

Connecting to a SoftEther VPN Server using the SoftEther client on Linux
#####

- Here is an article that we used to get the last few steps correct:
<https://www.cactusvpn.com/tutorials/how-to-set-up-softether-vpn-client-on-linux/>

*The CactusVPN tutorial sends ALL traffic through the VPN. We will enable **Split Tunneling***

- We are using the below variables as examples. Your setup will be different. Please see your Portal > VPN sections for your Server IP and credentials:
Server IP: 192.168.1.1
Username: user1
Password: friday

NOTE: The name of the tap interface will be prepended by "vpn_", so if you type in "tap0" in **vpncmd** it will create a virtual interface named "vpn_tap0".

- Ensure that ip_forward is enabled in the kernel:

```
$> echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
```

- Download the SoftEther **vpnclient**.
 - SoftEther's download website is here: <http://www.softether-download.com/en.aspx>
 - Here is a working link as of 12/27/2021:
https://github.com/SoftEtherVPN/SoftEtherVPN_Stable/releases/download/v4.38-9760-rtm/softether-vpnclient-v4.38-9760-rtm-2021.08.17-linux-x64-64bit.tar.gz
- Install the SoftEther **vpnclient**

```
$> wget  
https://github.com/SoftEtherVPN/SoftEtherVPN_Stable/releases/download/  
v4.38-9760-rtm/softether-vpnclient-v4.38-9760-rtm-2021.08.17-linux-  
x64-64bit.tar.gz  
$> tar -xvzf softether-vpnclient-v4.38-9760-rtm-2021.08.17-linux-x64-  
64bit.tar.gz  
$> cd vpnclient/  
$> make
```

* This tutorial assumes **vpncmd** and **vpnclient** are running from the source directory *

- Start the **vpnclient** using sudo:

```
$> sudo ./vpnclient start
```

- Test the **vpnclient** by connecting to the client demon:

```
$> ./vpncmd
```

- select '2' to connect to a client VPN instance
- Hit enter to use localhost as the address for the VPN client

```
VPN Client> check
```

- Create a new virtual network interface (called "**tap1**" in this tutorial):

```
VPN Client>NicCreate tap1
```

- Create an account (called "**newAccount**") using the server IP, username, and tap name:

```
VPN Client>accountcreate
AccountCreate command - Create New VPN Connection Setting
Name of VPN Connection Setting: newAccount
Destination VPN Server Host Name and Port Number: 192.168.1.1:4500
Destination Virtual Hub Name: DEFAULT
Connecting User Name: user1
Used Virtual Network Adapter Name: tap1
The command completed successfully.
VPN Client>
```

- Add a password for this vpn account:

```
VPN Client>AccountPasswordSet newAccount
* enter and confirm the password
* when asked about "standard" or "radius", choose "standard".
```

- Connect to the VPN server:

```
VPN Client>AccountConnect newAccount
```

- Check the log file to see if there are any errors:

```
$> vim client_log/*.log
```

- Run **dhclient** on the virtual interface

```
$> sudo dhclient vpn_tap1
```

- You may need to add a route to your Employees Subnet if **dhclient** doesn't

```
$> sudo ip route add 192.168.3.0/24 via 192.168.252.1 dev vpn_tap1
```

- The **vpncmd** commands can be ran from the command line:

```
$> ./vpncmd localhost /CLIENT /CMD accountconnect newAccount
```

If you are running Linux in a VM with Bridged Networking you may need to add a firewall rule for your adapter, vpn_tap1 is the example we used in this document.

```
sudo iptables --table nat --append POSTROUTING --out-interface vpn_tap1 -j MASQUERADE sudo
iptables --append FORWARD --in-interface vpn_tap1 -j ACCEPT
```