

Goals

- Learn the basic setup of the lab
- Learn how to configure network interface cards on three common operating systems
- Learn how to use some basic networking configuration and testing tools
- Learn how to use a packet capture tool for network testing

Lab Assignment

Today you will accomplish the following tasks:

1. Set up all the patch cables necessary to connect all three of your computers into your Ethernet hub and patch your hub into your assigned router port.
 - The network diagram and prelab assignment shows how it should be configured and which router port to use. Your lab instructor will help you with this, so if you need help, ask.
2. Your three computer systems, running Windows 2008 server, Mac OS-X, and Ubuntu Linux, should be freshly rebooted with only the admin interface on the Windows system configured for network access.
 - The Windows 2008 machine has **two** Ethernet interfaces that we will use in lab. The one labeled “admin” should always be configured for use when you begin a lab, although for some labs you will need to disable it. Whenever a lab requires you to configure a Windows 2008 interface, it implicitly means the interface labeled “*student*.”
3. Manually configure, enable, and test the connectivity for all three of your workstations. The diagram on the last page shows the general router infrastructure that you’ll be connecting to. Your prelab should tell you the appropriate IP addresses to use for each table.
 - *Windows 2008*. Setup is accomplished via the **Start/Settings/Network Connections** menu; the GUI setup is accessed by right-clicking the “**Student Interface**” connections and selecting “**Properties**”. Highlight “**Internet Protocol Version 4**” in the list, then click “**Properties**”. Fill in the fields for IP address, mask, and gateway/router, click “**OK**” and close the main property sheet. You will get a warning about multiple gateways at that time, select yes to proceed with the configuration.
Enable the student interface and disable the admin interface. Your TA will assist if you have questions.
To open a command window for testing network connectivity use the tool bar option (**Start/Programs/Accessories/Command**). The “ping”, “tracert”, “ipconfig”, and “netstat” commands are described in the readings for this lab and in lecture.
 - *Mac OS-X*. Setup is accomplished via the “dock” at the bottom of the desktop, **System Preferences/Networking**; in the GUI window select the **Built-in Ethernet** in the list. Next to **Configure, Manually** should already be selected. Fill in the IP address, mask, and gateway/router fields and click on **apply**. On the Mac you will have to add a nameserver IP addresses. At OU

these are at 132.235.64.1 and 132.235.64.2 (either one should work). Your TA will assist if you have questions.

To open a command window for testing select **Terminal** from the dock at the bottom of the screen (this should open multiple command windows for you use). The readings detail the use of the network diagnostic tools under Mac OS-X.

- *Linux*. Like the Windows and Mac OS machines, Ubuntu has a GUI for network setup. We will not be using this GUI interface at all this quarter in order to give you practice configuring network devices using command line equivalents. You must, however, use the GUI to open a terminal window in order to issue commands using either the **Applications/Accessories/Terminal** menu option, or the icon in the menu bar.

All of the network tools used in the prelab work at this command line. In order to use the tools such as `ifconfig` to *change* network settings rather than display them, you must prefix the commands with `sudo` (*superuser do*) to run the commands as root. Thus, use `ifconfig -a` to find the interface names and display the current status of the interfaces; also try just `ifconfig` and note the difference. In the lab we will configure and enable “eth1”.

You must use `sudo ifconfig eth1 inet 132.235.201.xxx` to set the IP address of an interface. See the readings for other command options for `ifconfig` to set the network mask, etc. You will need `sudo ifconfig eth1 up` to enable the interface.

To add a default router gateway under linux use: `sudo route add default gw routerIPAddress` after the interface has been configured for local network access using `ifconfig`.

4. Trace network connectivity from your Linux computer to the following Internet destinations recording the results for use in your lab report. The Linux command “`script filename`” will save a transcript of any series of commands until `exit` is typed. Save the file at the end of the lab to the off-line storage device of your choice.
 - (a) `www.phy.ohiou.edu`
 - (b) `www.frogn.net`
 - (c) `www.kame.net`
5. *Graduate Students Required (Undergrads extra credit)*. Repeat the previous traceroutes using `tracert` from the Windows 2008 computer (without transcript) and note the difference in behavior between `tracert` and `tracert` for locations beyond OU.
6. From your Windows 2008 computer start a wireshark capture using the instructions outlined in the next section. While the capture is running, issue the following commands from a command window being careful to note on paper at which point packets were generated after the command was issued.
 - (a) `ipconfig /all`
 - (b) `ping 132.235.24.25`
 - (c) `ping www.phy.ohiou.edu`
7. From your Mac OS computer start a wireshark capture using the instructions outlined in the next section. While the capture is running, issue the following commands from a

command window being careful to note on paper at which point packets were generated after the command was issued.

- (a) netstat -i
 - (b) traceroute -n 61.8.0.71
 - (c) traceroute www.oztravel.com.au
8. From your Ubuntu Linux computer start a wireshark capture using the instructions outlined in the next section. While the capture is running, issue the following commands from a command window being careful to note on paper at which point packets were generated after the command was issued.
- (a) netstat -nr
 - (b) netstat -r
 - (c) netstat -n
 - (d) netstat
 - (e) dig www.kame.net
 - (f) dig -x 203.178.141.194
9. Save your traceroute transcript and captures files to off-line storage.

Using Wireshark

As an aside, **Wireshark** is a newer version of the software package called **Ethereal**, the change in the name happened due to trademark issues when the maintainer changed companies. We will use **Wireshark** on all computers. However, there may be slight differences in the versions installed on the three operating systems.

- **Windows 2008 Server** Click on the **Wireshark** shortcut on the desktop. Go to the **capture/interfaces** menu option. On this screen you must select the student interface and click “Options” next to that interface; you will normally select **update list of packets in real time** and **scroll during live capture** in order to see the packets on the screen as they are captured. There are other useful options as well which you may want to explore if you have time.
- **Mac OS-X** **Wireshark** is an X-windows application; on Mac OS-X these applications are often started from the command line. Type “**sudo -b wireshark**”. You must use “**sudo**” because **wireshark** needs administrator (root) privileges to run and **sudo** will run such programs as root (“-b” insures that access to the command prompt is returned to you while **Wireshark** is running). You will typically be asked for a password the first time you use **sudo** each lab until it times out. Once **wireshark** starts up, you should go to through “**capture/interfaces**” menus and then click on “Options” for the interface **en0**. You would normally then select **update list of packets in real time** and **scroll during live capture** in order to see the packets on the screen as they are captured. There are other useful options as well which you may want to explore if you have time.
- **Linux** You must issue the command “**sudo -b wireshark** ” on Linux (like the Mac) because **wireshark** needs root privileges to capture packets. By adding the **-b** to

sudo it tells sudo to run the command in the background so that the same command line window can be used to issue other commands. Otherwise, you will need to open another terminal window to issue ping and traceroute commands. Refer to the Mac instructions above because the version and user interface is the same.

- You will typically save your packet capture files to disk for transfer to personal storage (USB, prime, or oak) until you prepare your lab report. Capture files saved under wireshark on the Mac and Linux systems will be saved as root and will be inaccessible without another “sudo” command. You can make the file accessible to all other commands by changing the ownership of the capture files using the following command: “`sudo chown itladmin <filename>`.” CS students can use wireshark on p1 in the prime lab to examine and print out packets for lab reports. Note: wireshark on p1 defaults to printing *all packets from a capture file* so you can exhaust your print quota very quickly if you do not click on **selected packets only**. You will also typically want to click on **print as displayed** so that your printout matches what you see on the screen.

Lab Report Guidelines

Each report is to be written individually, although the data for the lab can be collected during the lab with your partner/group. They should be typed/word processed and brought to class in printed form.

Lab writeups are due **in class** on the Monday following the lab. They don’t generally need to be more than a few (several) pages. Officially, they need to be “long enough to answer the questions”. See the web page for detailed guidelines. Each lab writeup **must** have a header on the first page that includes:

- Your name
- The lab section that you attended
- Your affiliation (CS ugrad, CS grad, ITS ugrad, ITS grad)
- Your lab partner’s name
- Your lab partner’s affiliation

Questions for Lab 1

1. For the Mac OS-X machine, the Linux machine, and the “student interface” on the Windows 2008 machine, answer the following questions:
 - (a) What IP address did you assign it?
 - (b) What IP address mask did you assign it?
 - (c) What IP address did you use as its router?
 - (d) What command-line arguments did you use (Linux only)?
 - (e) What are the names of all of the interfaces on the system

2. Based on the traceroute data that you collected in step 4, sketch a graph showing a tree that shows the combined routes to all of the places. At each junction in the tree (router), label the round trip times to that point.
3. *Graduate Students Required (Undergrads extra credit)* Note the difference in behavior between `tracert` and `tracert` for locations beyond OU; research the protocol differences between the two programs, and propose a logical explanation as to why they behave differently.
4. Use `wireshark` off-line after lab to examine the packets you captured in step 6. What IP network traffic protocols (ICMP, UDP, TCP) were used by each of the three commands. Print out **one** packet of each protocol and circle the IP protocol field and the source and destination IP addresses in the packet dump.
5. Use `wireshark` off-line after lab to examine the packets you captured in step 7. What IP network traffic protocols were used by each of the three commands. Print out **one** packet of each protocol and circle the IP protocol field and the source and destination IP addresses in the packet dump.
6. Use `wireshark` off-line after lab to examine the packets you captured in step 8. What IP network traffic protocols were used by each of the six commands. Print out **one** packet of each protocol and circle the IP protocol field and the source and destination IP addresses in the packet dump.
7. Generalizing from the previous three questions, if you are trying to diagnose a network connectivity problem from a machine that is lacking access to the Internet, which of the commands from steps 6, 7, and 8 are likely to not work or have long delays or time outs and why?
8. Attach the patch panel worksheet showing the patch cords that you used and where they were connected
9. Attach your entire prelab assignment, signed off by the lab TA at the beginning of lab.

