiting for the call retry timer to expire.

(c)

The received packet is VJ TCP header compressed.

(u)

The received packet is VJ TCP header uncompressed.

For example, the following log file line

```
9/6-14:06:26-83 tcp 63.1.6.3/1050 -> 8.1.1.9/smtp 44 syn
```

indicates that at 2:06:26 PM on September 6, process ID 83 sent a 44-byte TCP packet with the SYN bit set from port 1050 on 63.1.6.3 to the SMTP port on 8.1.1.9.

SIGNALS

Upon reception of the following signals, pppd closes and reopens the log file, re-reads the filter and key files, then takes the indicated actions:

SIGKILL

Don't use this. Never, never use this. Since pppd won't be able to shut down gracefully, it will leave your serial interfaces (whether /dev/tty) and your IP tunnel driver in some unknown state. Use SIGTERM instead, so pppd will shut down cleanly, and leave the system in a well-defined state.

SIGINT

Disconnect gracefully from an active session. If in `autocall' mode, reset the call retry delay timer and call retry backoff interval. If up was specified, attempt to re-establish the link. Exit if not in `autocall' mode.

SIGHUP

Disconnect abruptly from an active session. If up was specified, attempt to re-establish the link. Exit if not in `autocall' mode.

SIGTERM

Disconnect gracefully from an active session, clean up the state of any serial and IP interfaces that are open, then exit.

SIGUSR1

Increment the verbosity level for debugging information written to the log file.

SIGUSR2

Reset the debugging verbosity level to the base value (1 unless debug 0 was supplied on the command line).

SIGALRM

Take no action except to re-read the filter and key files.

EXAMPLE

To run a pair of daemons on `oursystem`, one maintaining a constant link with `backbonesystem` and the other prepared to initiate outbound calls to a neighboring machine named `theirsystem`, add the following to `/sbin/rc2.d/S522ppp`:

```
if [ -f /etc/ppp/Autostart ]; then
/etc/ppp/Autostart
fi
```

Then make `/etc/ppp/Autostart` look like this:

```
#!/bin/sh

PATH=/usr/etc:/bin:/usr/bin

if [ -f /var/adm/pppd.log ]; then
    mv /var/adm/pppd.log /var/adm/OLDpppd.log
fi

echo -n "Starting PPP daemons:" >/dev/console

pppd oursystem:backbonesystem auto up
    (echo -n ' backbonesystem') >/dev/console
pppd oursystem:theirsystem auto idle 120
    (echo -n ' theirsystem') >/dev/console

echo '.' >/dev/console
```

To allow a PPP implementation running on `theirsystem` to dial into `oursystem`, insert the following into `/etc/passwd` on `oursystem`:

```
Pthem:?:105:20:Their PPP:/etc/ppp:/etc/ppp/Login
```

where group 20 is the gid of the ppp group which owns `/usr/etc/pppd`, and `/etc/ppp/Login` is an executable shell script that looks something like

```
#!/bin/sh
PATH=/usr/bin:/usr/etc:/bin
msg n
stty -tostop
exec pppd `hostname` :
RECOMMENDATIONS
```

Use host names when running `/etc/ppp/Autostart` from `/sbin/rc2.d/S522ppp` only if they are known locally. If a PPP connection to a DNS server would be required to resolve a host name, use its literal IP address instead.

EXTERNAL INFLUENCES

Environment Variables

comp.os.linux.portable: Re: Actius MM10 modem

The environment variable PPPHOME, if present, specifies the directory in which pppd looks for its configuration files (Filter and Auth for all connections, along with Systems, Devices, and Dialers if the connection is `outbound'). You can specify PPPHOME either in the Autostart script or in an incoming connection's Login script. If PPPHOME is not present, pppd will expect to find its configuration files in /etc/ppp/*.

SECURITY CONCERNS

pppd should be mode 4750, owned by root, and executable only by the members of the group containing all the incoming PPP login `users'.

AUTHOR

pppd was developed by the Progressive Systems.

SEE ALSO

tun(4), ppp.Auth(4), ppp.Devices(4), ppp.Dialers(4), ppp.Filter(4), ppp.Keys(4), ppp.Systems(4), RFC 1548, RFC 1549, RFC 1332, RFC 1333, RFC 1334, RFC 1172, RFC 1144, RFC 1055, ds.internic.net:/internet-drafts/draft-ietf-pppext-compression-04.txt.

STANDARDS CONFORMANCE

HP PPP implements the IETF Proposed Standard Point-to-Point Protocol and many of its options and extensions, in conformance with RFCs 1548, 1549, 1332, 1333, 1334, and 1144. It can be configured to be conformant with earlier specifications of the PPP protocol, as described in RFCs 1134, 1171, and 1172. It implements the nonstandard SLIP protocol as described in RFCs 1055 and 1144.

--

Cheers, <http://www.dotware.co.uk>

Jim <http://www.dotware-entertainment.co.uk>

```
Unix Sex: { look; find; talk; grep; touch; finger; find; flex; unzip;
mount; workbone; fsck; yes; gasp; fsck; yes; eject; umount; makeclean;
zip; split; done; exit }
```