

FreeBSD interoperability with Cisco VPN Concentrator 3000 series

James Flemer
jflemer@alum.rpi.edu

October 10, 2002

Contents

1	Example Network Layout	1
2	IPsec	1
3	PPTP	1
3.1	Concentrator Configuration	1
3.2	Configuring mpd	2
3.2.1	mpd.links	2
3.2.2	mpd.conf	2
3.2.3	mpd.secret	3
3.3	Running mpd	3
3.4	Routing	4

1 Example Network Layout

For the purposes of this paper we will assume the following network layout. The *public* side of the network is the 192.168.0.0 Class B network, and the *private* side the 10.0.0.0 Class A network. The public side is the side the VPN clients connect from to be able to access the private network. The VPN concentrator has the following addresses on it's interfaces: 192.168.0.2 for the public side, and 10.0.0.2 on the private side.

We will assume that the client has the public address 192.168.2.42 for all examples, and that for it is assigned 10.0.2.42 for any VPN session it establishes.

2 IPsec

Good luck, I tried it and it did not work. The isakmpd port is lacking necessary features such as ISAMKP_CFG_SET, and XAUTH. If you have any success with IPsec interoperability please email me.

3 PPTP

FreeBSD supports PPTP with the netgraph(4) devices, and the mpd port. The PPTP implementation on the 3000 series is a bit lacking, but it is possible to work around the faults.

3.1 Concentrator Configuration

The 3000 series concentrator must be configured to allow PPTP connections and CHAP authentication. We will also assume that there some user authentication configured and there exists a valid login `grenel` with a password `adopt_a_sweed`.

3.2 Configuring mpd

The FreeBSD port for mpd installs all the configuration files in `/usr/local/etc/mpd`. Some of the files in this directory may contain plain-text passwords, so it is recommended that it be only readable by root.

3.2.1 mpd.links

The `mpd.links` file defines the individual links that are available to mpd. This file includes modems, ISDN devices, PPTP connections, and PPPoE connections. For our situation we need to define a PPTP link like the following:

```
pptp192:
    set link type pptp
    set pptp peer 192.168.0.2
    set pptp enable originate outcall
```

This link has the label `pptp192`, and will attempt to connect to a PPTP peer at the public address 192.168.0.2.

3.2.2 mpd.conf

The `mpd.conf` file is the general configuration file for mpd. It contains labeled sets of commands, we will add two sets for setting up the connection to the Cisco VPN. The first should look like the following:

```
ciscovpn:
    new -i ng0 ciscovpn pptp192
    set bundle authname "grenel"
    set ipcp ranges 10.0.2.42/8 192.168.0.2/16
    load ciscopptp
    open
```

The first line tells mpd to create (or reconfigure) the `ng0` interface¹ with the symbolic name `ciscovpn`, using the link definition `pptp192`. Next we set the

¹Use a different interface here if `ng0` is already in use.

username to use for authentication to the VPN. The `ipcp ranges` command tells mpd what addresses it will accept for the endpoints of the PPTP link. The first address says that we want 10.0.2.42, but will accept anything in the entire Class A; the second address says the other end should be 192.168.0.2, but anything in the whole Class B is acceptable. Finally have mpd load the set of commands with the label `ciscopptp`, and then open the connection.

Now we need to add the section for `ciscopptp` which will set up all the options needed to work with PPTP implementation on the 3000 series concentrators. It should look like this:

```
ciscopptp:
    set bundle disable compression encryption
    set bundle no crypt-reqd
    set iface idle 0
    set ipcp disable vjcomp
    set ipcp enable req-pri-dns req-sec-dns
    set link keep-alive 0 0
    set link disable pap chap
    set link disable acfcomp protocomp
```

This configures PPTP to disable encryption and compression, turn off keep-alives, and disable idle timeout. It also enables the client to request a primary and secondary DNS server from the concentrator.² Consult the mpd documentation for details on each of these commands if you are interested.

Depending on the configuration on your VPN concentrator you may be able to use encryption and/or compression. See the notes in the concentrator documentation on PPTP authentication and encryption.

If you would like this connection to be the default for mpd, and to be opened automatically when mpd starts, add the following to `mpd.conf`:

```
default:
    load ciscovpn
```

3.2.3 mpd.secret

The final file needed to connect is the `mpd.secret` file, which contains the passwords for each username (`authname`). This file should only be readable by root. All that we need in this file is one line that maps the username `grenel` to the password `Adopt_A_Sweed`:

```
grenel      "Adopt_A_Sweed"
```

3.3 Running mpd

If you have made `ciscovpn` the default for mpd, then connecting is as simple as just running `mpd`. If you did not make it default then you need to run

²The DNS servers returned by the concentrator are currently ignored by mpd.

`mpd ciscovpn`. That is it. If you have configured everything correctly you will see `mpd` printing out lots of messages reporting status as it negotiates the connection. When you see the following message then you will be connected.³

```
[ciscovpn] IFACE: Up event
```

The `mpd` process is actually interactive at this point, and if you hit enter you will get a command prompt. There are several commands you can issue to view link statistics and more, see the `mpd` documentation for details or just type `help`. To close the connection, just type `quit`.

3.4 Routing

Unfortunately, this does not work completely. It successfully establishes the PPTP connection, but cannot send anything over it. The problem is that the PPTP implementation for the concentrator forces its end of the PPP link to have the same IP as the address of its public interface (192.168.0.2 in this network). This causes FreeBSD to have routing problems, because the default gateway becomes 192.168.0.2 (via `ng0`), but in order to use that tunnel it has to send GRE packets to 192.168.0.2.

The solution to this is as follows. Once the PPTP link is up, you need to re-address the `ng0` interface and then change your default route. In the example network, you have to execute the following commands (assuming we are assigned 10.0.2.42 for our side of the link⁴):

```
# ifconfig ng0 inet 10.0.2.42 10.0.0.2 netmask 0xffffffff
# route delete default
# route add default -interface ng0
```

At this point you should be set, the default route will be over the PPTP link. You may also need to manually set your DNS servers if you want to use one on the private side.

³If you do not see this, you will have to figure out what is failing, and recheck your configurations.

⁴You may want to run `ifconfig ng0` first to verify the address of your end.