PPTP-VPN-Tunnel in den Fachbereich Informatik mit Linux als Client-Betriebssystem

Christian J Dietrich <christian.dietrich@informatik.fh-gelsenkirchen.de>

Anmerkung: Im folgenden Text sollte <USERNAME> immer durch den Pool-Benutzernamen, <PASSWORT> durch das Pool-Passwort ersetzt werden, <PPTP-GATEWAY-IP> durch die IP-Adresse des PPTP-Gateways, <PPTP-NETZWERK> durch die Netzwerk-Adresse des PPTP-Netzes und <PPTP-NETZWERKMASKE> durch die Netzmaske des PPTP-Netzwerkes.

1 PPTP installieren/checken

Das Paket pptp muss installiert sein. Das prüft man – wie auf http://www.poptop.org beschrieben – nach (dann bleibt einem – falls es schon drauf ist – eine Menge Arbeit erspart).

Falls es nicht drauf ist müssen – wie ebenfalls auf http://www.poptop.org beschrieben – Kernel und pppd installiert werden.

2 PPPD Peer konfigurieren

Nun muss ein PPPD Peer konfiguriert werden. Die PPPD Peers sind quasi die Netzwerkverbindungen unter Windows.

Wenn man pptp.php-gtk (eine GUI für pptp) installiert hat, kann man diese GUI nutzen, um die Verbindung einzurichten. Im Folgenden beschreibe ich die manuelle Vorgehensweise.

Hierzu legt man eine Datei "fh_tunnel" mit folgendem Inhalt in /etc/ppp/peers/ an:

```
--- Datei-Anfang
# tunnel fh_tunnel

# name of tunnel, used to select lines in secrets files
remotename fh_tunnel

# name of tunnel, used to name /var/run pid file
linkname fh_tunnel

# name of tunnel, passed to ip-up scripts
ipparam fh_tunnel

# data stream for pppd to use
pty "pptp <PPTP-GATEWAY-IP> --nolaunchpppd "
```

```
# domain and username, used to select lines in secrets files
# Hier muss der Benutzername eingetragen werden, der auch im Pool verwendet wird
name <USERNAME>

# wenn man drauf besteht, dass die Verbindung abgebrochen wird, wenn sich nicht
# auf eine Verschlüsselung geeinigt werden kann, folgendes stehen lassen
require-mppe

# do not require the server to authenticate to our client
noauth

# adopt defaults from the pptp-linux package
file /etc/ppp/options.pptp

# end of tunnel file

--- Datei-Ende
```

3 Passwort eintragen

Das Passwort für die Verbindung muss in die Datei `/etc/ppp/chap-secrets` eingetragen werden. Die entsprechende Zeile sieht dann so aus:

```
<USERNAME> fh_tunnel <PASSWORT> *
```

4 Verbindung ausprobieren

Dann kann man (als root oder mit /usr/sbin/pppd als SUID-root) mit dem Befehl

```
$ pppd call fh_tunnel dump debug logfd 2 nodetach
```

versuchen eine Verbindung aufzubauen.

Nachdem man die leicht kryptische Ausgabe bewundert hat, muss man beurteilen, ob es geklappt hat oder nicht :-) Geklappt hat die Authentifizierung auf jeden Fall, wenn im Debug-Log die Meldung

```
rcvd [CHAP Success id=0x1 "Welcome to vpn-tunnel."]
```

auftaucht.

Dann sollte dem Client noch eine IP-Adresse zugewiesen werden, zu erkennen an:

```
local  IP address <EINE-IP-ADRESSE>
```

Das ip-up Skript sollte sich auch ohne Fehler beenden:

```
Script /etc/ppp/ip-up finished (pid 4135), status = 0x0
```

Dies erkennt man am status code. Wenn der != 0x0 ist, lief etwas schief.

Wenn man in der Peer-Konfiguration `require-mppe` nicht auskommentiert hat, folgt als letzte Meldung

```
MPPE enforced
```

Dann hat man es mit dem Verbindungsaufbau geschafft. Weiter geht's mit dem Anpassen der Routing-Tabelle.

5 Routing anpassen

Hier als Beispiel meine Routing-Tabelle (IP-Adressen abgeändert), wenn ich den Tunnel nicht gestartet habe:

```
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.17.0    0.0.0.0         255.255.255.0   U     0      0        0 eth1
0.0.0.0         192.168.17.5    0.0.0.0         UG    0      0        0 eth1
```

Es gibt eine Default-Route zu meiner Firewall dann weiter ins Internet (unterer Eintrag) und das lokale Netz (192.168.17.0/255.255.255.0) ist über ein lokales Interface (eth1) erreichbar.

Sobald der Tunnel gestartet wurde, legen wahrscheinlich die meisten ip-up-Skripte automatisch eine Host-Route zum VPN-Gateway an, sodass die Routing-Tabelle dann wie folgt aussieht:

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.16.24.1     0.0.0.0         255.255.255.255 UH    0      0        0 ppp0
192.168.17.0    0.0.0.0         255.255.255.0   U     0      0        0 eth1
0.0.0.0         192.168.17.5    0.0.0.0         UG    0      0        0 eth1
```

Bei der o.g. automatisch angelegten Route handelt es sich allerdings meist um eine Host-Route, das bedeutet, lediglich ein einziger Host (die interne IP des VPN-Tunnel-Servers) kann über diese Route erreicht werden (oberster Eintrag). Da aber auch das

dahinterliegende Netz erreichbar sein soll, legt man je nach pppd-Version entweder folgendes Skript fh_tunnel in /etc/ppp/ip-up.d/ an

```
---Datei-Anfang
#!/bin/bash

INTERFACE=$1

export PATH=/sbin:/usr/sbin:/bin:/usr/bin

if [ "$LINKNAME" = "fh_tunnel" ]; then
      /sbin/route add -net <PPTP-NETZWERK> netmask <PPTP-NETZWERKMASKE> \
      gateway 172.16.24.1 dev $INTERFACE
fi

exit 0
---Datei-Ende
```

Oder man fügt den relevanten Teil direkt in die Datei /etc/ppp/ip-up.local. Die resultierende Routing-Tabelle sollte wie folgt aussehen:

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.16.24.1     0.0.0.0         255.255.255.255 UH    0      0        0 ppp0
192.168.17.0    0.0.0.0         255.255.255.0   U     0      0        0 eth1
172.16.0.0      172.16.24.1     255.255.0.0     UG    0      0        0 ppp0
0.0.0.0         192.168.17.5    0.0.0.0         UG    0      0        0 eth1
```

6 SAMBA-Test

Die PPTP-Einwahl ist nun abgeschlossen. Abhängig von der Netzkonfiguration (z.B. im FB5) hat man nun noch keinen Zugriff auf die Netze der Lehr- und Forschungsbereiche. Um Zugriff auf eins dieser Teilnetze zu bekommen, muss man sich einem Fileserver gegenüber authentifizieren.

Um zu überprüfen, ob der Fileserver erreichbar ist, kann man probehalber mit smbclient versuchen, ein Freigaben-Listing zu bekommen. <USERNAME> ist hierbei wieder der Pool-Benutzername. <FILESERVER> ist beispielhaft die IP-Adresse eines SAMBA-Fileservers, <ARBEITSGRUPPE> ist der Name der Arbeitsgruppe.

```
$ smbclient -L <FILESERVER> -U  <USERNAME>
```

Nach Eingabe des Passworts sollte man eine Ausgabe ähnlich der folgenden erhalten.

```
leo:~ # smbclient -L <FILESERVER> -U <USERNAME>
added interface ip=192.168.17.12 bcast=192.168.17.255 nmask=255.255.255.0
Password: <PASSWORT>
Domain=[INF_502] OS=[Unix] Server=[Samba 3.0.2a]

        Sharename      Type      Comment
        ---------      ----      -------
        netlogon       Disk      Network Logon Service
        tmp            Disk      Temporary file space
        transfer       Disk      Transferverzeichnis
        sophos         Disk      Sophos Instalation
        IPC$           IPC       IPC Service (Samba Server)
        ADMIN$         IPC       IPC Service (Samba Server)
        poolprinter    Printer   Created by redhat-config-printer 0.6.x
        printer        Printer   printer
        <USERNAME>     Disk      Home Directories

        Server               Comment
        ---------            -------
        BRAVA0
        STROMBOLI            Samba Server

        Workgroup            Master
        ---------            -------
        ARBEITSGRUPPE        ORACLE9I
        INF_502              STROMBOLI
```

Dann mountet man sein Home-Directory auf diesem Fileserver z.B. per:

```
#! /bin/sh

smbmount //<FILESERVER>/<USERNAME> /home/chris/mnt/<FILESERVER>/<USERNAME>/ \
-o username=<USERNAME>%<PASSWORT>,fmask=644,dmask=755, \
ip=<FILESERVER>,workgroup=<ARBEITSGRUPPE>
```

Dies kann man automatisieren, indem man den Code-Schnipsel als Ergänzung in die oben angelegte Datei /etc/ppp/ip-up.d/fh_tunnel oder in die /etc/ppp/ip-up.local unter das route-Kommando einfügt.

Bevor man den VPN-Tunnel abbaut, sollte man das Home-Directory wieder sauber unmounten.