SSH Tunnel in 30 Seconds (Mac OSX & Linux)

Some days, I wonder why VPN's are really necessary when we can just use an SSH tunnel.
If you're on Mac or a flavour of Linux, this SSH tunnelling tutorial is for you.

> *"A secure shell (SSH) tunnel consists of an encrypted tunnel created through a SSHprotocol connection. Users may set up SSH tunnels to transfer unencrypted traffic over a network through an encrypted channel." – Wikipedia*

Launch an SSH tunnel

To initiate your SSH tunnel, simply open Terminal and connect to your remote server via SSH with the following flags:

```
ssh -D 8080 -C -N username@example.com
```

This will launch our SSH tunnel on port 8080 and route all traffic (securely) through the server at example.com.

Browse the Web with Your SSH Tunnel (Chrome)

Now, let's start browsing the web using our new SSH tunnel.

Mac OSX:

1. Open Google Chrome
2. Select 'Chrome' up the top left
3. Select 'Preferences'
4. Select 'Show advanced settings…'

5. Select ‘Change proxy settings…’
6. Select ‘SOCKS Proxy’
7. Enter ’127.0.0.1’
8. Enter port ’8080’
9. Save changes by selecting ‘OK’

Fedora Linux:

1. Open Google Chrome
2. Select the wrench icon on the top right
3. Select ‘Settings’
4. Select ‘Show advanced settings…’
5. Select ‘Change proxy settings…’
6. Select ‘SOCKS Proxy’
7. Enter ’127.0.0.1’
8. Enter port ’8080’
9. Save changes by selecting ‘OK’

Search Google for ‘my ip’ and take a look at what your IP address is now. Cool right?

Exiting the SSH Tunnel

To exit the SSH tunnel, simply disable the SOCKS proxy within your browser.

Hope this helps, let me know if you have any suggestions in the comments below!

This entry was posted in How To, Linux and tagged encrypted, how-to, linux, ssh, tunnel on October 3, 2012 [http://drewsymo.com/how-to/ssh-tunneling-how-to-with-examples/] .