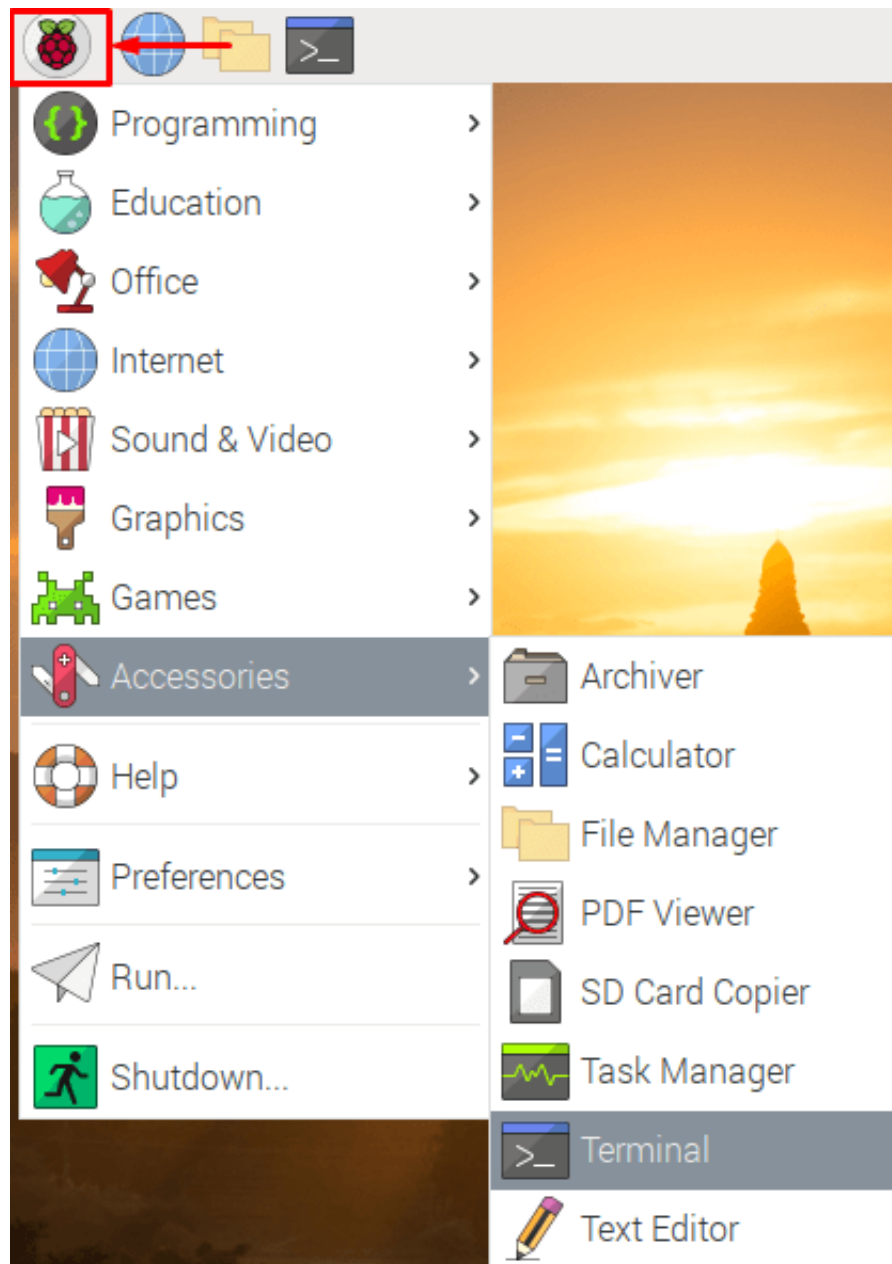


How to setup PPTP for Raspberry Pi Desktop (Debian)

The following is required for setting a PPTP-VPN connection on Raspberry Pi.

- A working installation of Debian (This guide uses the official supported Raspberry Pi operating system based on Debian Buster, version February 2020)
- Own a premium PureVPN account (If you do not already own one, you can buy a subscription from [here](#))

1 Go to the **Menu bar** and select/open the “**Terminal**” from there.



2 Now you need to install PPTP packages. Type the following to install it:

- **sudo bash**
- **sudo apt-get install pptp-linux pptpd ppp curl**

```
pi@raspberrypi:~ $ sudo bash
root@raspberrypi:/home/pi# sudo apt-get install pptp-linux pptpd ppp curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.64.0-4).
ppp is already the newest version (2.4.7-2+4.1).
pptp-linux is already the newest version (1.10.0-1).
pptpd is already the newest version (1.4.0-11+b1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@raspberrypi:/home/pi#
```

3 Run the following command to open the Editor and enter details as mentioned:

- **sudo nano /etc/ppp/chap-secrets**

4 Now edit the chap-secrets file as shown below:

vpnusernamePPTPvpnpassword*

Press Tab button to give space in place of

```
GNU nano 3.2 /etc/ppp/chap-secrets
# Secrets for authentication using CHAP
# client      server  secret          IP addresses
purevpn0s    [REDACTED] PPTP    [REDACTED]    *
```

Note:

Press CTRL+O to save the file, and CTRL+X to exit the editor.

5 Now to create a PureVPN profile, run the following command:

- **sudo nano/etc/ppp/peers/purevpn**

And enter the following information:

Note

nolaunchpppd and debug contains 2 "-" symbols.

pty "pptp usca.pointtoserver.com -nolaunchpppd -debug"

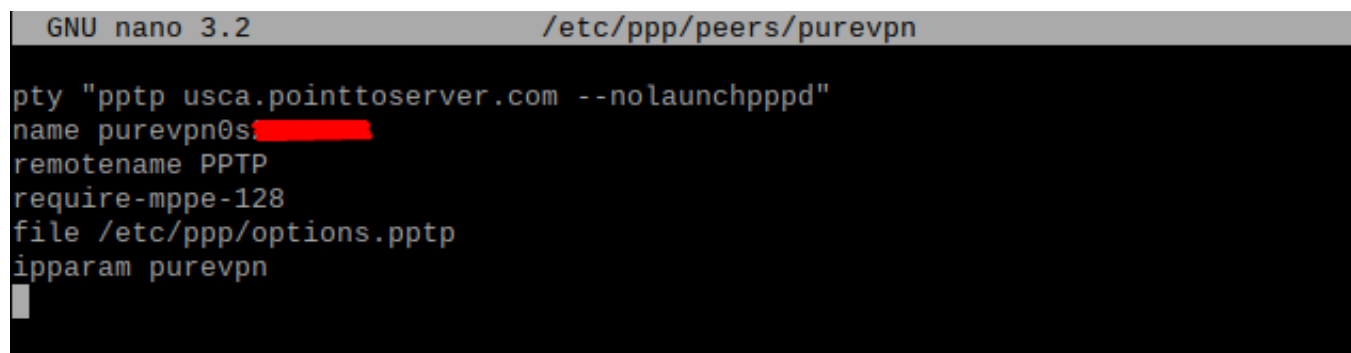
name yourVPNusername

remotename PPTP

require-mppe-128

file /etc/ppp/options.pptp

ipparam purevpn



```
GNU nano 3.2 /etc/ppp/peers/purevpn
pty "pptp usca.pointtoserver.com --nolaunchpppd"
name purevpn0s
remotename PPTP
require-mppe-128
file /etc/ppp/options.pptp
ipparam purevpn
```

Please note that we have currently used usca.pointtoserver.com as the server address. You can choose any server of your choice by [clicking here](#).

6 Now run the mentioned command and edit the file as following, make sure that there is no '#' sign before these lines:

- **sudo nano /etc/ppp/options.pptp**

lock

noauth

refuse-pap

refuse-eap

refuse-chap

nobsdcomp

nodeflate

require-mppe-128

```
GNU nano 3.2 /etc/ppp/options.pptp
# and the kernel MPPE module available from the CVS repository also on
# http://ppp.samba.org/, which is packaged for DKMS as kernel_ppp_mppe.
#####
# Lock the port
lock

# Authentication
# We don't need the tunnel server to authenticate itself
noauth

# We won't do PAP, EAP, CHAP, or MSCHAP, but we will accept MSCHAP-V2
# (you may need to remove these refusals if the server is not using MPPE)
refuse-pap
refuse-eap
refuse-chap
#refuse-mschap

# Compression
```

7 Now run the following command, add the mentioned line in it and save/exit the file.

- **sudo nano /etc/ppp/ip-up.local**
- `/sbin/route add default ppp0`

```
GNU nano 3.2 /etc/ppp/ip-up.local
/sbin/route add default ppp0
```

8 In order to execute the VPN profile, run the following command:

- **sudo chmod 755 /etc/ppp/ip-up.local**

9 In order to connect to the VPN, run the command:

- **sudo pppd call purevpn**

10 You are connected to **PureVPN** now, to check your current IP and location run the command:

curl ipinfo.io

```
root@raspberrypi:/home/pi# sudo pppd call purevpn
root@raspberrypi:/home/pi# curl ipinfo.io
{
  "ip": "172.94.69.47",
  "city": "San Francisco",
  "region": "California",
  "country": "US",
  "loc": "37.7509,-122.4153",
  "org": "AS46562 Total Server Solutions L.L.C.",
  "postal": "94110",
  "timezone": "America/Los_Angeles",
  "readme": "https://ipinfo.io/missingauth"
}root@raspberrypi:/home/pi# █
```

11 In order to disconnect to the VPN, run the command:

- **sudo pkill pppd**