CREATING PPTP CLIENT WITH AUTO-CONNECT SCRIPT

Step 1: Update and Install.

sudo apt-get update -y

sudo apt-get upgrade -y

sudo apt-get install -y pptp-linux

Step 2: Creating a PPTP connection configuration.

A. ADDING SECRET ON MIKROTIK ROUTER:

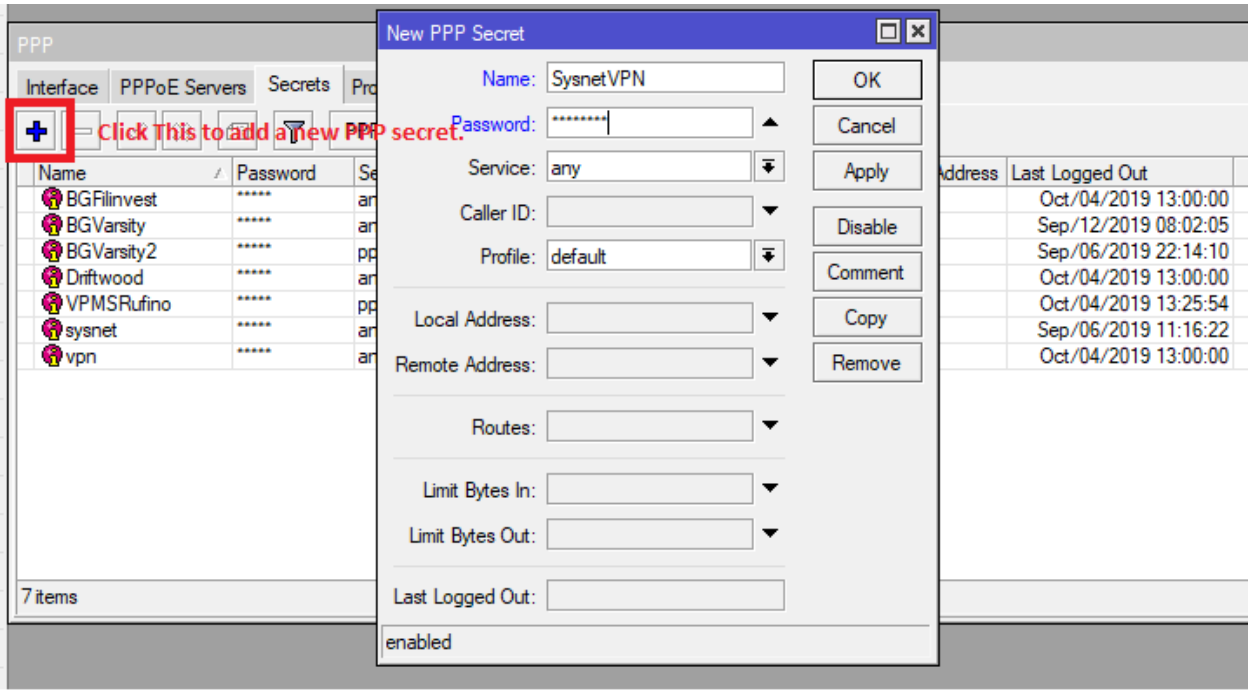

Figure 1: Adding a new PPP secret on MikroTik Router

B. CREATING PPTP CONNECTION ON UBUNTU LINUX TERMINAL.

sudo pptpsetup --create [vpn name] --server [Host] --username [PPP name] -- password [PPP password] --encrypt option –start
Note:

- The *"pptpsetup"* creates a configuration file that gets placed in **/etc/ppp/peers/** which is a superuser protected directory so you will need to run sudo to view its content.

Command on viewing the restricted file:

sudo ls -l /etc/ppp/peers/

sudo cat /etc/ppp/peers/[vpn name]

C. CHECKING OF CONFIGURATION.

Upon the successful application of creating a VPN on *"pptpsetup"* a good System Administrator should properly check its configuration on **/etc/ppp/chap-secrets**

Format should be:

```
# Secrets for authentication using CHAP
# client        server          secret          IP addresses
[PPP name]      [VPN name]   "[PPP password]"        *
```

Where:

- **PPP name** is the MikroTik PPP name.
- **VPN name** is the name of your PPTP connection.
- **PPP password** is the MikroTik PPP password.
- ***** is the IP Address you assigned from the server.

Also upon a successful application of creating VPN through *"pptpsetup"* Linux system will creating a .txt file on **/etc/ppp/peers/[vpn name]** for which it will list down your host connection.

Format should be:

```
pty "pptp [Host] --nolaunchpppd"
name [VPN name]
remotename [PPP name]
ipparam [PPP name]
persist
refuse-pap
refuse-chap
refuse-mschap
maxfail 0
require-mppe-128
file /etc/ppp/options.pptp
mru 1500
mtu 1500
```

D. CONNECTION/DISCONNECTION

```
sudo pon [VPN name] updetach
```

- Connection of PPTP Client to PPTP Server.

```
sudo poff -a
```

- Disconnection of every active PPTP connection.

Step 3: Linux Bootup Script.

On Ubuntu 16.04, rc.local script is already implemented but on Ubuntu 18.04 you need to create your own rc.local file on **/etc.**

To create the file:

```
touch rc.local
```

- To create the file.

```
sudo nano rc.local
```

- To edit the file

```
sudo nano -B /etc/rc.local
```

- To create and edit the file using nano text editor to backup the rc.local file before making any changes to it.

On editing the file format should be like this:

```
#!/bin/bash or #!bin/sh

vpn="on"
if [ $vpn = on ]; then
printf "\nYour Description\n"
pon [VPN name] updetach
printf "Add Internet traffic route through ppp0\n"
sudo route add -net "0.0.0.0/0" dev "ppp0"
netstat -a | grep "/var/run/pptp/"
fi

exit 0
```

Make your rc.local file executable by:

```
sudo chmod +x rc.local
```

Run the file to test your script by:

```
./rc.local
```

This procedure will make the Operating System start the script upon booting or rebooting.

Step 4: Linux Auto Reconnect Script.

Upon testing and successful implementation of your scripts. Adding this will make you auto reconnect the PPTP connection once it was disconnected to the internet. On **/bin** create a file name vpnauto.sh by:

```
sudo nano vpnauto.sh
```

- To create and edit the file.

Format should be like this:

```
#!/bin/bash
date
sudo poff
sleep 10s
sudo pon [VPN name] updetach persist
exit
```

Make your vpnauto.sh file to be executable by:

```
sudo chmod +x vpnauto.sh
```

Run the executable vpnauto.sh file by:

```
./vpnauto.sh
```

Step 5: Testing your Connection

Since you have successfully implemented every command and script given above you must test it before putting it on production. Restart the device or remove its LAN cable to check if it will automatically reconnect.

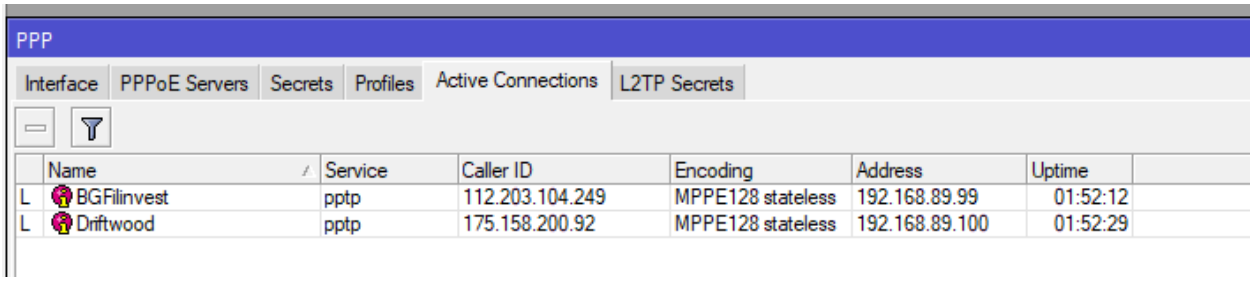

Figure 2: Successful Application of PPTP connection.