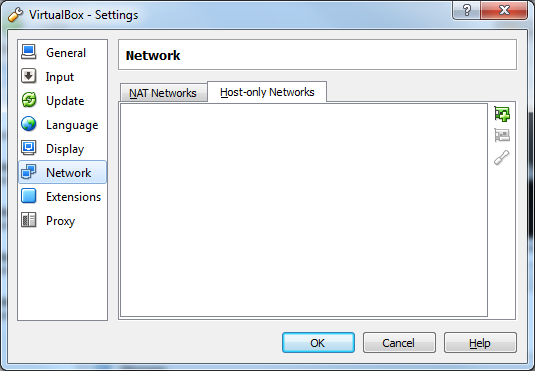
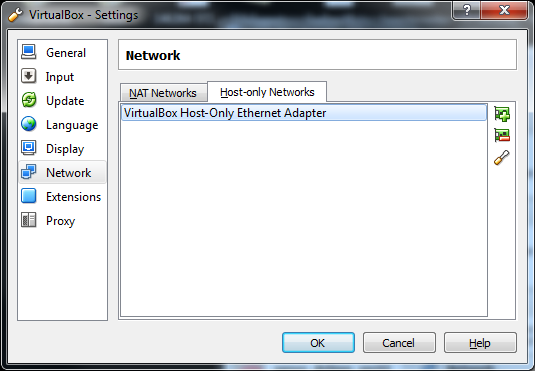
## Creating a network between virtual machines in VirtualBox

The process to create a virtual network between virtual machines within VirtualBox is as below. **ENSURE YOUR VIRTUAL MACHINES ARE POWERED OFF UNTIL TOLD TO POWER THEM ON**:

1) Create a new network with DHCP for the new, internal network. It will be used by our internal devices. There are a couple of ways to do this, however, we are going to focus on creating a network on our host operating system. This is to say, we will have a network set up as if there was a switch inside our virtual environment that our virtual machines could connect to. This is called a Host-only Network. It is likely already in place, we are going to check/verify settings.

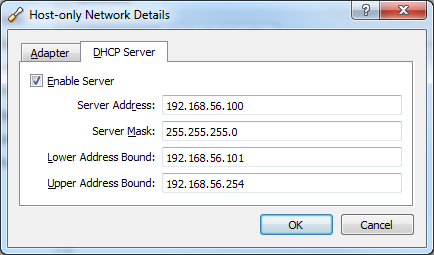
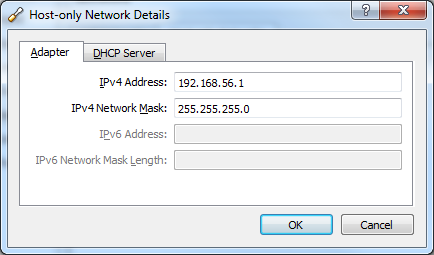
We also need to specify that we want to use DHCP, and give a range of IP addresses to use with DHCP. As such, let’s configure the following. First, with all your virtual machines off, go to the settings of VirtualBox (not the settings for any virtual machine, not yet). Select Network settings, and click on the Host Only Networks tab. Click on the icon that looks like a network card with a plus symbol, as below circled in orange. You may need to give Oracle permission to do this:  


This will create a new Host-Only Ethernet Adapter. We need to now ensure the settings are correct:



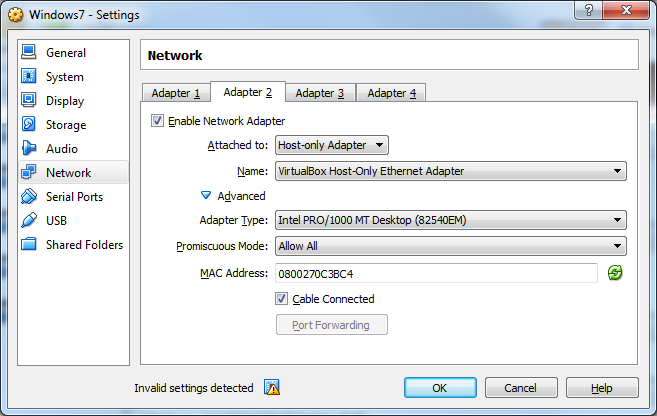
Click on the screwdriver icon circled in orange above, to configure your new Host-Only Adapter. You will need to ensure your adapters are similar to the settings below. An IP address is automatically selected, (mine was set to 192.168.56.1) so yours may not match exactly what is below, but as long as all the addresses are from the same 192.168.x.0-255 pool of addresses, you are good to go. You need to now go to your DHCP Server tab, and enable the DHCP server.

Ensure that whatever address you have on the adapter tab is used for the server side. Again in my example, my adapter is 192.168.56.1, so I ensure my DHCP Server address is 192.168.56.100, my Server Mask is 255.255.255.0 (same as on the adapter tab), my Lower Address Bound is 192.168.56.101, and my Upper Address Bound is 192.168.56.254.

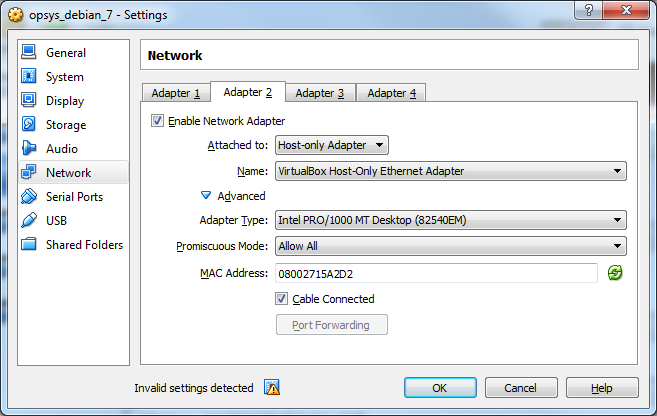


As above, you may need to allow Oracle to make changes to your computer, for security reasons.

2) We need to create adapters for our virtual machines for this task specifically. First, let’s add an adapter for Windows. With the VM powered off, go to the settings dialog box for your Windows virtual machine instance, select Network and click on the Adapter 2 tab. You want to enable this (check the checkbox) and change the “Attached to” dropdown to “Host-Only Adapter”. It should find the Host-Only network you created above. Expand the Advanced settings, and ensure Promiscuous Mode is set to Allow All. Click OK to save your settings, as the rest of the settings should be OK.

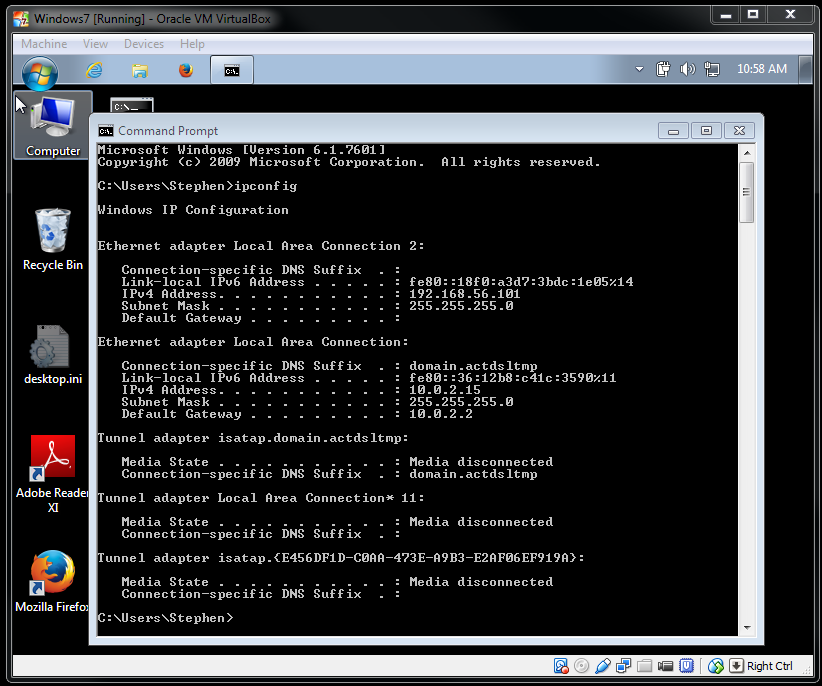


We will need to do the same for our Debian VM. Again, with the VM powered off, go to the settings dialog box for your Debian instance, select Network and click on the Adapter 2 tab. You want to enable this (check the checkbox) and change the “Attached to” dropdown to dropdown to “Host-Only Adapter”. It should find the Host-Only network you created above. Expand the Advanced settings, and ensure Promiscuous Mode is set to Allow All. Again, click OK to save your settings, as the rest of the settings should be OK.



3) Enable the new network interfaces in their individual operating systems

For Windows, you need only confirm that the new network interface has been activated. Start your Windows VM, Go to a command window and type ipconfig. You should see the following:

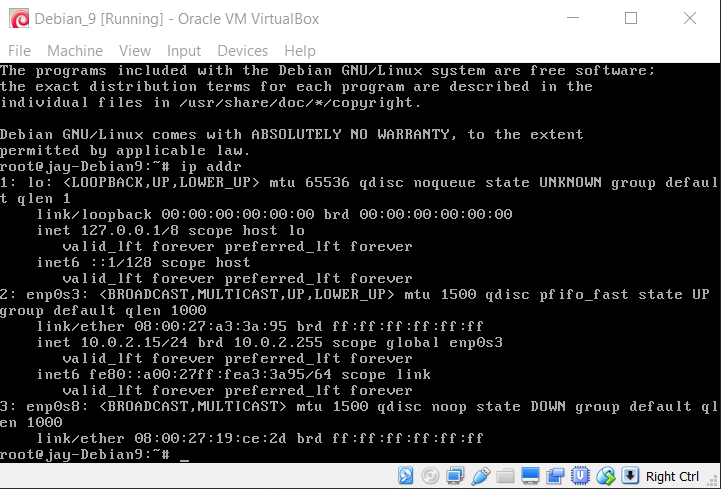


Note you now have 2 network adapters. The Ethernet Adapter identified as Local Area Connection 2 is the newly created adapter, and is using an IP address from the pool of IP addresses we specified in our new network configuration we did in step 1. Windows should be sorted now.

Now let’s look at Debian:

When we start our Debian machine, log in as normal, become root and check our network configuration with the ip addr command, we get the following:

ip addr



Unlike Windows, Debian didn’t automatically load the new network adapter we set up for it. We also need to take note of the adapter names. In the past, names were pretty straight forward. Not so much now. Note above I have three adapter names yours may be different:

* **lo** used for loopback connections
* **enp0s3** used for my initial network setup
* **enp0s8** newly created but not configured.

We can configure the new network adapter by editing a file called /etc/network/interfaces using the utility nano, as follows:

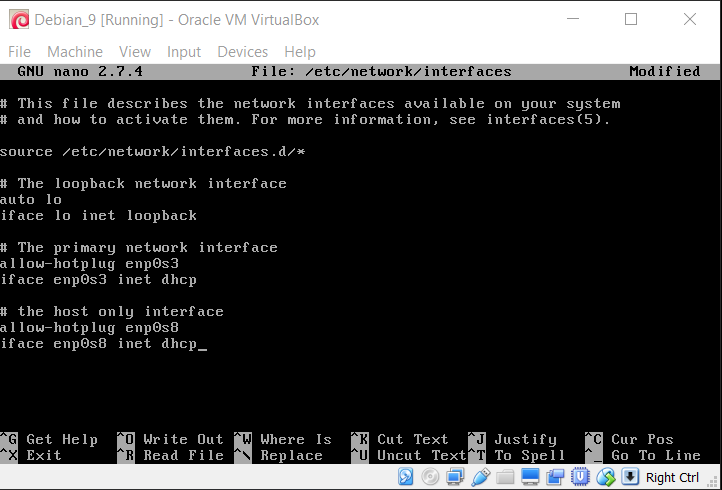
nano /etc/network/interfaces

You will need to add the following 3 lines, based on your adapter names:

# the host only interface  
allow-hotplug enp0s8  
iface enp0s8 inet dhcp

These lines can be added to the bottom of your file. You should note they are very similar to the three lines already there.

Your modified file should look like the following:



Another is to restart network services with the following command as root:

ifdown enp0s8; ifup enp0s8;

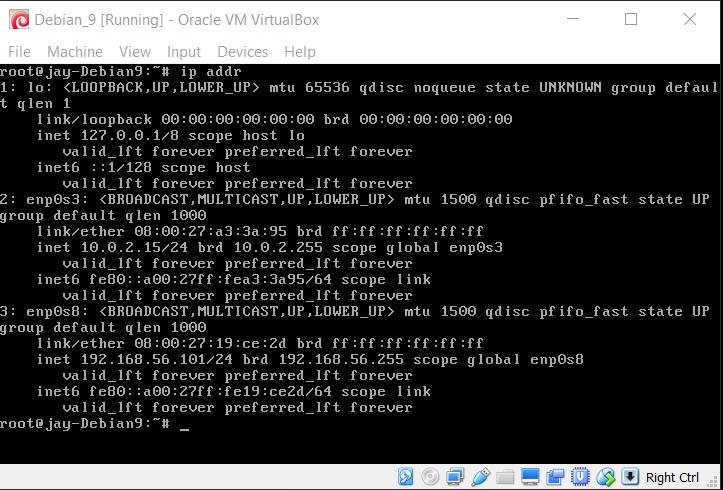
You are better to restart your virtual machine, as that seems to work better, and ensures that the network settings show up properly in the future.

Hit Ctrl and O to write out, or save your work. Keep the same file name, and Ctrl and X to exit.

We need to activate this setting change. Easiest is to restart our VM.

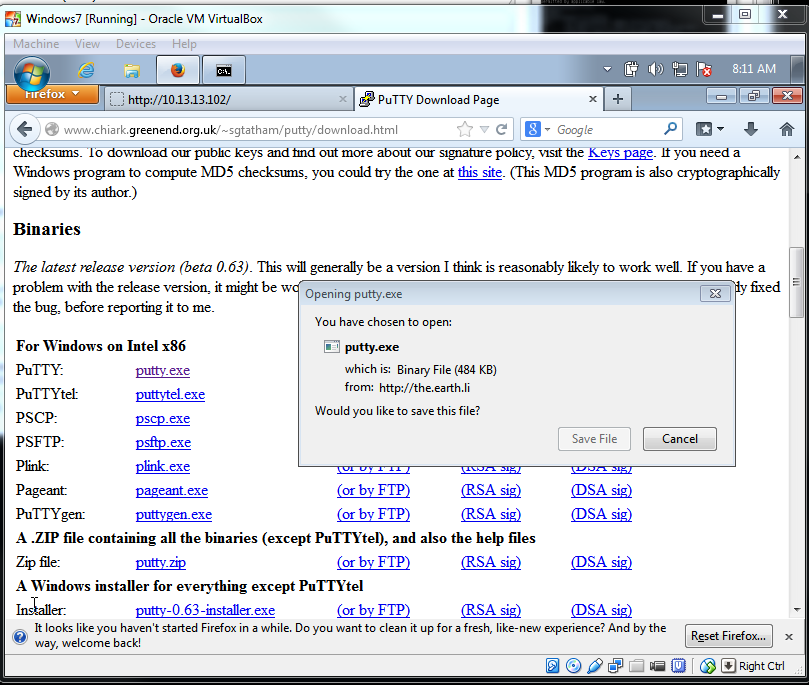
Once restarted, you should be able to show the network interfaces in Debian by running either /sbin/ifconfig as a regular user, or ifconfig as root. Run the command in Debain below and you should see the results indicated in the image at the top of the next page:

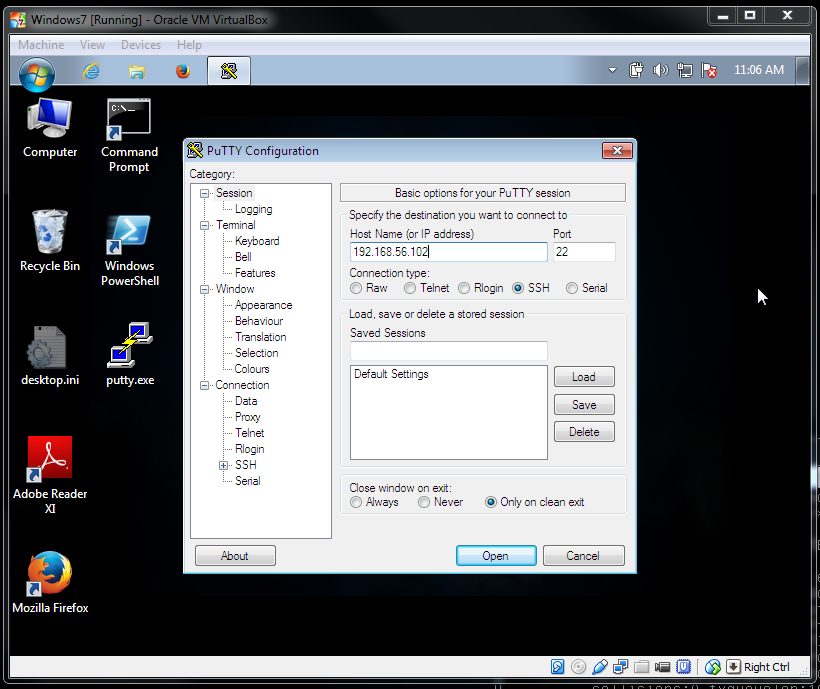
ip addr



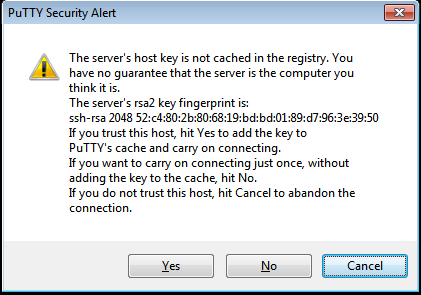
Note again that the new interface, inp0s8 now uses an IP address from the pool we set up earlier. **Note this IP address, as we will be using this to test our connection.**

4) Test Connection – First you want to download PuTTY. Google putty, and download from the greenend.org.uk domain:

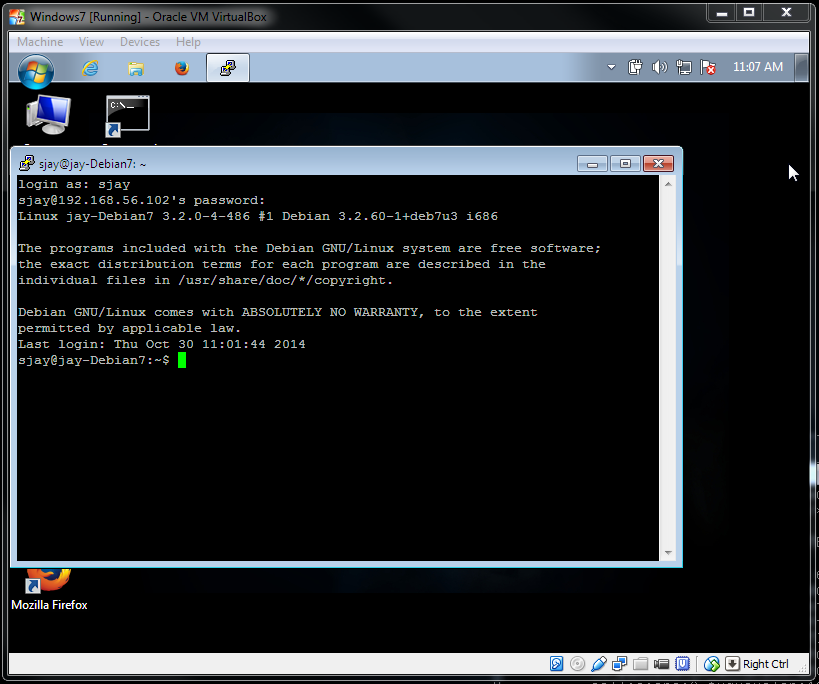


You may save/move this file to somewhere easily accessible. I put mine on my desktop. When you run it, you will be prompted to ensure you wish to run this, as it is from the internet. This is fine. Once launched, you want to connect to your Debian machine, so use the IP address from the eth1 interface. In this example, the IP address was **192.168.56.102; yours will likely be different.** 

Click open, and accept SSH key. Your key will have been randomly generated when you installed sshd during the original installation of Debian, and should be different than the one below. This is OK.



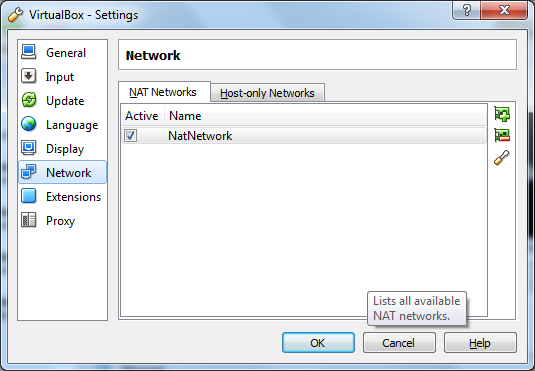
You should now be able to log into your Debian VM from your Windows VM.   
**When done, type exit to quit**



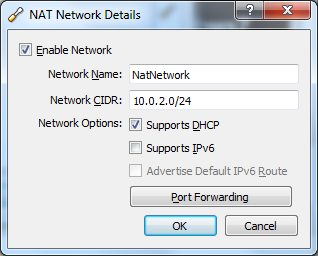
### Problems

If you are having problems with your VM, try the following:

1) Try ensuring/adding a NAT network to your system. Go to NAT Networks, verify you have something like the below:



The details should be as follows:



To prevent problems from putting your VMs to sleep and restarting them at home, run the following command (you should substitute the name for your Debian VM below as necessary).

cd C:\program files\oracle\virtualbox

VboxManage modifyvm "VM Name" --natdnsproxy1 on