OpenVPN: Easy and Secure Setup Guide

Steven Roddis

2010-01-06

Intended Audience:

Everyone with basic computer knowledge: editing files, use of a SFTP client like FileZilla, transferring files, should be comfortable with Windows and have a little network experience.

What is a VPN

A VPN allows secure access to resources inside a firewall, it also allows you to secure all your traffic and make it pass out of a server you trust, preventing snooping on your traffic. This is good at a cafe where you don't trust (nor should you) the network.

Request for Comments:

This guide is a work in progress, if it was too much text or not enough let me know.
If you are having troubles **let me know** and I'll try help you out. :)
This will enable me to make revisions to this guide and make it more helpful.
Email me: vpnguide@stevenroddis.com

Why OpenVPN:

The easiest way to secure your net traffic would be to set up Hamachi and a proxy such as: FreeProxy from Hand-Crafted Software on a spare machine. However OpenVPN secures all traffic and will work in locations that block ports like cafe hotspots, universities, etc.

Things to remember when securing your traffic with a VPN server:

Protect the server from ARP spoofing, DHCP hijacks etc, use static ARP tables and configure DNS and gateway IPs. Don't worry this will be explained later.

Why use Linux:

Originally I was going to write this guide for Windows, however there was no [obvious] easy way to route the traffic between the OpenVPN adapter and NIC that connects to the gateway. (Don't worry if this doesn't make sense to you it doesn't need to) Therefore part of the guide is dedicated to getting **Ubuntu Server to run on Windows** using VMware. Don't worry **this guide** is written for people with **zero knowledge of Linux.**

Linux on Windows:

Download and install: http://www.vmware.com/go/tryworkstation older computers may need an older version of vmware. I couldn't find a previous version on their site so:
http://btjunkie.org/torrent/VMWare-Workstation-v6-5-3-185404-Incl-Keygen/40329ecc1a64e8c052dde486216feb58c621f67e3c28
http://www.rapidsharefilez.com/softwares/vmware-workstation-6-5-3-185404.html
Do NOT use the "keygen" contained within, they are illegal in Australia (as of writing). Purchase the software if you want to use it past the trial. Once you have a working virtual machine you can use the free VMWare Player (included) to run it **without purchasing** VMware workstation.

Download Ubuntu Server: http://www.ubuntu.com/getubuntu/download-server unless you know your server is 64 bit choose 32 bit. 32 bit will run on 64 bit but not vice-versa.

So now to install Ubuntu: Open VMWare workstation. File->New->Virtual Machine

Typical Install->Installer Disc Image (iso)->select the ubuntu image you downloaded.

Fill in the next screen, USE LOWERCASE for the username.
Then in the screen after that: name your virtual machine.
After pressing next, just leave the options as is (don't split).
Press next then **untick** "Power on this virtual machine after creation" and click finish.

Edit virtual machine settings->Network Adapter-> choose bridged and **tick** "replicate physical network connection state".

Press OK to close that window.

Now to fix the most common problem with bridging:
Edit->Virtual Network Editor
In the Automatic Bridging tab: Add all adapters to the "Excluded Adapters" list (click add) that are not your connection to the Internet.
These may include Hamachi etc.
(Don't close the window yet)

Then go to the Host Virtual Network Mapping for VMnet0 (default) which should say "Bridged to an automatically chosen adapter" select the adapter that is your connection to the the Internet.

Press OK to close that window.

Turn on your newly created virtual machine and it should install by itself.

Setting up on Ubuntu

Type these commands into console after login, type 'y' and press enter if it asks you for [y/n] for any of the below commands.
One command per line; commands through out this document are in italics:
To execute a command type the line and press enter.
sudo apt-get update
sudo apt-get upgrade
sudo apt-get dist-upgrade
sudo apt-get install ssh
sudo apt-get install openssl
sudo apt-get install openvpn

Change root password:
sudo passwd
(enter your password)

Now to switch to the root super-user
Use this command:
su
and type the password you just set for root.

Prevent ARP Spoofing and DHCP Hijacking

You can read up on what ARP Spoofing and DHCP Hijacking which are in the advanced stuff section at the end of this guide.
To do this we are going to set a static IP along with static gateway and DNS entries and add static arp entries.

Static IP and Gateway

sudo nano /etc/network/interfaces

If you are using DHCP for your primary network card which is usually eth0, you will see the following lines

auto eth0
iface eth0 inet dhcp

It's probably using DHCP right now, if not you can skip this step.

Change the above two lines to these seven.

auto eth0
iface eth0 inet static
address 192.168.2.22 #**replace** with your server's (the machine your are working on) IP
netmask 255.255.255.0
network 192.168.2.0 #If your gateway (router) IP is X.Y.Z.A **replace** this with X.Y.Z.0
broadcast 192.168.2.255 #If your gateway IP is X.Y.Z.A **replace** this with X.Y.Z.255
gateway 192.168.2.254 #**replace** with your gateway IP

Restart the networking service using the following command
Press CTRL-O and ENTER to save and CTRL-X to exit.

Static DNS

sudo nano /etc/resolv.conf

Edit this file to contain your dns servers:
Example:

```
nameserver 208.67.222.222
nameserver 208.67.220.220
```

Press CTRL-O and ENTER to save and CTRL-X to exit.

Static ARP

sudo nano /etc/rc.local

Add:
Example: (replace 192.168.2.254 with your gateway (router) IP and 02:00:00:00:00:00 with the MAC address of your gateway; to securely get your gateway's MAC address unplug all devices bar yours and run *arp -a* in command prompt (on your client pc))
arp -i eth0 -s 192.168.2.254 02:00:00:00:00:00
to /etc/rc.local, right before the "exit 0" line
Press CTRL-O and ENTER to save and CTRL-X to exit.

sudo /etc/init.d/networking restart

Now it is time for OpenVPN

Now running as root, see above for how to in Ubuntu.

nano /etc/default/openvpn

Then make sure each line is blank or has '#' (no quotes) at the start of it. Then type AUTOSTART="openvpn" at the end of file (press the down arrow to scroll down). Press CTRL-O and ENTER to save and CTRL-X to exit.

This tells OpenVPN which configuration file to use by default, it just makes things easier. Configuration files are in /etc/openvpn so the above means to look at /etc/openvpn/ openvpn.conf (don't worry it doesn't exist yet).

Now we want to copy some files from the help section so we can generate the keys.
cp -r /usr/share/doc/openvpn/examples/easy-rsa/ /etc/openvpn/

Now we need to edit a file which will make it easier to create certificates without having to retype stuff.
cd /etc/openvpn/easy-rsa/2.0/
nano vars

Now edit the file, to your tastes; use an email like vpnadmin@example.com not your primary email.

Change these lines:
export KEY_COUNTRY="US"
export KEY_PROVINCE="CA"
export KEY_CITY="SanFrancisco"
export KEY_ORG="Fort-Funston"
export KEY_EMAIL="me@myhost.domain"

-to what is correct for you.

Press CTRL-O and ENTER to save and CTRL-X to exit.

Run it:
*. ./vars (***Remember that is:** dot-space-dot-slash-v-a-r-s-[ENTER])

And now:
./clean-all

(to start fresh)

Notes for the next two commands:
Just press enter to use the values in the brackets ([value]), so when it says for example ["AU"] just press enter, don't type "AU" (no quotes) again.
Just press enter (don't type a challenge password) when it asks you for a challenge password.
And also just press enter for "An optional company name"

For Department just use "ITSec" (no quotes)
at the end answer "y" (no quotes) to each of the two questions.

Here are the two commands:
./build-ca
./build-key-server server

Generate the keys for the clients
./build-key client1

Again,
For Department just use "ITSec" (no quotes)
at the end answer "y" (no quotes) to each of the two questions.
If you want more than one client replace client1 with client2 (for the second client) and repeat.

Finish up:
./build-dh

Now we need to generate the TLS-Auth (For more info: http://www.openvpn.net/index.php/open-source/documentation/howto.html#security) key, which adds more security to our installation.

openvpn --genkey --secret /etc/openvpn/ta.key
Problem? Did you type dash (-) insted of dashdash (--) for --genkey and/or --secret

Get The Keys Off The Server

On your windows client download and install FileZilla (http://filezilla-project.org)
Create a new site for the server and choose server type "SFTP - SSH File Transfer Protocol".

User: root
Password: <what you set>

See here for verifying the fingerprint: http://cafe.elharo.com/security/verifying-ssh-host-fingerprints

Navigate to /etc/openvpn/easy-rsa/2.0/keys/
and copy
ca.crt
clientX.crt
clientX.key

Also from /etc/openvpn/
grab:
ta.key
for all clients

to each of the clients over a secure channel.

A Touch of NAT

You are almost there, just a few more commands on the server side
nano /etc/sysctl.conf
Change (uncomment):
#net.ipv4.ip_forward=1
to
net.ipv4.ip_forward=1
CTRL-O, enter for save and then CTRL-X for exit.

Two more commands:
echo 1 > /proc/sys/net/ipv4/ip_forward

In the next command you don't need to change the IP see the notes below to learn more.

iptables -t nat -A POSTROUTING -s 10.3.0.0/24 -o eth0 -j MASQUERADE

Note: You don't need to change the IP, it is what is allocated to the client, it just needs to be DIFFERENT from your normal internal IP, if this doesn't make sense don't worry it should work without change.

To make it permanent you need to:

sudo nano /etc/rc.local

Add the line:
iptables -t nat -A POSTROUTING -s 10.3.0.0/24 -o eth0 -j MASQUERADE
just before: exit 0

CTRL-O, enter for save and then CTRL-X for exit.

Server Config

On the server (this next file is blank (non-existent) by default):
nano /etc/openvpn/openvpn.conf

and type in:

```
dev tun
proto udp
port 443
ca /etc/openvpn/easy-rsa/2.0/keys/ca.crt
cert /etc/openvpn/easy-rsa/2.0/keys/server.crt
key /etc/openvpn/easy-rsa/2.0/keys/server.key
dh /etc/openvpn/easy-rsa/2.0/keys/dh1024.pem
tls-auth ta.key 0
cipher AES-256-CBC
user nobody
group nogroup
server 10.3.0.0 255.255.255.0 #If you changed it above replace 10.3.0.0
to suit.
persist-key
persist-tun
cipher AES-256-CBC
client-to-client
push "dhcp-option DNS 208.67.222.222" #Replace with your #1 DNS
Server
push "dhcp-option DNS 208.67.220.220" #Replace with your #2 DNS
Server
```

CTRL-O, enter for save and then CTRL-X for exit.

And finally start OpenVPN
/etc/init.d/openvpn start

Forward Port 443 (UDP)

See http://portforward.com for help with forwarding ports, you need to forward port 443 (UDP).
Common mistake: Make sure you choose UDP not TCP.

Install The Client

Download and install the "Windows Installer" from http://openvpn.net/index.php/open-source/downloads.html on to your client(s)

Client Config

On the client...
Windows XP:
C:\Program Files\OpenVPN\config

Windows Vista and above:
C:\Program Files (x86)\OpenVPN\config

In the above directory:
Install the certificate (and all those files previously mentioned)
ca.crt
clientX.crt
clientX.key
ta.key

And make two text files below using, say, notepad:

external.ovpn

```
dev tun
client
proto udp
remote 8.8.8.8 443 #Replace with your external IP or hostname
resolv-retry infinite
nobind
user nobody
group nogroup
redirect-gateway def1
persist-key
persist-tun
ca ca.crt
cert client1.crt
key client1.key
tls-auth ta.key 1
cipher AES-256-CBC
remote-cert-tls server
verb 3
```

internal.ovpn

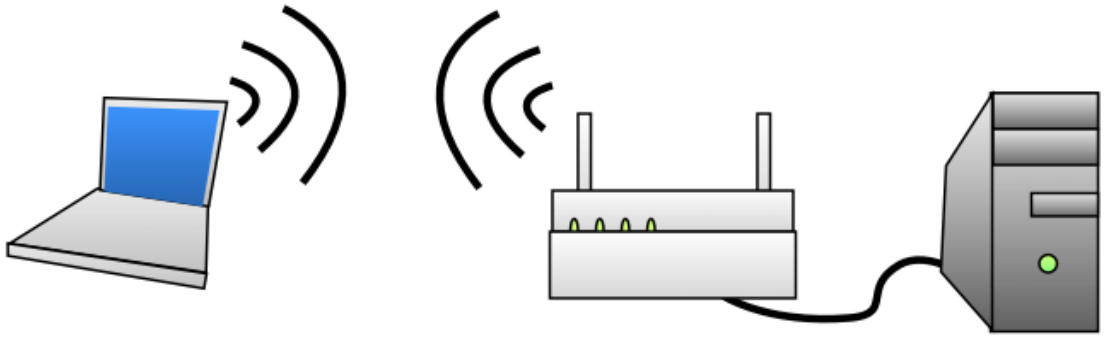
```
dev tun
client
proto udp
remote 192.168.22.6 443 #Replace with your OpenVPN server's IP
resolv-retry infinite
nobind
user nobody
group nogroup
redirect-gateway local def1
persist-key
persist-tun
ca ca.crt
cert client1.crt
key client1.key
tls-auth ta.key 1
cipher AES-256-CBC
```

```
remote-cert-tls server
verb 3
```

Using the client

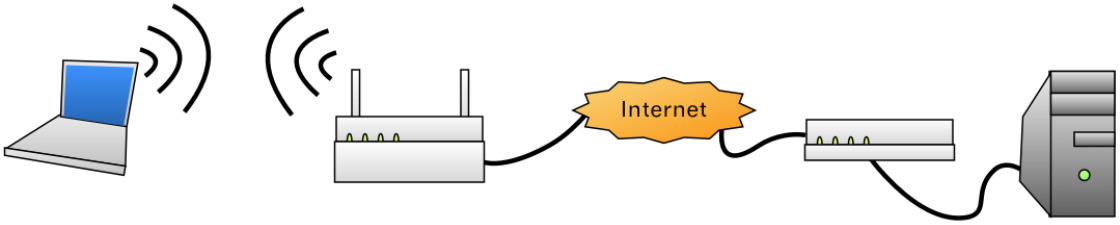
Now you need to have a look at how you are going to use your VPN.

Internal:



For example you are securing your wireless connection which means the VPN server and the client (eg. laptop) are on the same network. You do not have to use the VPN on the same network, this may be useful if you are on an insecure wireless network at home.

External:

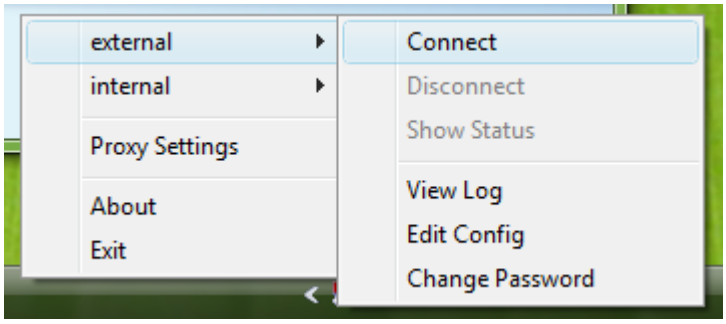


Or for example are you connecting your laptop via WiFi at a cafe which means the VPN server and the client (eg. laptop) are NOT on the same network.

Connect

> Note that on Windows Vista and later (with UAC enabled), you will need to run the OpenVPN GUI with administrator privileges. You can do this by right-clicking on the OpenVPN GUI desktop icon, and selecting "Run as administrator".

Double click on the OpenVPN GUI icon (should be on your desktop).



Tips

Start, Stop and Restart

Start OpenVPN:
/etc/init.d/openvpn start

Stop OpenVPN:
/etc/init.d/openvpn stop

Restart OpenVPN:
/etc/init.d/openvpn restart

Dynamic DNS

If you have a dynamic external IP you can use a free service like: https://www.dyndns.com which lets you use a hostname in place of your IP that changes. The client automatically updates the hostname with your new IP.

Add New Clients

Just like before in the section "Generate the keys for the clients".
./build-key clientX
And transfer the configs like in: "Client Config".

Advanced Stuff

Unless your interested you don't need to read this section.

ARP Spoofing and DHCP Hijacking

Both ARP Spoofing and DHCP Hijacking allow a malicious person to intercept your traffic.

ARP Spoofing:
http://en.wikipedia.org/wiki/ARP_spoofing

DHCP Hijacking:
Involves sending DHCP responses from a malicious person that makes them the gateway, all your outbound traffic goes to them.

Why You Shouldn't Use TCP For Your VPN

Sometimes you cannot avoid tunnelling over TCP, but if you can avoid it, please DO.
Have a look at this page for more info: http://sites.inka.de/~bigred/devel/tcp-tcp.html

Static Keys

They seem simple, and are much faster to generate than a public/private keypair however you don't get the added security benefits of Perfect Forward Secrecy and more

importantly TLS-Auth that adds another layer of security to your OpenVPN server.

Thanks

Aaron for reviewing my guide and making sure it was readable to the novice user.
Ben for converting my sketches into nice colourful diagrams.
Patricia for reviewing my guide for grammar and spelling.