Lab02:TCP/IP overview (contd.)

OSL, Mon, 6th August 2012

Objective:

1. Revise concepts and tools learnt previously in Lab01.
2. Do further exercises using tcpdump and wireshark, individually.

General instructions

1. This lab is to be done individually.
2. Create a directory called <rollnumber>_lab02. Open a file "lab02.txt" inside the directory using a text editor. As you proceed with the lab instructions below, for each exercise, note down the answers to the exercise along with any interesting observations in the file. ***Take care to ensure that the content in the file is neatly organized***. You will be submitting this file for grading at the end of the lab.
3. Apart from the above file, you may have to submit some code, along with input/output files. The details for this are specified in the respective exercises.

Lab instructions:

Exercise 1: Tcpdump

Explain briefly the purpose of the following tcpdump expressions (from reading man pages) in the report.

1. tcpdump udp port 520
2. tcpdump -x -s 120 ip proto 89
3. tcpdump -x -s 70 host ip_addr1 and not ip_addr2

Exercise 2: Link Layer

Note down in the lab report the answers to all questions asked in the following sub-parts.

1. Run the command "/sbin/ifconfig eth0". Note down the following information about this interface
    1. ethernet address
    2. IP address
    3. MTU size (maximum frame size that the interface supports)
    4. Number of packets transmitted and the overall number of bytes transmitted
    5. Number of packets received and the overall number of bytes received
2. Also find out the manufacturer of the ethernet card, given the first 24-bits of the ethernet address, using google. Note down the manufacturer's name in the lab report.
3. Run the command "/sbin/ifconfig -a". What does this command do? How many interfaces are listed? What is this second interface that is listed? (Google to get information).
4. Run "tcpdump host your_ipaddr" in a window. In another window, "ping remote_host". Choose a remote host/ip_addr that you know is working say www.cse.iitb.ac.in. Do you see the ICMP messages in the tcpdump window?
5. Again run "tcpdump host your_ipddr" in a window. In another window, "ping 127.0.0.1". From the ping output, is the interface "127.0.0.1" up/active? Do you see these ICMP messages in the tcpdump window? Why or why not?
6. Run the command "netstat -i". Compare the out of this command to the statistics you collected as part of exercise 1.1. If different, why different?

Exercise 3: DeMultiplexing

First run "tcpdump -enx -w exercise3.out". This command captures packets all packets going through the ethernet interface and directs the output to the file exercise3.out. Then in another window (say window-2), run "ping ip_address" to generate some traffic. Terminate ping. In the same window, (window-2) ssh to some machine you have an account in, login and then terminate the connection.

Now open the "exercise3.out" in wireshark using ""wireshark -r exercise3.out". You will see a whole lot of packets since you are capturing all packets being received/sent by your interface. We will just focus on a few of these packets. Click on the "protocol" field of the wireshark GUI. It should now order the packets according to the protocol. Answer the following in the report.

1. Arp protocol
   - Click on any one of the ARP packets. Expand the "Ethernet" header. What is the value of the field used in Ethernet header to pass packets to the ARP module? Express it in decimal format.
2. ICMP protocol (used by ping)
   - Click on any one of the ICMP packets. Expand the "Ethernet" header. Which higher level process (protocol) is this packet passed to and what is the value in decimals?
   - Now expand the IP header. What is the value of the field used in this header to pass packets to the ICMP module? Express it in decimal format.
3. NFS protocol
   - Click on any one of the NFS packets. Trace the flow of this packet up the protocol stack i.e specify what all processes handle this packet. Ethernet --> _____(1) ->_____(2) -> _____(3)
   - What are the fields and the corresponding values used at process (2) to pass the packet to process (3)? Does it match the value you noted using /etc/services in lab01?
4. SSH protocol
   - Click on any one of the SSH packets. Click on the IP header field. Specify the source and destination IP addresses. Expand the TCP header. Specify the source and destination port numbers. Which machine (i.e. IP address) is the SSH server?

Apart from the above, add the "exercise3.out" file to the directory.

Exercise 4: More Demultiplexing

In this exercise, we will open two ssh sessions, both between your machine and the same remote host and capture packets exchanged between the machines. Determine the correct tcpdump command to run to capture these packets, where the output is stored in exercise4.out. Open two more windows and in each window ssh to the same remote_ipaddr (run these commands as close in time as possible). Terminate the tcpdump and open the exercise4.out in wireshark.

We will use filters to filter out unecessary packets and capture just the first packet of both sessions in either direction (client to server and back). Type **tcp.flags.syn == 0x02** in the filter's field (this esentially makes use of the syn flag of TCP header, which is set to 1 only in the first packet of the TCP connection). You should see 4 packets listed. Note that the first and the third packet belong to the two different ssh sessions, going from your machine to the remote machine.

Answer the following in the report:

1. What is the port number used by the remote machine for the first and the second ssh session (look at the first and third packet to figure it out)? Are both sessions connected to the same port number on the remote machine?
2. When your machine receives packets from the remote host (second and third packets), how does the TCP layer figure out to which ssh session this packet has to be passed? Specify the value of the fields used by TCP to do this.

Apart from the above, add the "exercise4.out" file to the directory.

<The End>

The directory named <rollnumber>_lab02 that you will submit should contain the following files:

1. lab02.txt
2. exercise3.out
3. exercise4.out

Now tar it as follows:
tar -zcvf <rollnumber>_lab02.tgz <rollnumber>_lab02/

Submit the file <rollnumber>_lab02.tgz via moodle for grading.