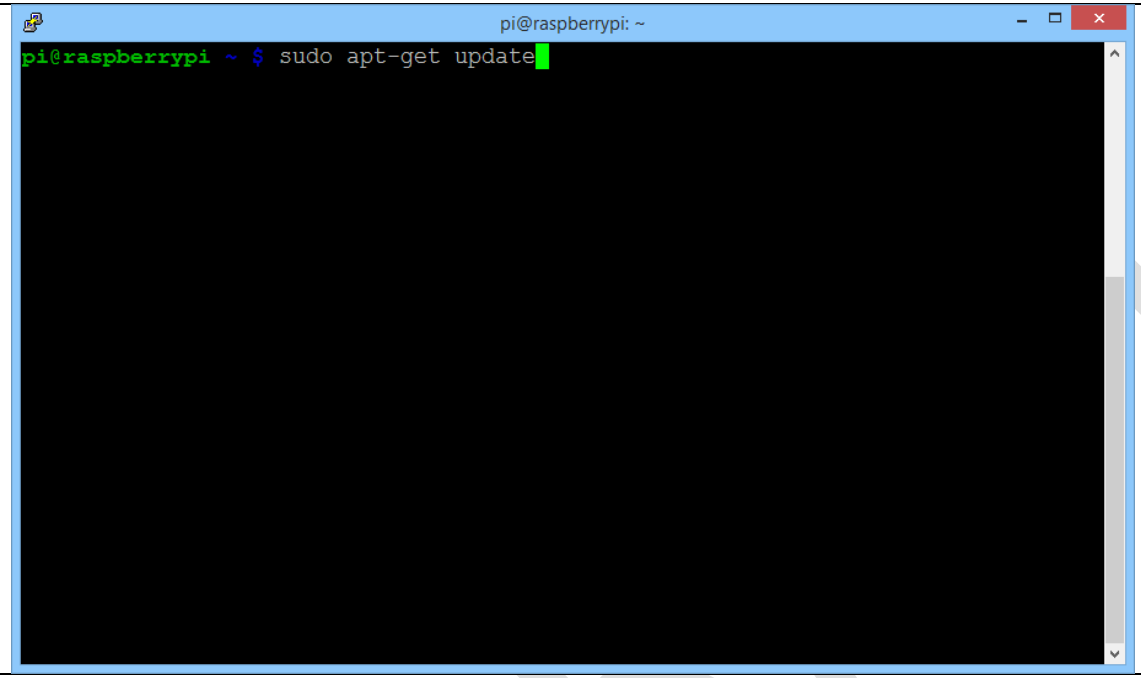
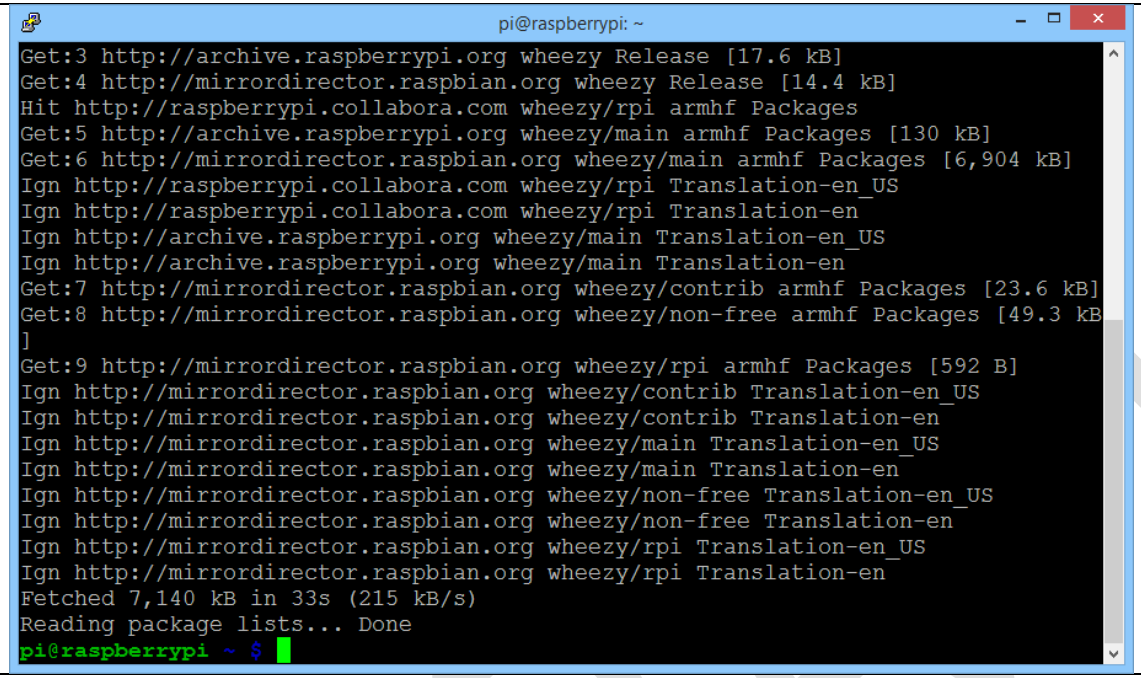
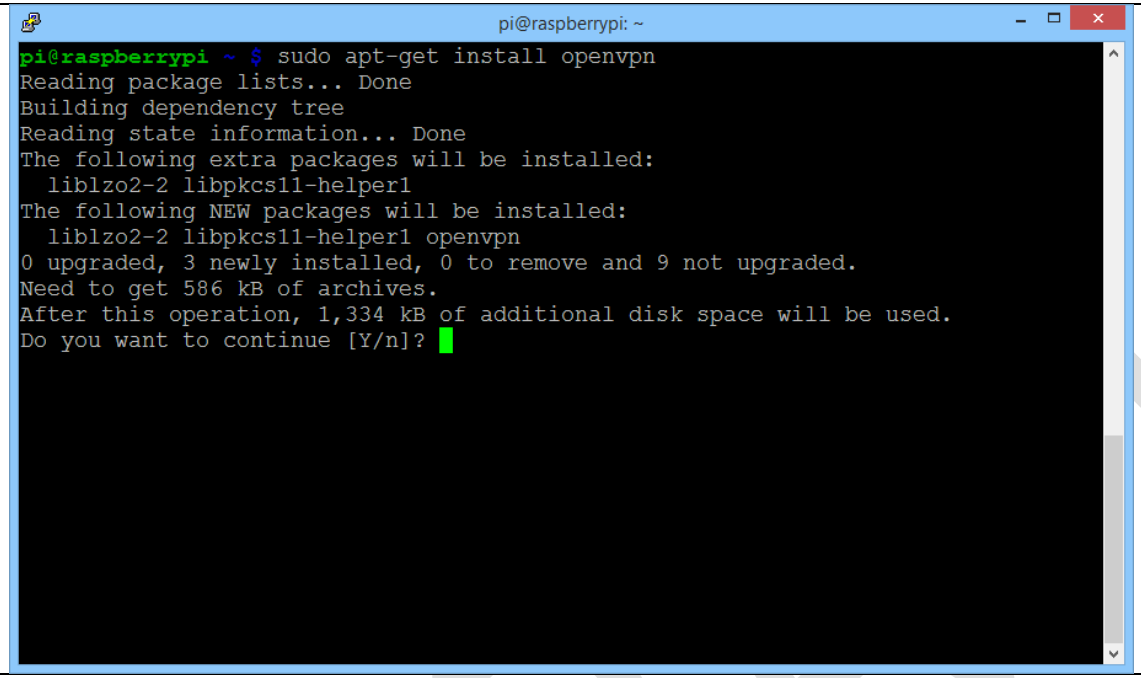
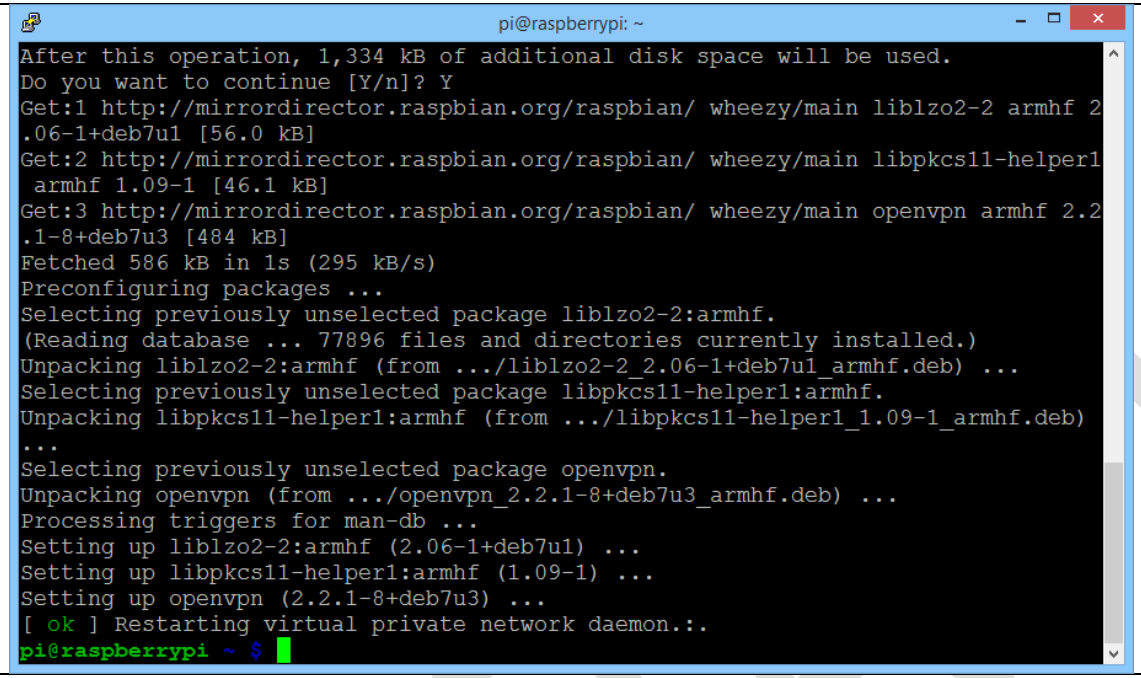
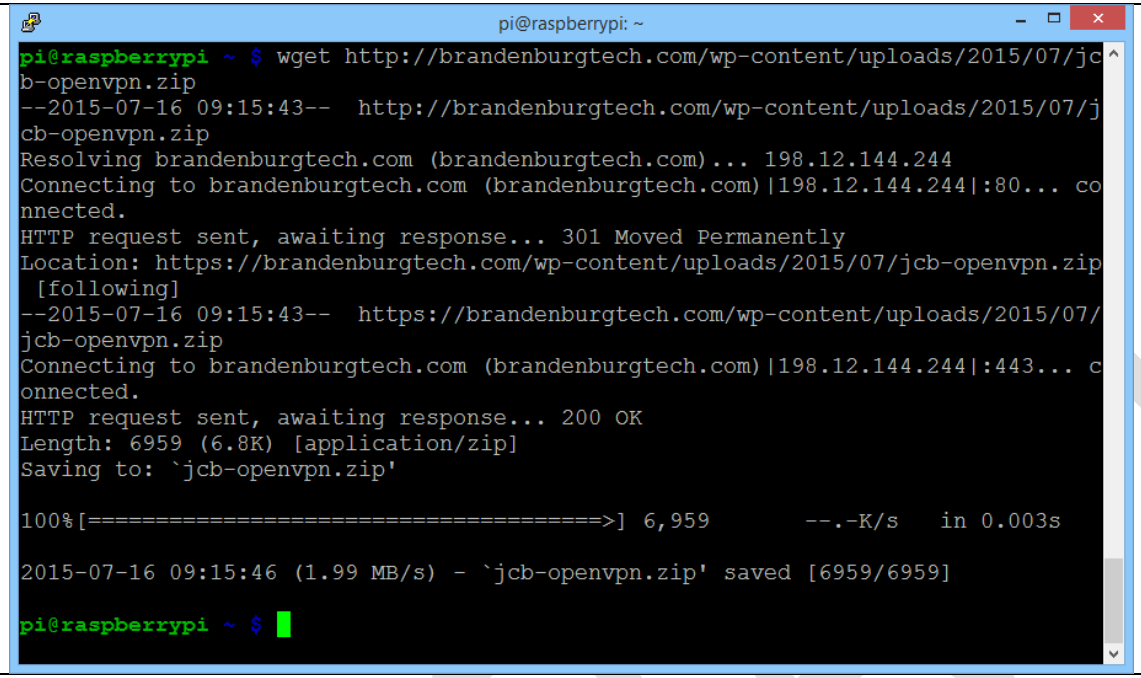
Configure Raspberry Pi as a VPN Tunnel

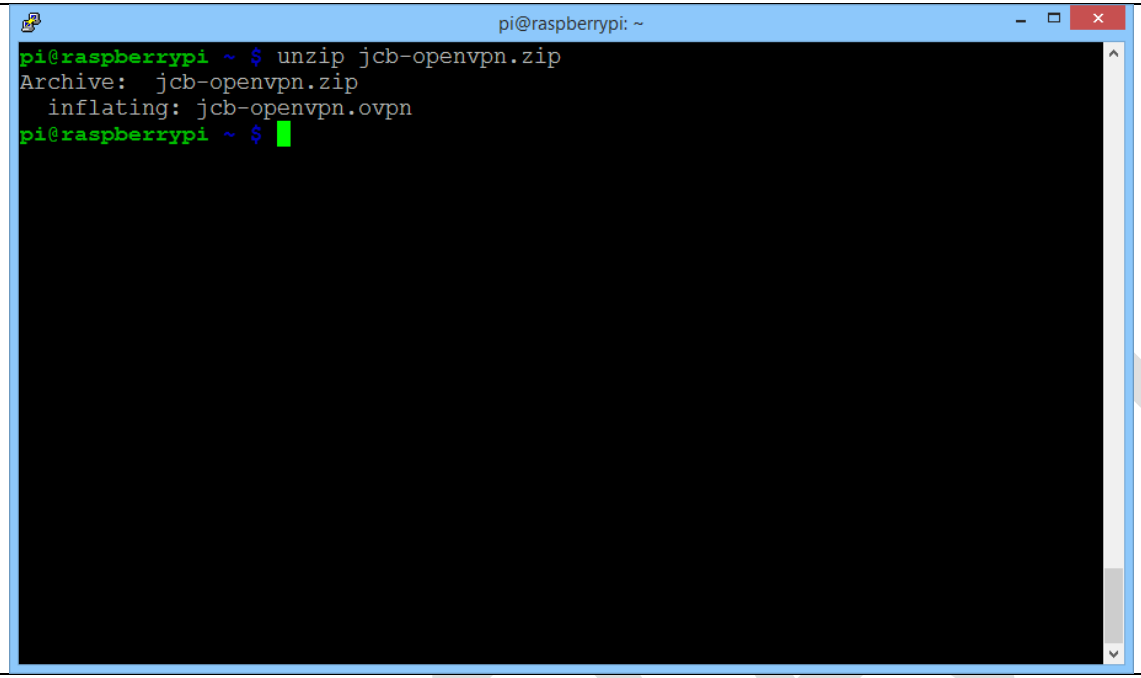
1		Ensure the Raspberry Pi has the latest catalog of updates. Execute “sudo apt-get update”.

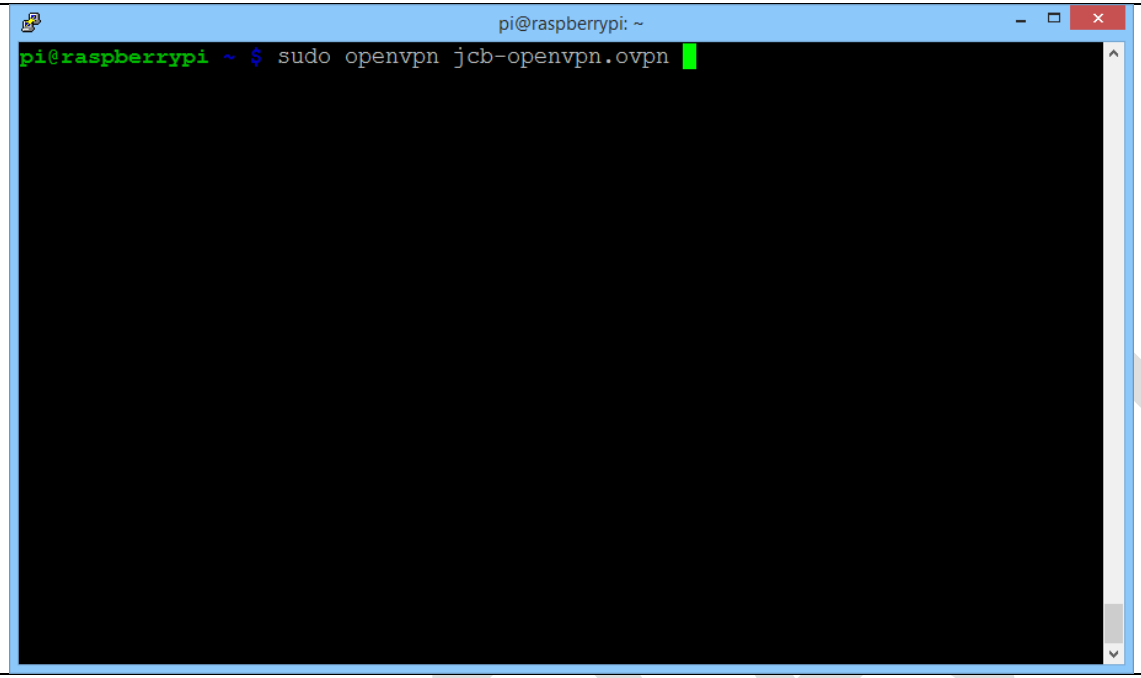
2		The results should look similar to the results in the screenshot.

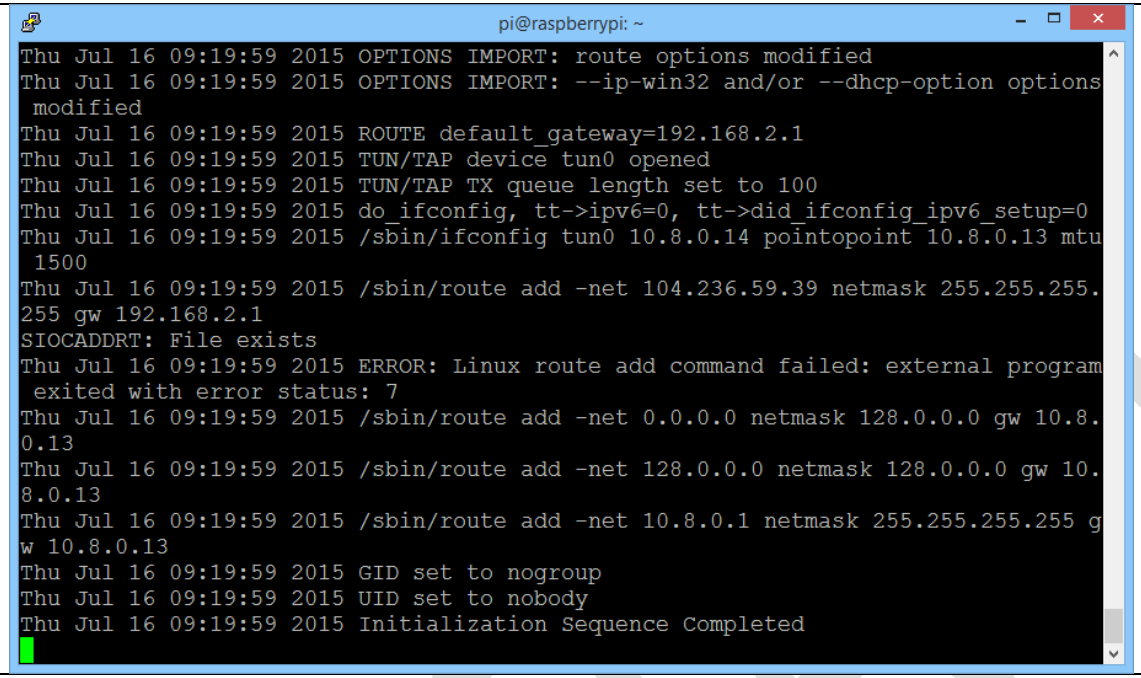
3		Install OpenVPN. Execute “sudo apt-get install openvpn”. When prompted, enter ‘Y’ to continue the installation of OpenVPN.

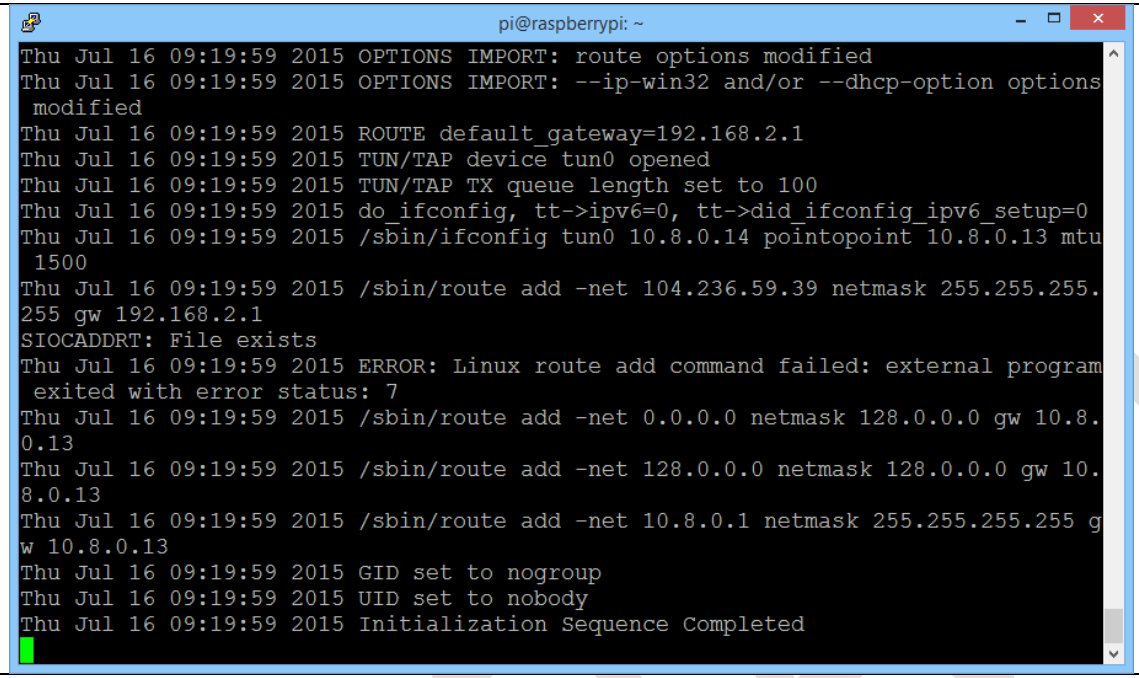
4		The results should look similar to the results in the screenshot.

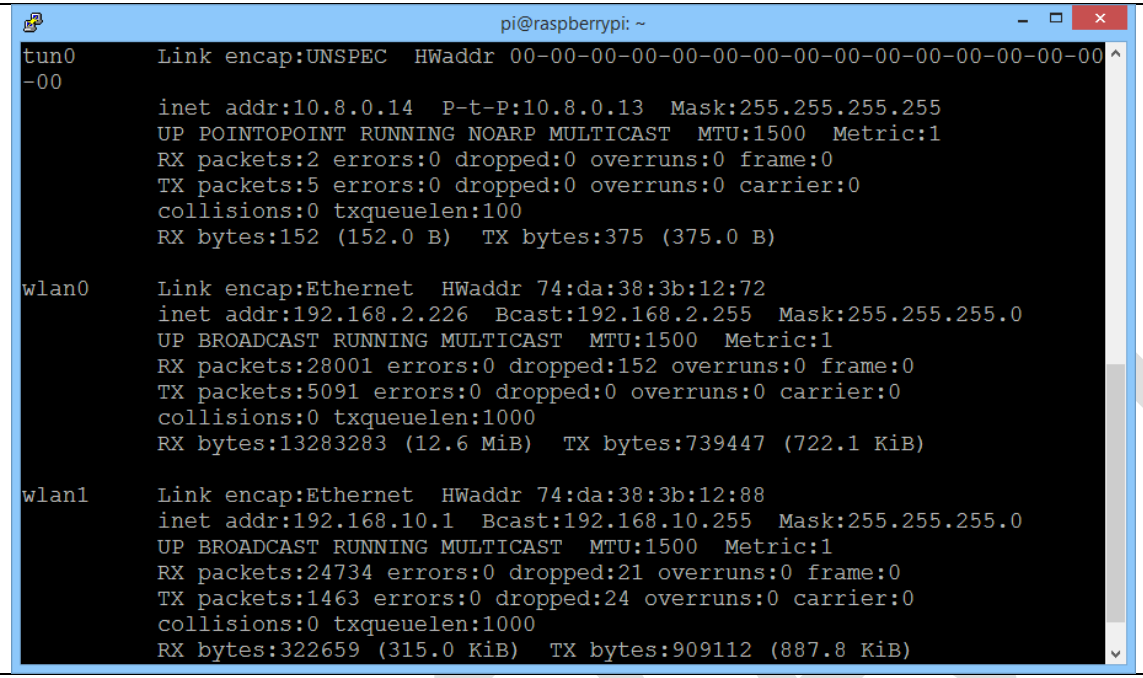
5		Retrieve the OpenVPN client profile for a sample OpenVPN server configured for this class. Execute “wget http://brandenburgtech.com/wp-content/uploads/2015/07/jcb-openvpn.zip”

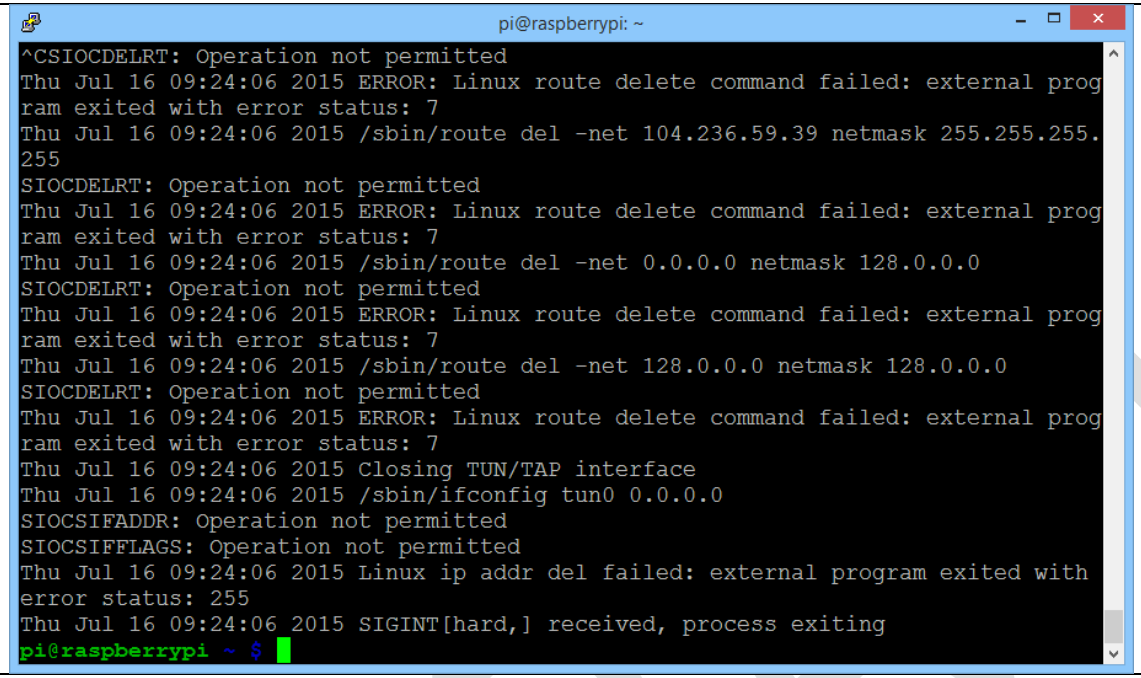
6		Extract the OpenVPN client profile from the zip file. Execute “unzip jcb –openvpn.zip”

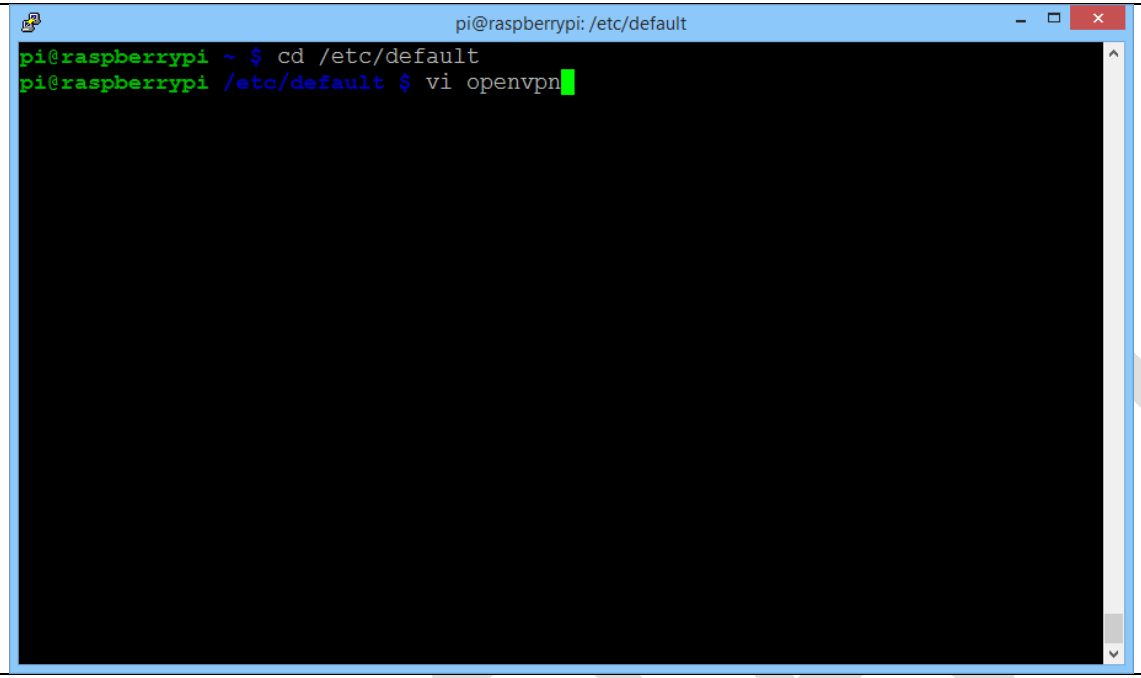
7		Test the OpenVPN configuration by forming a tunnel to the test server. Execute “sudo openvpn jcb-openvpn.ovpn”

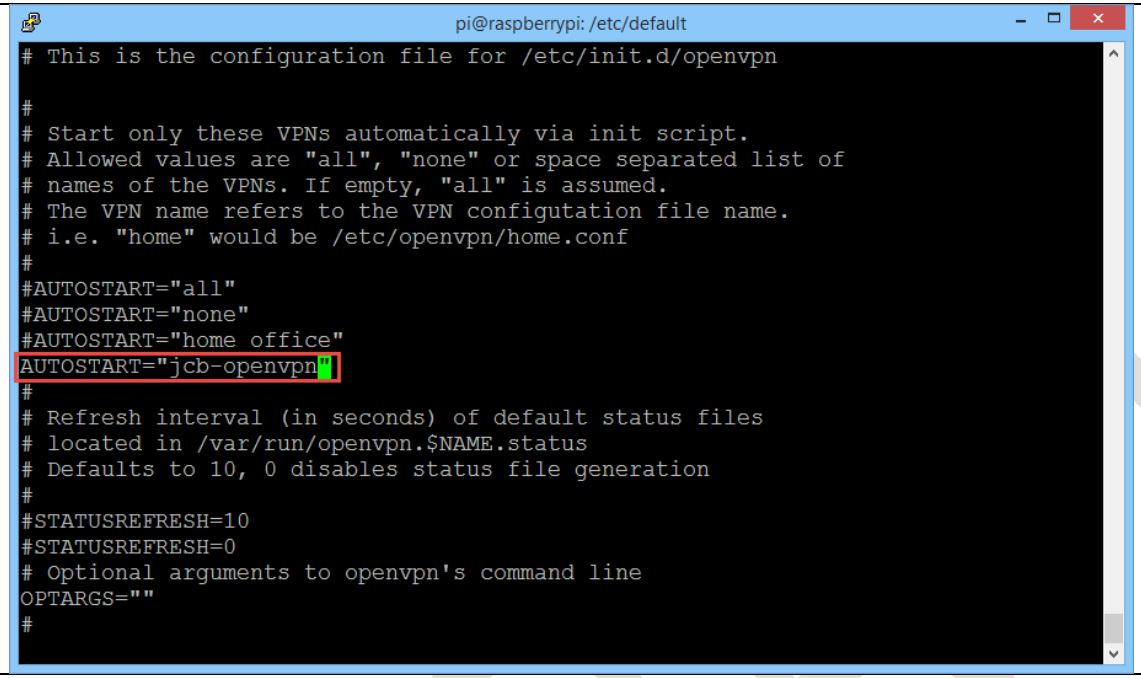
8		The results should look similar to the results in the screenshot.

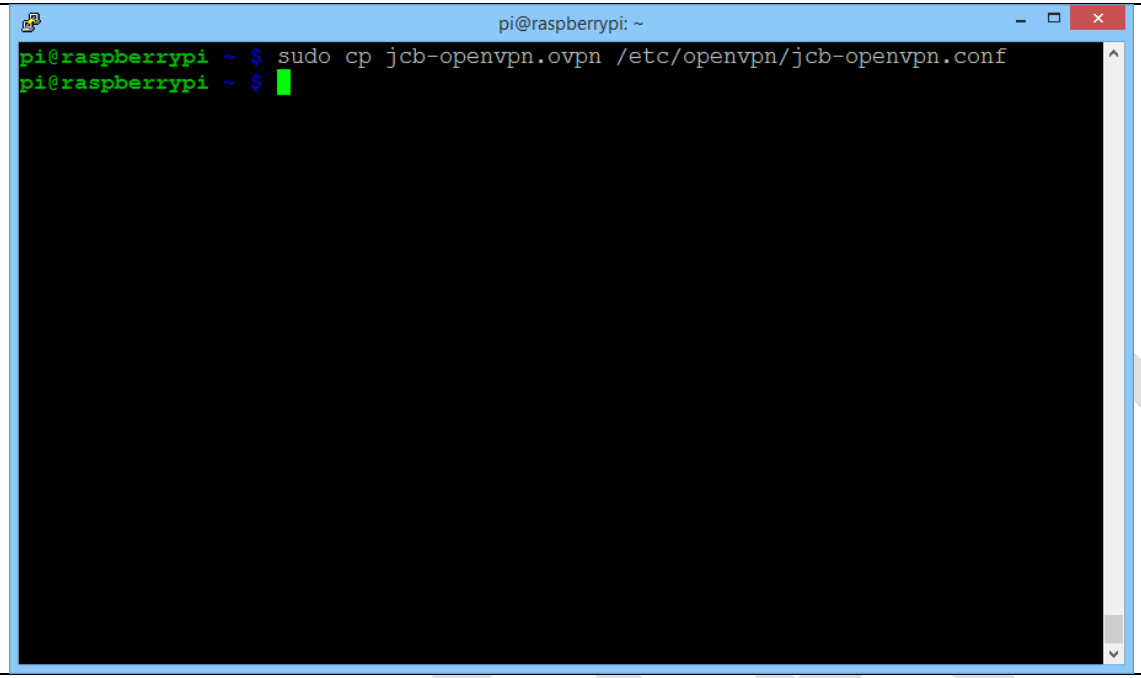
9		See if the OpenVPN tunnel is properly created. Execute “ifconfig”. Note the appearance of the “tun0” interface.

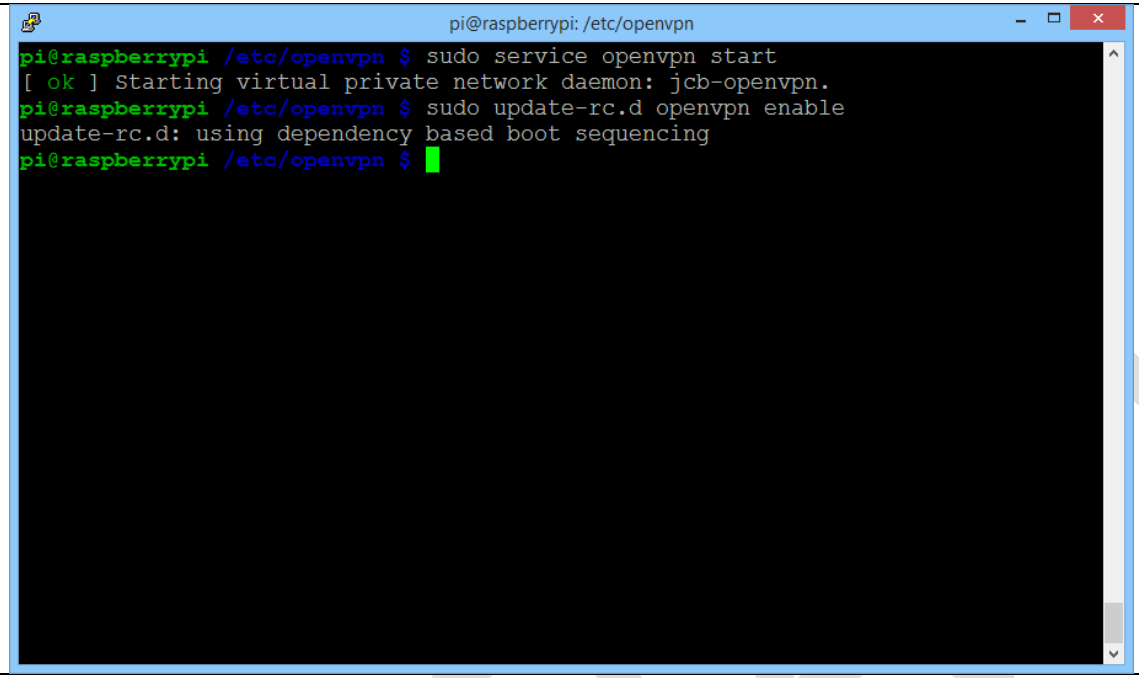

pi@raspberrypi: ~
tun0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
inet addr:10.8.0.14 P-t-P:10.8.0.13 Mask:255.255.255.255
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:2 errors:0 dropped:0 overruns:0 frame:0
TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
RX bytes:152 (152.0 B) TX bytes:375 (375.0 B)
wlan0 Link encap:Ethernet HWaddr 74:da:38:3b:12:72
inet addr:192.168.2.226 Bcast:192.168.2.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:28001 errors:0 dropped:152 overruns:0 frame:0
TX packets:5091 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:13283283 (12.6 MiB) TX bytes:739447 (722.1 KiB)
wlan1 Link encap:Ethernet HWaddr 74:da:38:3b:12:88
inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:24734 errors:0 dropped:21 overruns:0 frame:0
TX packets:1463 errors:0 dropped:24 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:322659 (315.0 KiB) TX bytes:909112 (887.8 KiB)

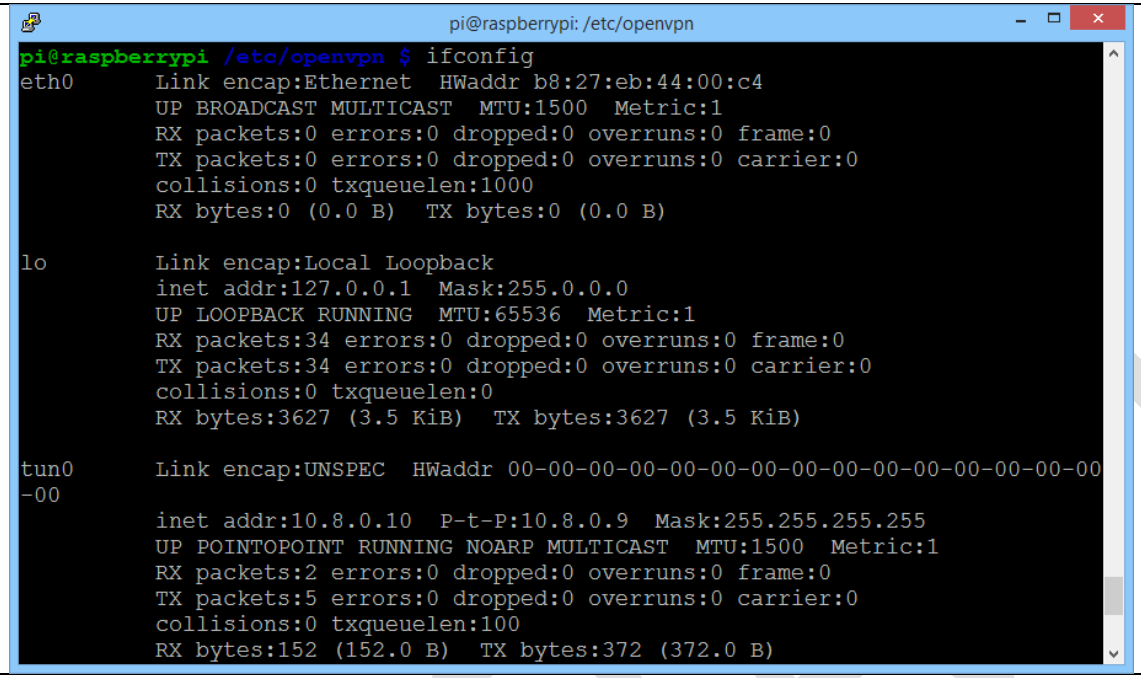
10		However, the routing rules are not yet configured to use the VPN tunnel. And, we'd like the OpenVPN tunnel to start on boot. In the window running the "openvpn" client, type Ctrl-C.

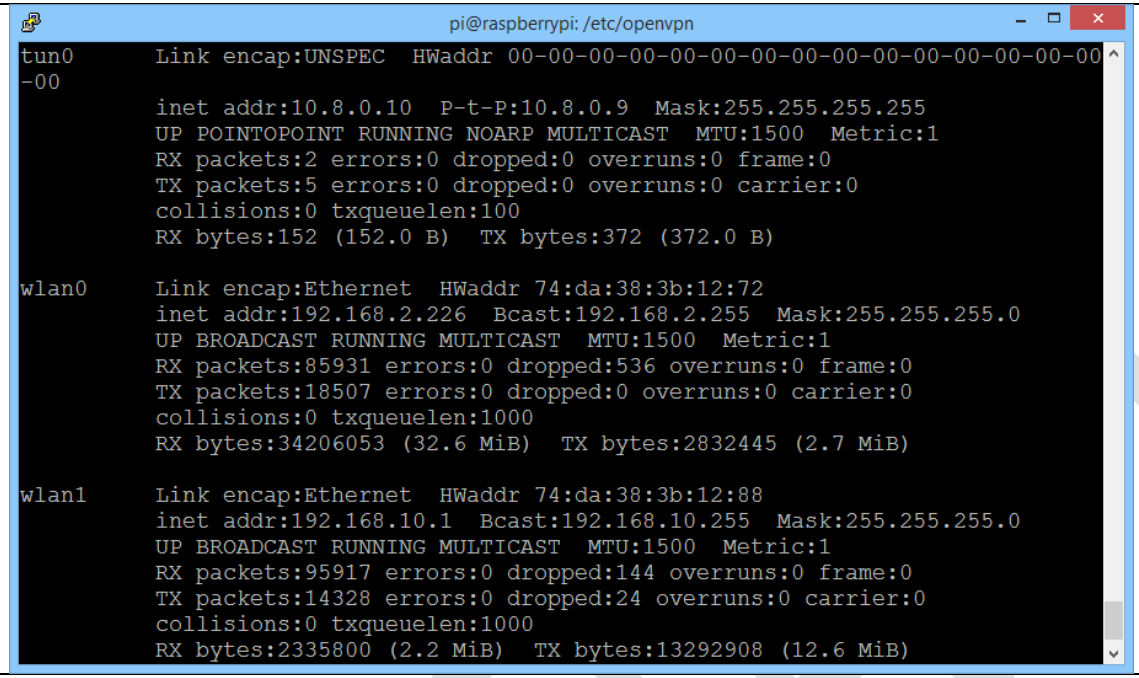
11		We need to edit the startup options for OpenVPN. Execute “cd /etc/default”. Execute “vi openvpn”

12		Configure OpenVPN to automatically start a VPN connection using the configuration for “jcb-openvpn”. Add the line AUTOSTART=”jcb-openvpn” as seen in the screenshot and save the file.

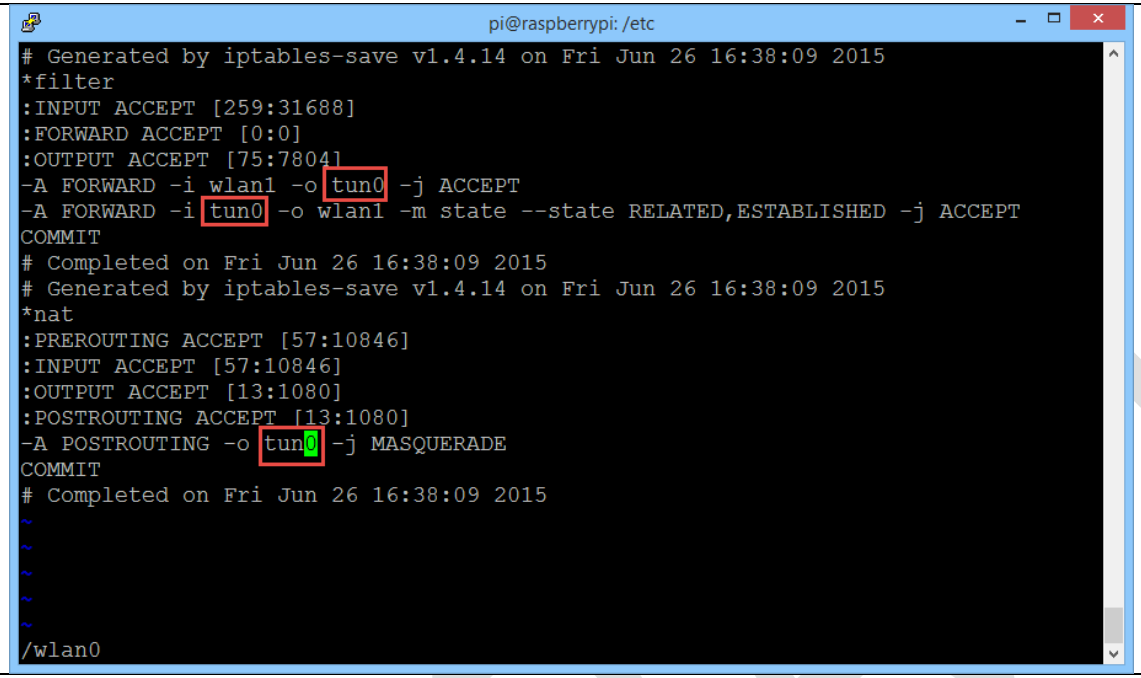
13		Copy the OpenVPN client configuration file to the location expected by OpenVPN. Execute “sudo cp jcb-openvpn.ovpn /etc/openvpn/jcb-openvpn.conf”

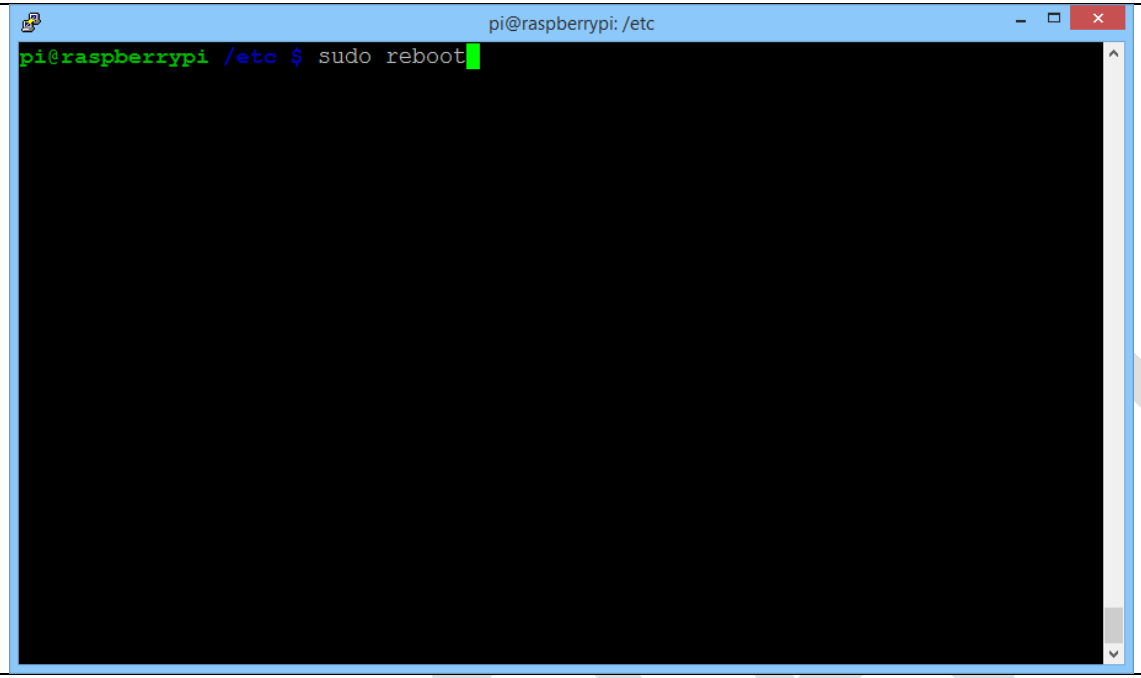
14		Start the OpenVPN service and configure it to automatically start on boot. Execute “sudo service openvpn start” Execute “sudo update-rc.d openvpn enable”

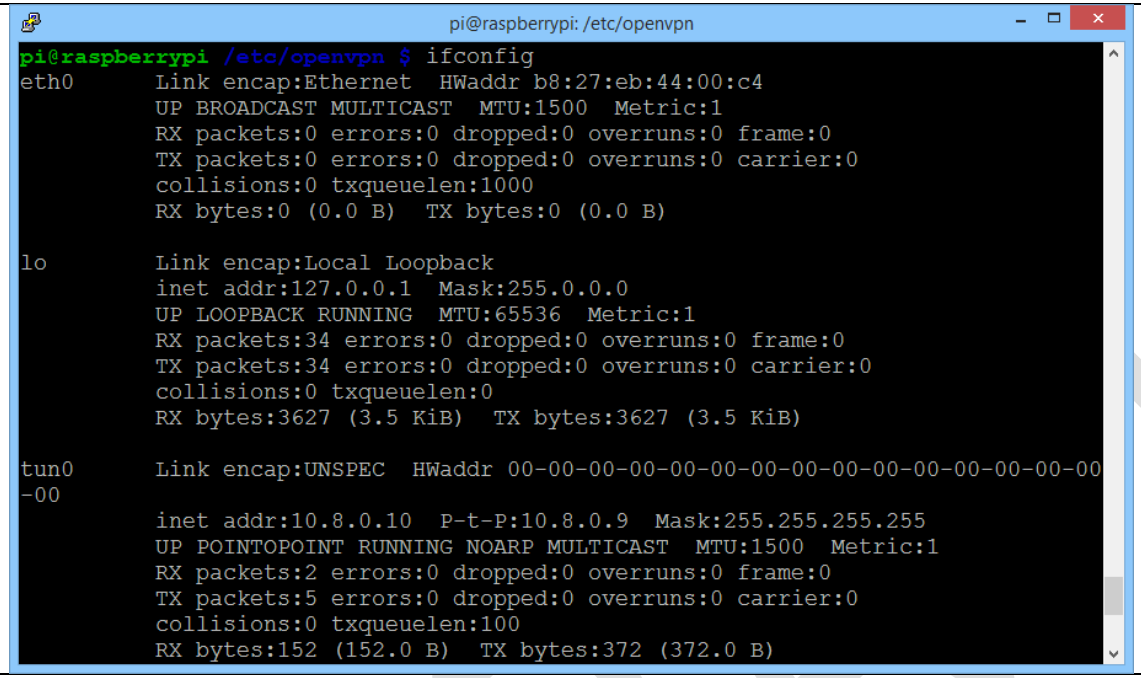
15		See if the OpenVPN tunnel is properly created. Execute “ifconfig”. Note the appearance of the “tun0” interface.

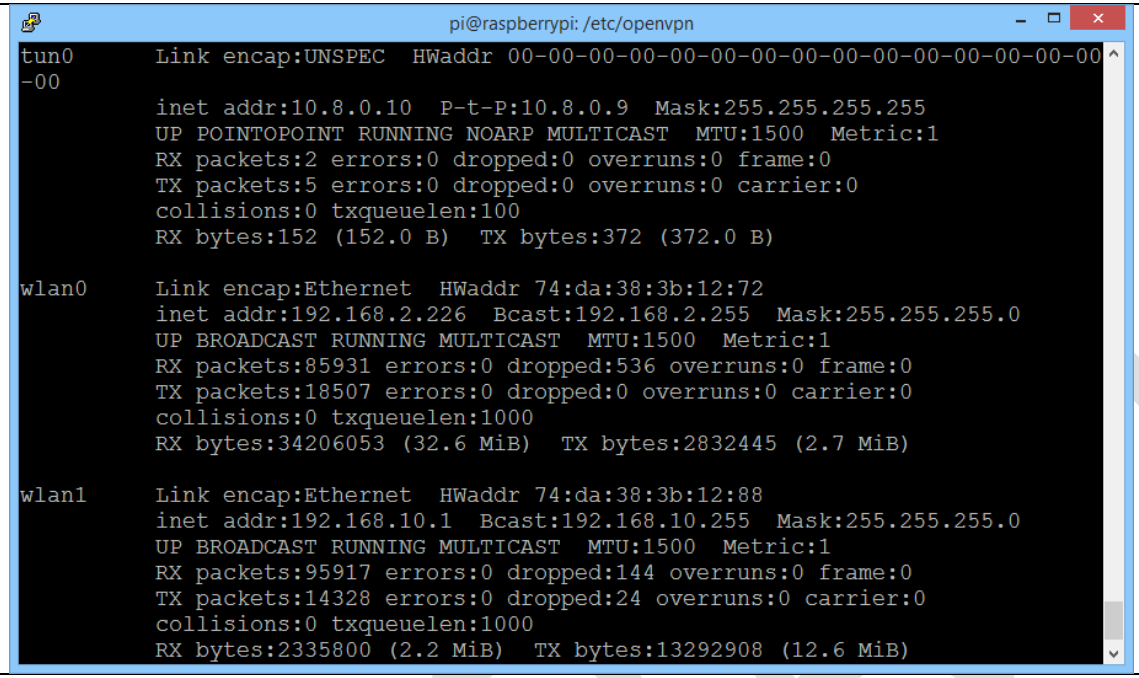

pi@raspberrypi: /etc/openvpn
tun0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
inet addr:10.8.0.10 P-t-P:10.8.0.9 Mask:255.255.255.255
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:2 errors:0 dropped:0 overruns:0 frame:0
TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
RX bytes:152 (152.0 B) TX bytes:372 (372.0 B)
wlan0 Link encap:Ethernet HWaddr 74:da:38:3b:12:72
inet addr:192.168.2.226 Bcast:192.168.2.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:85931 errors:0 dropped:536 overruns:0 frame:0
TX packets:18507 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:34206053 (32.6 MiB) TX bytes:2832445 (2.7 MiB)
wlan1 Link encap:Ethernet HWaddr 74:da:38:3b:12:88
inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:95917 errors:0 dropped:144 overruns:0 frame:0
TX packets:14328 errors:0 dropped:24 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:2335800 (2.2 MiB) TX bytes:13292908 (12.6 MiB)

16		We need to update the routing rules to use the VPN tunnel. Execute “cd /etc” Execute “sudo vi iptables.restore”

17		Change all instances of “wlan0” to “tun0”

18		Reboot the Raspberry Pi to both make the routing rule changes effective and check the automatic start of the VPN.

19		See if the OpenVPN tunnel is properly created. Execute “ifconfig”. Note the appearance of the “tun0” interface. Test the VPN by connecting to the hot spot from another Raspberry Pi, computer, or even mobile device.


pi@raspberrypi: /etc/openvpn
tun0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
inet addr:10.8.0.10 P-t-P:10.8.0.9 Mask:255.255.255.255
UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
RX packets:2 errors:0 dropped:0 overruns:0 frame:0
TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
RX bytes:152 (152.0 B) TX bytes:372 (372.0 B)
wlan0 Link encap:Ethernet HWaddr 74:da:38:3b:12:72
inet addr:192.168.2.226 Bcast:192.168.2.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:85931 errors:0 dropped:536 overruns:0 frame:0
TX packets:18507 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:34206053 (32.6 MiB) TX bytes:2832445 (2.7 MiB)
wlan1 Link encap:Ethernet HWaddr 74:da:38:3b:12:88
inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:95917 errors:0 dropped:144 overruns:0 frame:0
TX packets:14328 errors:0 dropped:24 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:2335800 (2.2 MiB) TX bytes:13292908 (12.6 MiB)