NAT Lab

Objectives

- You are going to set up a NAT server. We have learned that NAT maps a private IP address to a public one and vice-versa. Using 2 virtual machines, you are going to implement NAT in the following manner: Your windows host will be on a completely private network and will have to go through a newly created gateway (your linux box) in order to get out to the Internet.
- Understand what a default gateway is.
- Practice using private and public ip addresses

Instructions

1. You will need access to both your windows virtual machine and your linux virtual machine for this assignment.
2. From your windows machine do the following:
   - Change your default IP address to a private address 192.168.1.2
   - You default gateway should be 192.168.1.1
   - Your subnet mask should be 255.255.255.0
   - Your nameservers should be 144.38.192.2, 144.38.192.3
3. From your linux machine, do the following:
   - Make sure that your eth0 address has been assigned one of your public IP addresses. To find your public ip address visit the [vm.cs.dixie.edu](http://vm.cs.dixie.edu) webpage. It doesn’t matter which one you use as long as you aren't using it anywhere else. You will want to use an appropriate default gateway, 255.255.255.248 as the default subnet mask. If it asks you for nameservers: 144.38.192.2, 144.38.192.3. Once you have appropriately assigned the address to your nic, you should be able to access off-campus websites with your browser on that machine. See the sample screenshot at the end of these instructions for the format of these lines.
     - If you have installed a GUI version of Linux, you should be able to find the network setting easily enough. If you can’t find it, or are using a non-GUI version of Linux, you will need to edit `/etc/network/interfaces`. You are going to have to edit the file anyways in the next step regardless of GUI.
     - Do an `sudo ifdown eth0` and `sudo ifup eth0`
     - Use `ifconfig` to verify that eth0 has an ip address assigned.
4. Before moving to the next step, check to see that your Linux machine can ping other machines. You could try pinging a nameserver, or even www.google.com or some other website of your choice.
5. Create another `interface`.
   - Create a second “virtual nic” on your machine. Obviously this isn’t a physical nic that you are plugging in.
   - Remember that you must be root (admin) in order to change any of your configuration settings. From the command line: `sudo bash`
   - The easiest way to add another virtual nic is to open the file `/etc/network/interfaces` and edit it so it looks similar to the following lines. (You should already have an entry for iface eth0 (and you may have a couple of extra lines, which is ok). The critical part is to add the 4 lower lines.)
     ```
     auto eth0
     iface eth0 inet static
     address 192.168.1.2
     netmask 255.255.255.0
     gateway 144.38.192.1
     
     iface eth0:0 inet static
     address 192.168.1.1
     netmask 255.255.255.0
     gateway yourIPAddress
     ```
   - Note that is one of you public IP addresses. We are telling all traffic coming in from eth0:0 to go to eth0 ip address.
6. You can “turn on” that newly created interface by issuing the command `sudo ifup eth0:0`. If it gives an error, just ignore it.
   - Now when you issue the command `ifconfig`, you should see entries for both eth0 and eth0:0, and you should be able to ping your windows machine and your windows machine should be able to ping your linux machine. If this does not work, you need to double check everything from above. THIS IS A CHECKPOINT, if you cannot ping from one machine to the other (your windows to your linux machine), something is messed and you should stop and go fix it. (Hint: You can open the windows command prompt and type `ping 192.168.1.1` to ping your Linux machine)(You won’t be able to ping in reverse because the windows firewall is on)
7. Those settings from above are nice, but you will not be able to get beyond your linux machine. Right now our windows machine has a private address that is connected to our linux machines private address. We haven’t told our linux machine how to route and most importantly how to do NAT. Remember that our
private address can’t be used on the public internet, so we have to tell our Linux machine that it needs to rewrite all packets that come in on the private interface to use its' public address so we can send it on the network. To do this, issue the following commands:

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
echo '1' | sudo tee /proc/sys/net/ipv4/ip_forward
```

8. Now, you should see if a browser on your windows machine can access off campus. If not, see some steps below.

**NOTES AND TROUBLESHOOTING:**

- Before doing any other troubleshooting, issue the command `sudo ifdown eth0 && sudo ifup eth0; sudo ifdown eth0:0 && sudo ifup eth0:0;` from your linux machine.
- If the `ifup` command yells at you, just ignore it. You should see that your interface is up if you issue `ifconfig`.
- Double check your networks and subnet masks. Remember that between your linux and windows machine you should be using a 192.168.x.1-2 IP address, with a subnet mask of 255.255.255.0. Your IP from your Linux machine to the rest of campus should have a 144.38.. network address with a 255.255.255.248 subnet mask.
- If you need to clear out your nat table and start over (messed up on command 4a above). That command gets rid of all NAT entries. You will have to reissue the above commands.

```
sudo iptables -t nat --flush
```

- Turn off your windows firewall
- Control + C will kill a process on your linux machine.
- See your instructor for other problems.

Here is a fancy image of what my linux network settings look like:
How to pass this off:

You need to show the following:

- The output of `ipconfig` on your Windows machine. Immediately followed by the output of `ping 144.38.192.2`.
- The output of `ifconfig` on your Linux machine
- The output of `iptables -t nat -L` on your Linux machine