



CIS 192 Linux Lab Exercise

Lab 2: Temporary Network Interface Card Configuration Spring 2009

Lab 2: Temporary Network Interface Card Configuration

The purpose of this lab is to configure the NICs (network interface controllers) of several Linux systems to join one or more networks. This includes setting the IP address, network mask, default gateway, and DNS settings for different distributions of Linux. Putty and SSH will be used to traverse through the various systems after the interfaces have been configured.

Supplies

- VMWare Server 1.08 or higher
- 192 VMs: Frodo, Elrond, and Fang
- Virtual networks: VMnet3

Preconfiguration

- Original versions of all VMs. Note, this will set the network configurations back to DHCP settings.
- You will need access to a DHCP server to assign addresses to the 172.30.4.0 /24 network. This is already configured if the lab is done using the CIS VMware Stations in the CIS Lab (room 2504) or the CTC. If you plan to do this lab at home see: <http://simms-teach.com/howtos/129-working-at-home.pdf>

Forum

Use the forum to ask for help, post tips and any lessons learned when you have finished. Forum is at: <http://simms-teach.com/forum/viewforum.php?f=18>

Background

There are two ways to configure a NIC. One way uses the **ifconfig** and **route** commands to make immediate changes, however these changes are temporary and will only remain in effect till the next system reboot. The second way, which is permanent, is to edit network configuration files which are then used to configure the NIC when the system is booted. We will be using the temporary method for this lab.

The NIC configuration can be static or dynamic. If static, an IP address is manually configured and care must be taken to avoid duplicate IP addresses. If dynamic, a DHCP server is required to provide an IP address.

Each distribution has graphical utilities for configuring the NICs however they are all different. We will use the command line which is the same for all distributions. For this lab we will use the Frodo (Ubuntu), Elrond (CentOS) and Fang (OpenSUSE) VMs.

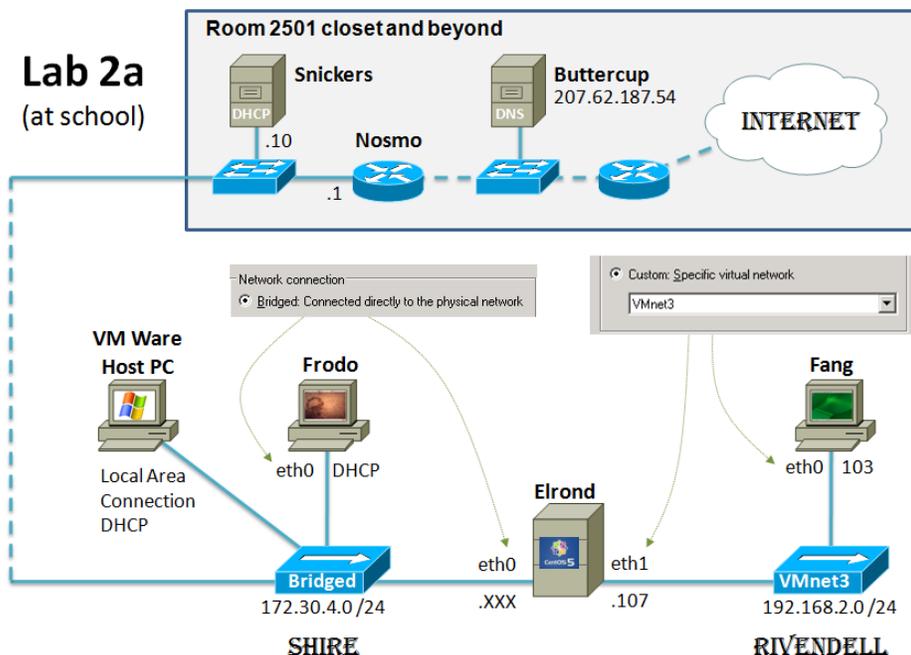
The new commands we will be using for this lab are:

- **ifconfig**
- **dhclient**
- **ping**
- **route**
- **ssh**

Procedure

There are VMware Host Stations in the CIS Lab (room 2504) and the CTC. These are Windows PC's with VMware Server installed. All the stations are physically connected to the Shire (172.30.4.0 /24) network. The gateway for the Shire network is Nosmo which is a Cisco router at 172.30.4.1. Nosmo is in the Room 2501 closet and is connected to the Cabrillo network. Snickers, at 172.30.4.10, is a Windows 2003 Domain Controller that provides DHCP services on the Shire network. The Cabrillo DNS server at 207.62.187.54 provides name resolution services for the Internet.

Using VMs we will implement the network diagram below. There will be two networks called Shire and Rivendell. The Shire network is physical and for VMs to join it their Ethernet device must be set to **Bridged**. These VMs are sharing the physical NIC on the host PC. You can think of the Bridged setting as all the VMs and the host PC have been plugged into the same network hub. The Rivendell network is virtual and uses **VMnet3**. For VMs to join this network, their Ethernet device setting must be set to VMnet3. Think of this as Elrond's eth1 and Fang's eth0 are plugged into the same network hub.



Record this:

1. Create a **text** file on Opus or on the station you selected. Use a text editor like vi or Notepad++. Name the text file *lab2*.
2. Add the following to *lab2*:
 - a. CIS 192 Lab 2
 - b. Your Name
 - c. Date
 - d. TBA hours used to complete this lab (fill this in at the end of the lab)
 - e. Station number of the VMware host PC

Step 1 – “Cabling”

1. **Revert** the Frodo, Elrond and Fang VMs back to their snapshot.
2. Edit the Ethernet Settings for each VM and check that the current “cabling” of your VMs matches the diagram above. Adjust if necessary.

Step 2 – Power on Frodo and check the NIC configuration

1. Power up Frodo which is running the Ubuntu Linux distribution. Ubuntu is based on Debian and has been working hard to become widespread on desktops and notebook PCs.
2. Login as **cis192**
3. Start up a terminal window and use **su –** to become root. Note the “–” on the su command is important because it establishes root’s shell environment settings.
4. Use **lspci** to check the NIC hardware (AMD 79C970).
5. Use **lsmod | grep pcnet32** to see the loaded NIC driver kernel module (pcnet32).
6. The pristine (reverted back to snapshot) version of Frodo has **DHCP** configured for eth0. This greatly simplifies NIC configuration because it is done automatically using a DHCP server.
7. Issue the **ifconfig** and **ifconfig eth0** commands and check that eth0 and lo are up and running. For eth0, note the MAC address (HWaddr), IP address (inet addr), subnet mask (Mask) and broadcast address (Bcast).
8. If your eth0 interface does not have an IP address then recheck that the VM has its Ethernet device settings set to **bridged** and issue the **dhclient** command to request an IP address again from the DHCP server.
9. Use the **ping 172.30.4.1** command to ping the router. You must hit Ctrl-C to stop the pinging. Alternatively you can use the **-c** option to set the ping count. Try **ping -c 5 172.30.4.1** to test this. Note the success or failure of the ping.
10. Shut down the eth0 interface with **ifconfig eth0 down** and verify it is down with the **ifconfig** command (should no longer see the eth0 interface). Try to ping the router again which should now fail.
11. Bring the interface back up again with **ifconfig eth0 up** and make sure you can ping the router again.
12. Now let’s test if DHCP also picked up a default gateway and the address of a DNS server. This is required to ping hosts out on the Internet. Use **ping 207.62.186.9** (which is opus.cabrillo.edu) and **ping google.com** to verify that the default gateway and DNS settings were configured.
13. Use **cat /etc/resolv.conf** to see the DNS server being used. Note the DNS server.
14. Use the **route -n** command to show the default gateway. The default gateway will have a destination of 0.0.0.0 (matches all addresses) and a G in the Flags column. Note the default gateway.

Step 3 – Power on Elrond and check the eth0 and eth1 interfaces

1. Power up the Elrond VM which is running the CentOS Linux distribution. CentOS is a replica distribution of Red Hat Enterprise Server. Elrond has two NICs so it can be configured as a router in later labs.
2. Elrond is configured to boot up to run level 3.
3. Log in as root.
4. Use **lspci** to check there are **two** NICs, both AMD 79C970's.
5. Use **lsmod | grep pcnet32** to see the loaded NIC driver kernel module (pcnet32).
6. The pristine (reverted back to snapshot) version of Elrond is preconfigured with both eth0 and eth1 down. That means there will be no network connectivity without some manual NIC configuration.
7. Issue the **ifconfig** command and check that only the lo (loopback) interface is up and running.
8. Try **ping 172.30.4.1** which should fail because the interface is down.
9. Configure a static IP address on eth0 using **ifconfig eth0 172.30.4.1xx netmask 255.255.255.0 broadcast 172.30.4.255** where **1xx** is determined by your station number. To avoid duplicate IP conflicts use the table below to determine **1xx**.

Station	IP	Static 1	Static 2
CIS-Lab-01	172.30.4.101	172.30.4.121	172.30.4.122
CIS-Lab-02	172.30.4.102	172.30.4.123	172.30.4.124
CIS-Lab-03	172.30.4.103	172.30.4.125	172.30.4.126
CIS-Lab-04	172.30.4.104	172.30.4.127	172.30.4.128
CIS-Lab-05	172.30.4.105	172.30.4.129	172.30.4.130
CIS-Lab-06	172.30.4.106	172.30.4.131	172.30.4.132
CIS-Lab-07	172.30.4.107	172.30.4.133	172.30.4.134
CIS-Lab-08	172.30.4.108	172.30.4.135	172.30.4.136
CIS-Lab-09	172.30.4.109	172.30.4.137	172.30.4.138
CIS-Lab-10	172.30.4.110	172.30.4.139	172.30.4.140
CIS-Lab-11	172.30.4.111	172.30.4.141	172.30.4.142
CIS-Lab-12	172.30.4.112	172.30.4.143	172.30.4.144
Pod 1		172.30.4.113	172.30.4.145
Pod 2		172.30.4.114	172.30.4.146
Pod 3		172.30.4.115	172.30.4.147
Pod 4		172.30.4.116	172.30.4.148

10. Use **ifconfig** to check eth0 is up and note the MAC address (HWaddr), IP address (inet addr), subnet mask (Mask) and broadcast address (Bcast).

11. **ping 172.30.4.1** again to ping the router. It should work now. You must hit Ctrl-C to stop the pinging. Alternatively you can use the `-c` option to set the ping count. Try **ping -c 5 172.30.4.1** to test this.
12. Next **ping 207.62.186.9** and **ping opus.cabrillo.edu**. Opus is on the Internet and both pings should fail. What errors do you get?
13. Set the default gateway with **route add default gw 172.30.4.1** and use **route -n** to verify. Try pinging 207.62.186.9 again. It should succeed now as we have a route out and back. Can you ping opus.cabrillo.edu?
14. Set the primary DNS server address with **echo nameserver 207.62.187.54 > /etc/resolv.conf** and retry pinging opus.cabrillo.edu. This should succeed now that the DNS server has been configured and the hostnames can be resolved to IP addresses.
15. Let's configure Elrond's eth1 interface with **ifconfig eth1 192.168.2.107 netmask 255.255.255.0 broadcast 192.168.2.255**.
16. Use **ifconfig** to check **eth1** is up and note the MAC address (HWaddr), IP address (inet addr), subnet mask (Mask) and broadcast address (Bcast).

Step 4 – Power on Fang and check the eth0 interface

1. Power up the Fang which is running the openSUSE Linux distribution. SUSE is a distribution popular in Europe.
2. Login as cis192
3. Start up a terminal window and use **su -** to become root. Note the `-` on the `su` command is important because it establishes root's shell environment settings.
4. Use **lspci** to check the NIC hardware (AMD 79C970).
5. Use **lsmod | grep pcnet32** to see the loaded NIC driver kernel module (pcnet32).
6. Issue the **ifconfig** command and check the interfaces. Fang's interface has been preconfigured to be down.
7. Configure Fang's eth0 interface with **ifconfig eth0 192.168.2.103 netmask 255.255.255.0 broadcast 192.168.2.255**.
8. Use **ifconfig** to check eth0 is up and note the MAC address (HWaddr), IP address (inet addr), subnet mask (Mask) and broadcast address (Bcast).
9. Use the **ping 192.168.2.107** command to ping Elrond's eth1 interface. This should work.
10. Use the **ping 172.30.4.1** command to ping the router. This should fail. Use **route -n** to see why. There is no default gateway or any route to the 172.30.4.0 /24 network.
11. Shut down the eth0 interface with **ifconfig eth0 down** and verify it is down with the **ifconfig** command (should no longer see the eth0 interface). Try to ping Elrond again. Bring the interface back up again with **ifconfig eth0 up** and make sure you can ping the Elrond's 192.168.2.107 interface.
12. Don't worry about setting the DNS server. Until we learn how to set up routing Fang will not be able to get out on the Internet. Setting up routing will happen in the next lab!

Step 5 – Traversing the VM's using Putty and SSH

Even though we don't have routes set up between the Shire and Rivendell networks we can still use Putty/SSH to traverse from system to system. Starting with Putty on your Windows system you can Putty to Frodo, from Frodo SSH to Elrond, from Elrond SSH to Fang. You can also Putty directly into Elrond as well.

1. On the Windows desktop, use Putty to login into Frodo. Note you will need the IP address you noted earlier for Frodo to do this.

2. Still in Putty, from Frodo, use **ssh 172.30.4.1xx** to login to Elrond. Use Elrond's eth0 IP address you noted earlier.
3. Still in Putty, from Elrond, use **ssh 192.168.2.103** to login to Fang.
4. Use **exit** to log off Fang.
5. Use **exit** to log off Elrond.
6. Use **exit** to log off Frodo.

Tip: Use Putty to enable a reliable way to use copy and paste between Windows and the Linux VMs. You can SSH to the VM of choice and then copy command output from the Putty window and paste it into your windows document. Likewise you can copy commands from a Windows document and paste them into the command line of a VM. For example, an easy way to collect data for this lab would be to paste the following commands into each VM:

```
ifconfig eth0
ifconfig eth1
route -n
cat /etc/resolv.conf
ping -c 2 172.30.4.1
ping -c 2 172.30.4.1xx
ping -c 2 192.30.4.1xx
ping -c 2 192.168.2.107
ping -c 2 192.168.2.103
ping -c 2 google.com
```

Note, **Putty uses a non-Windows method of doing copy and paste.** To copy data from putty **just select it.** Any highlighted text in Putty is automatically put on the Windows clipboard. To paste text or commands from the Windows clipboard into the Putty session, just **click the right mouse button.** Be careful using right mouse click in Putty as this will do an immediate paste whether you wanted to or not.

Record this:

Complete and add the following table to your *lab2* file. The 3rd entry has been completed as an example.

Interface Configuration Table

Host	Interface	DHCP/		Subnet Mask	Broadcast	MAC Address
		Static	IP Address			
Frodo	eth0					
Elrond	eth0					
Elrond	eth1	Static	192.168.2.107	255.255.255.0	192.168.2.255	00:0C:29:4E:21:A5
Fang	eth0					

Record this:

Complete and add the following table to your *lab2* file. The 2nd entry has been completed as an example. Use "na" if an entry is not configured.

DNS and Gateway Configuration table

Host	Default	
	Gateway	DNS server
Frodo		
Elrond	172.30.4.1	207.62.187.54
Fang		

Record this:

Complete and add the following table to your *lab2* file. The 1st entry has been completed as an example.

Ping Connectivity Test Results Table

To Destination	From Frodo	From Elrond	From Fang
----- 172.30.4.1	success	success	fail
Frodo			
Elrond's eth0 interface			
Elrond's eth1 interface			
Fang			
google.com			

Record this:

Record in your *lab2* file example commands you used for this lab. This should be a concise set of documented examples that can be used as a resource for repeated operations in future labs.

To turn in

Your *lab2* **text** file should contain the following sections. For details on what to record see the "Record this" sections in blue above.

- Course and Lab title, your name and date
- TBA hours used to complete
- VMware host PC station number
- Interface Configuration Table
- DNS and Gateway Configuration table
- Ping Connectivity Test Results Table
- Linux command examples

Check your work for completeness then submit as many times as you wish up until the due date deadline. Remember, **late work is not accepted**, so start early, plan ahead for things to go wrong and use the forum to ask questions.

scp lab2 cis192@opus.cabrillo.edu:lab2.lastname

(Note: use pscp, rather than scp, if uploading from Windows)

Grading rubric (30 points)

- 6 points for complete submittal, professional appearance and quality
- 6 points for correct Interface Configuration Table
- 6 points for correct DNS and Gateway Configuration table
- 6 points for correct Ping Connectivity Test Results Table
- 6 points for complete and concise command summary