

## Forum:[Hack Forums](#)

Thread: [How to secure an OpenVPN Server](#)

**Cyph Jr** - December 14, 2014, 3:29 pm

This tutorial will be based on securing your OpenVPN from attacks. You can use OpenVPN using routing or bridging. If you know what the difference is, you don't need my help choosing. For this tutorial, we'll use routing. We'll also use OpenVPN's default network range, which is 10.8.0.0/24. Unless you already use this network range somewhere, this will be fine. If you do need a different range, pick a private range and make sure you adjust all the later configuration steps accordingly.

We need to configure routing and setup a firewall as well. I use a tool called firehol to configure iptables, which makes it very simple to set up a complex firewall. So, type the following commands.

PHP Code

```
<?php
sudo apt-get install firehol
```

This will install firehol which will make it easier to configure your iptables. Now that we've installed firehol, we want to open up our fire-wall configuration file with firehol.

Now of course, nano is optional. You can use your favorite text editor, I personally use it because it's pre-installed on nearly every GNU/Linux system.

PHP Code

```
<?php
nano /etc/firehol/firehol.conf
```

While we could allow incoming OpenVPN connections from any address, we're going to limit these connections to the IP addresses of the computers you want to connect. This way people whom purchase/decide to use your VPN can only access your VPN. This will limit people from giving out your VPN to others whom may have not been selected to use it.

PHP Code

```
<?php
version 5
interface eth0 inet
client all accept           // allow all outgoing connections
server ssh accept           // allow all incoming SSH connections
server openvpn accept src "1.2.3.4 2.3.4.5" // allow incoming OpenVPN connections
interface tun0 vpn
server all accept           // allow all incoming connections on the VPN
client all accept           // allow all outgoing connections on the
router inet2vpn inface eth0 outface tun0
route all accept            // route freely to the VPN
router vpn2inet inface tun0 outface eth0
masquerade                  // use NAT masquerading from the VPN
route all accept            // route freely to the VPN
```

You're going to want to edit the part of that configuration that is quoted with an IP address, there you will specify the IP address's that you wish to be able to connect to your OpenVPN server. Now

we're going to want to start our fire-wall to ensure that our fire-wall is up and secure. To do so, we're going to want to execute the following commands

```
PHP Code
<?php
firehol start
```

Please note that if you restart your server, the firehol configuration won't be saved. So if you messed up/poorly configured, just restart your server. To make the firewall permanent, input the following.

```
PHP Code
<?php
nano /etc/default/firehol
```

We're going to see a configuration file, we're going to want to find the line that says 'START\_FIREHOL=NO' We're going to want to change 'NO' to 'YES' and then save the configuration file. Once we've done that, exit with Control-O, Enter, Control-X. Once you are logged in to the server as root we can begin editing the sshd\_config file. As with any modification to an important server file, it is always a good idea to back it up before making any changes.

```
PHP Code
<?php
cp /etc/ssh/sshd_config /etc/ssh/sshd_config_backup
```

Now we are ready to modify the file. Open /etc/ssh/sshd\_config in your favorite text editor (for this example we will use Vim) and locate the line that specifies the port. If your sshd\_config file has not been altered the Port setting will be commented out with a # symbol (example below).

```
PHP Code
<?php
nano /etc/ssh/sshd_config
```

The # symbol tells the server to ignore anything after it on the same line, so we will need to remove that character and then change the number 22 to something else. Be sure you do not pick a port number that is in use by a different service. If you are unsure, Wikipedia has a list of port numbers and their uses. Try to use a port number that is not listed. For this example, we will use Port 2222 (which is used by Windows servers but not by default on Linux servers). Save your changes to the port you specified and success. Your OpenVPN server is now completely secure.

**Arzel** - December 14, 2014, 3:39 pm  
cool tutorial cyph nice job nice ?

**Tirus** - December 14, 2014, 6:52 pm  
Sad to see such a small response on such a hard effort ed thread you've made, Cyph. Keep up the good work buddy.

**Cyph Jr** - December 14, 2014, 7:57 pm  
(Dec 14, 2014 07:52 PM)Tirus Wrote: [\[CITE\]](#) Sad to see such a small response on such a hard effort ed thread you've made, Cyph. Keep up the good work buddy.  
At least someone see's my value in this.

**Cyph Jr** - December 16, 2014, 10:11 pm

Thanks for your feedback everyone, looking for ideas on a next guide.

**uid=2444325** - *December 16, 2014, 10:12 pm*

Bumping this. Looks like you worked hard on the tutorial man. People need to see this ;)

**Bucked** - *December 17, 2014, 8:01 am*

Really helpful post! Thanks for the share (:

**guycheap** - *December 23, 2014, 11:16 pm*

Good securing techniques man. I use some of these already with my clients :).

**Cyph Jr** - *December 24, 2014, 7:08 am*

(Dec 24, 2014 12:16 AM)guycheap Wrote: [\[CITE\]](#) Good securing techniques man. I use some of these already with my clients :).

Thanks for your feedback. Any tips on how I can improve it?