

alternatives to expensive programs

## Encrypting web traffic:

The following guide will show you how to use ssl tunnelling to encrypt your web traffic. Unsecured network are very common and people are using them on daily basis. The problem with those kind of network is that everyone can see just about everything you are doing on the network. This is also true for corporate networks where everything you do on the internet may get scrutinized by the IT department. I have put the following guide together using bits and pieces of information found on the internet.



### Requirements:

What you will need.

- 2 computers, 1 server your ubuntu computer and 1 client your windows computer.
- Ubuntu ( at home ) acting as your ssl server ([Download here](#))
- Windows ( used to connect to your server )
- Putty ( windows program to set up the tunnelling )
- Firefox ( web browser )

alternatives to expensive programs

### Step 1 ( setting up your server )



This step is crucial and Ubuntu is the easiest way I have found to perform this. You can do it with a windows operating system using cygwin, but it is much more complicated. Install Ubuntu on an old computer, or create dual boot with your windows operating system. When installing Ubuntu it will detect windows and ask you if you want to create a dual boot. ( Make sure you backed up your important documents before you do this just in case something goes wrong )

Once Ubuntu is installed, install ssh  
open a terminal window and type `apt-get install ssh` ( this will install the package )

To test that it's working, you can try to ssh into your own machine:

```
> ssh localhost
```

```
The authenticity of host 'localhost (127.0.0.1)' can't be established.  
RSA key fingerprint is 98:8a:b8:b2:9e:8a:84:e0:d4:08:27:fb:74:f0:de:d4.  
Are you sure you want to continue connecting (yes/no)?
```

Looks like it's working! Naturally our ssh client doesn't have the key for the server, since we just installed it. You can type yes to continue or just hit Ctrl-C to stop.

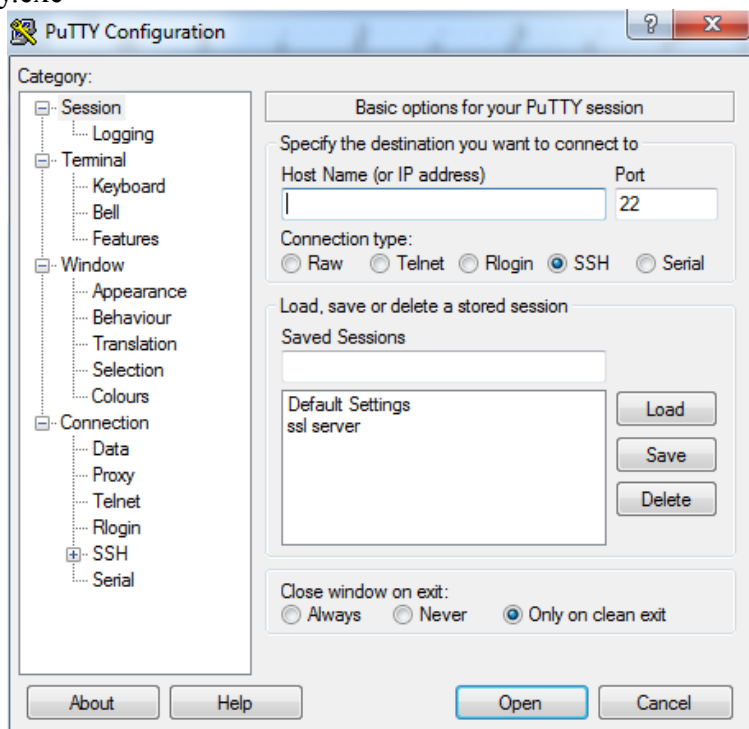
alternatives to expensive programs

**Step 2 ( installing putty )**



Download putty ([Download Here](#))

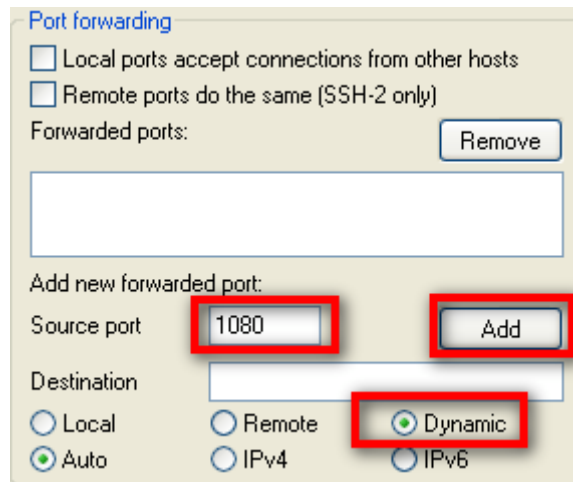
- click on putty.exe



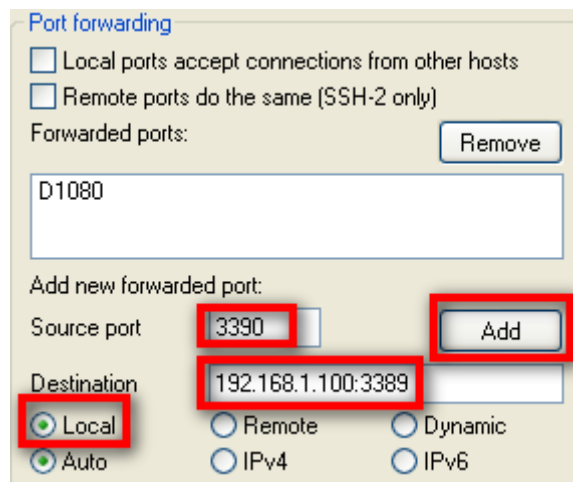
- You will need to enter the host name ( ip address of your ubuntu server, in ubuntu start a terminal window and type iwconfig )

alternatives to expensive programs

- click on the ssh / tunnel menu and add 1080 and pick dynamic. Press the add button.



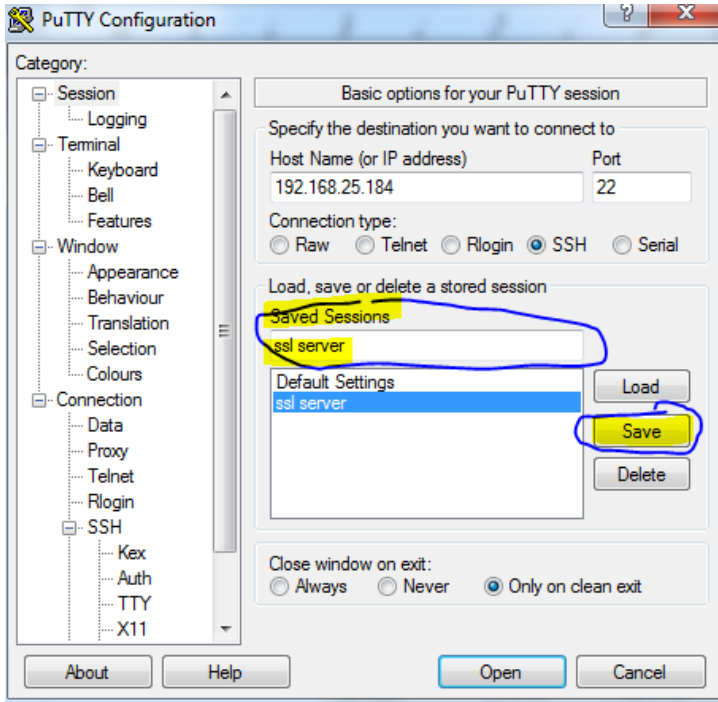
- also if you want to use remote desktop add the following ports



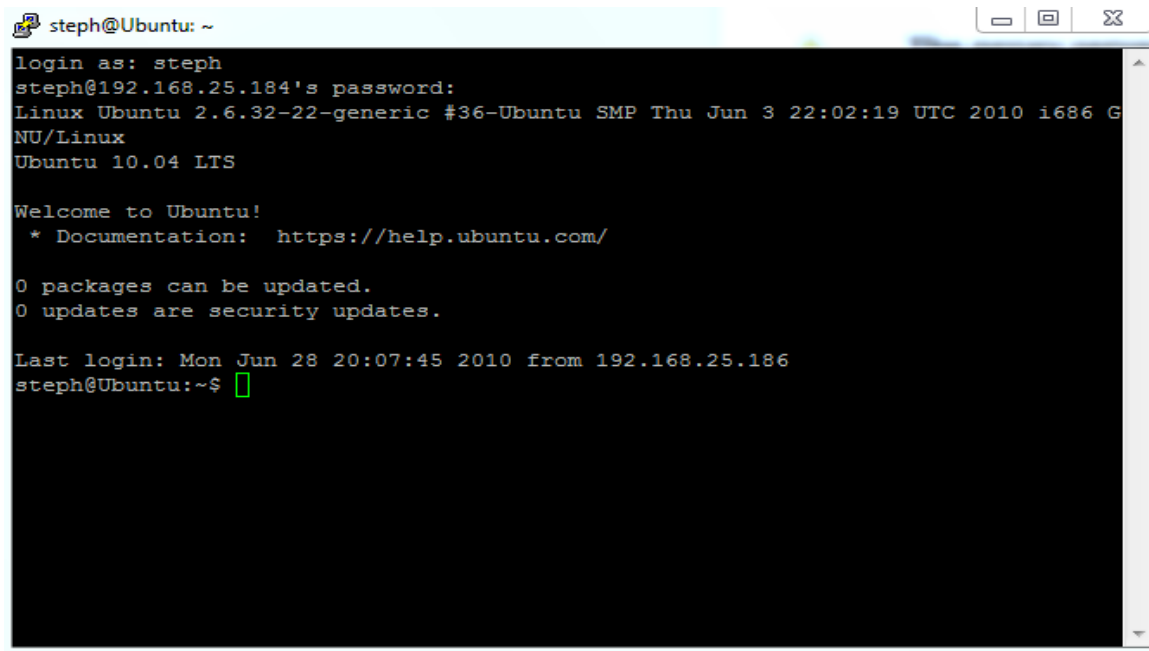
*(The destination being the computer ip address used to connect to your ubuntu server.)*

alternatives to expensive programs

- Now save your session by giving it a name.



- Click open and put your ubuntu user name and password. You are now connected via ssl and your traffic is being encrypted.

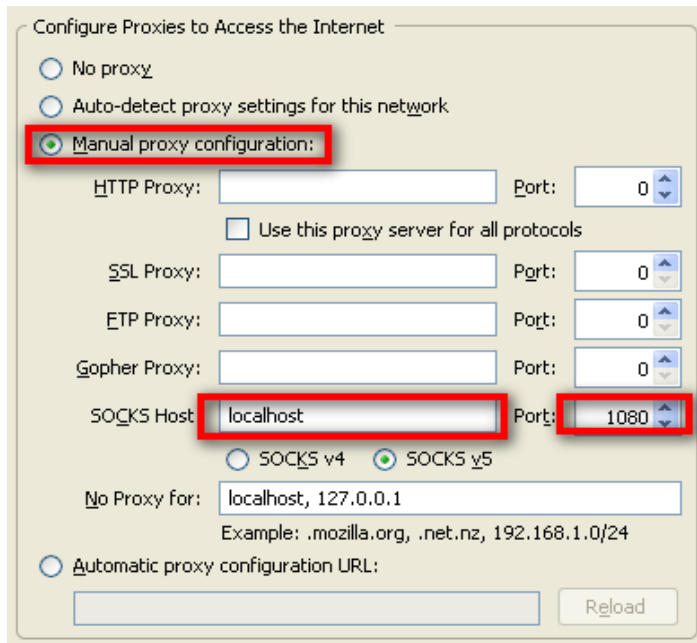


alternatives to expensive programs

### Step 3 ( setting up firefox as a sock proxy )

*(This will allow you to browse the internet while your connection is encrypted.)*

To set up Firefox to use the dynamic tunnel as a SOCKS proxy, go to the Firefox **Options** > **Advanced** > **Network** and click the **Settings...** button. In the settings page, click **Manual proxy configuration**, enter localhost for the **SOCKS Host**, and 1080 for the **Port**.



The image shows the 'Configure Proxies to Access the Internet' dialog box in Firefox. The 'Manual proxy configuration' option is selected and highlighted with a red box. Below it, the 'SOCKS Host' field is set to 'localhost' and the 'Port' is set to '1080', both highlighted with red boxes. The 'SOCKS v5' option is selected. Other proxy options like HTTP, SSL, FTP, and Gopher are set to port 0. The 'No Proxy for' field contains 'localhost, 127.0.0.1'. A 'Reload' button is visible at the bottom right.

alternatives to expensive programs

### Step 4 ( port forwarding )

You will have to open ports on your router in order to make this to work. Here's an example of my settings in the router:

**PORT FORWARDING**

This option is used to open multiple ports or a range of ports in your router and redirect data through those ports to a single PC on your network. This feature allows you to enter ports in various formats including, Port Ranges (100-150), Individual Ports (80, 68, 888), or Mixed (1020-5000, 689). This option is only applicable to the INTERNET session.

**24 -- PORT FORWARDING RULES**

			<b>Ports to Open</b>	
<input checked="" type="checkbox"/>	Name <input type="text" value="putty"/>	<input type="button" value="&lt;&lt;"/> <input type="text" value="Application Name"/>	TCP <input type="text" value="5150"/>	Schedule <input type="text" value="Always"/>
	IP Address <input type="text" value="192.168.25.186"/>	<input type="button" value="&lt;&lt;"/> <input type="text" value="Computer Name"/>	UDP <input type="text" value=""/>	Inbound Filter <input type="text" value="Allow All"/>
<input type="checkbox"/>	Name <input type="text" value="ssl"/>	<input type="button" value="&lt;&lt;"/> <input type="text" value="Application Name"/>	TCP <input type="text" value="3389"/>	Schedule <input type="text" value="Always"/>
	IP Address <input type="text" value="192.168.25.186"/>	<input type="button" value="&lt;&lt;"/> <input type="text" value="Computer Name"/>	UDP <input type="text" value=""/>	Inbound Filter <input type="text" value="Allow All"/>
<input type="checkbox"/>	Name <input type="text" value="ssl2"/>	<input type="button" value="&lt;&lt;"/> <input type="text" value="Application Name"/>	TCP <input type="text" value="3390"/>	Schedule <input type="text" value="Always"/>
	IP Address <input type="text" value="192.168.25.186"/>	<input type="button" value="&lt;&lt;"/> <input type="text" value="Computer Name"/>	UDP <input type="text" value=""/>	Inbound Filter <input type="text" value="Allow All"/>

Now the final test open your firefox browser and you should have connectivity. Now all your traffic is being encrypted. This does not mean that your are safe from everything..... good practices makes you secure not hardware or software.