NAME

pptp − PPTP driver

SYNOPSIS

pptp *<pptp-server-IP> <pptp-options> [ppp-options]* ...

DESCRIPTION

pptp establishes the client side of a Virtual Private Network (VPN) using the Point-to-Point Tunneling Protocol (PPTP). Use this program to connect to an employer's PPTP based VPN, or to certain cable and ADSL service providers.

By default, **pptp** establishes the PPTP call to the PPTP server, and then starts an instance of **pppd** to manage the data transfer. However, **pptp** can also be run as a connection manager within **pppd**.

OPTIONS

The first non−option argument on the **pptp** command line must be the host name or IP address of the PPTP server.

All long options (starting with "−−") are interpreted as pptp options, and a fatal error occurs if an unrecognised option is used.

All command−line arguments which do not start with "−" are interpreted as ppp options, and passed as is to **pppd** unless −−**nolaunchpppd** is given.

−−phone <number>
> Pass <number> to remote host as phone number

−−nolaunchpppd
> Do not launch **pppd** but use stdin as the network connection. Use this flag when including **pptp** as a **pppd** connection process using the **pty** option. See EXAMPLES.

−−quirks <quirk>
> Work around a buggy PPTP implementation, adopts special case handling for particular PPTP servers and ADSL modems. Currently recognised values are BEZEQ_ISRAEL only

−−debug
> Run in foreground (for debugging with gdb)

−−sync Enable Synchronous HDLC (pppd must use it too)

−−timeout <secs>
> Time to wait for reordered packets (0.01 to 10 secs)

−−nobuffer
> Completely disables buffering and reordering of packets. Any −−timeout specified will be ignored.

−−idle-wait <secs>
> Time to wait before sending a control connection echo request. The RFC2637 default is 60 seconds.

−−max-echo-wait <secs>
> Time to wait for an echo reply before closing the control connection. The RFC2637 default is 60 seconds.

−−logstring <name>
> Use <name> instead of 'anon' in syslog messages

−−localbind <addr>
> Bind to specified IP address instead of wildcard

−−rtmark <n>
> Use specified policy routing mark for all packets. This causes both the TCP control connection's packets as well as the GRE packets to bear the given policy routing / netfilter mark. This can be used with *ip rule* (from iproute2) to use a separate routing table for the pptp client.

(requires root privileges or the CAP_NET_ADMIN capability.)

−−nohostroute

Do not configure a host route pointing towards the PPTP server. (cf. ROUTING below)

−−loglevel <level>

Sets the debugging level (0=low, 1=default, 2=high)

−−test-type <n>

Enable packet reordering tests that damage the integrity of the packet stream to the server. Use this only when testing servers. Zero is the default, and means that packets are sent in the correct order. A value of one (1) causes a single swap between two packets, such that the sequence numbers might be 1 2 3 4 6 5 7 8 9. A value of two (2) causes ten packets to be buffered, then sent out of order but ascending, such that the sequence numbers might be 1 2 3 4 16 6 7 8 9 10 11 12 13 14 15 17 18 19 20. A value of three (3) causes ten packets to be buffered, then sent in the reverse order, like this; 1 2 3 4 16 15 14 13 12 11 10 9 8 7 6 5 17 18 19 20.

−−test-rate <n>

Sets the number of packets to pass before causing a reordering test. Default is 100. Has no effect if test-type is zero. The result of test types 2 and 3 are undefined if this value is less than ten.

ROUTING

When PPTP is used in conjunction with a default route on top of the tunnel (or just any route encompassing the PPTP server), the mechanics of routing would cause the PPTP packets themselves to be routed over the tunnel. This would result in an encapsulation loop, destroying connectivity.

pptp by default works around this by looking up the route towards the PPTP server at startup and configures a host route with that data. This essentially "freezes" routing for PPTP packets at the startup configuration. This behaviour can be disabled with **--nohostroute** if undesired (like when using **--rtmark** to implement policy routing).

NB: the route added by **pptp** is currently not deleted at exit!

QUIRKS

BEZEQ_ISRAEL

modifies packets to interoperate with Orckit ADSL modems on the BEZEQ network in Israel.

EXAMPLES

Connection to a Microsoft Windows VPN Server

 pppd noauth nobsdcomp nodeflate require−mppe−128 name domain\\\\username remotename PPTP pty "pptp 10.0.0.5 −−nolaunchpppd"

Note that the **chap−secrets** file used by **pppd** must include an entry for domain\\username

STATISTICS

The pptp process collects statistics when sending and receiving GRE packets. They are intended to be useful for debugging poor PPTP performance and for general monitoring of link quality. The statistics are cumulative since the pptp process was started.

The statistics can be viewed by sending a SIGUSR1 signal to the "GRE-to-PPP Gateway" process, which will cause it to dump them to the system logs (at the LOG_NOTICE level). A better way to present the

statistics to applications is being sought (e.g. SNMP?).

The following statistics are collected at the time of writing (April 2003):

rx accepted
> the number of GRE packets successfully passed to PPP

rx lost the number of packets never received, and presumed lost in the network

rx under win
> the number of packets which were duplicates or had old sequence numbers (this might be caused by a packet-reordering network if your reordering timeout is set too low)

rx over win
> the number of packets which were too far ahead in the sequence to be reordered (might be caused by loss of more than 300 packets in a row)

rx buffered
> the number of packets which were slightly ahead of sequence, and were either buffered for reordering, or if buffering is disabled, accepted immediately (resulting in the intermediate packets being discarded).

rx OS errors
> the number of times where the operating system reported an error when we tried to read a packet

rx truncated
> the number of times we received a packet which was shorter than the length implied by the GRE header

rx invalid
> the number of times we received a packet which had invalid or unsupported flags set in the header, wrong version, or wrong protocol.

rx acks the number of pure acknowledgements received (without data). Too many of these will waste bandwidth, and might be solved by tuning the remote host.

tx sent the number of GRE packets sent with data

tx failed
> the number of packets we tried to send, but the OS reported an error

tx short
> the number of times the OS would not let us write a complete packet

tx acks the number of times we sent a pure ack, without data

tx oversize
> the number of times we couldn't send a packet because it was over PACKET_MAX bytes long

round trip
> the estimated round-trip time in milliseconds

SEE ALSO
> *pppd*(8)
>
> Documentation in */usr/share/doc/pptp-linux*

AUTHOR
> This manual page was written by James Cameron <james.cameron@hp.com> from text contributed by Thomas Quinot <thomas@debian.org>, for the Debian GNU/Linux system. The description of the available statistics was written by Chris Wilson <chris@netservers.co.uk>. Updates for the Debian distribution by Ola Lundqvist <opal@debian.org>.