LAB-08

Reference:
1. Man pages of the commands.
2. Man page of a new tool 'arping'
3. Video/Slides of 'OSI Protocol stack' and 'Inter-Layer Communication' under Introduction.
4. Video/slides of 'Supporting Protocols' under 'Network Layer'.

Exercise 8.1: Some More Demultiplexing

Goal: Ms. Rani finds ssh protocols fascinating and now wants to capture what happens when two ssh sessions are established (at about the same time) between her machine and the same remote host. Since the sessions have same source and destination IP address, she wants to figure out how the sessions are uniquely identified. Help her capture such a trace. There should be no other background traffic.

Transmission Control Protocol, Src Port: 52138 (52138), Dst Port: ssh (22), Seq: 0, Len: 0
    Source port: 52138 (52138)
    Destination port: ssh (22)
    Stream index: 0
    Sequence number: 0   (relative sequence number)
    Header length: 40 bytes
    Flags: 0x002 (SYN)
    Window size value: 29200
    Calculated window size: 29200
    Checksum: 0x4efa [validation disabled]
    Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale

Guidance:
1. Use two windows to run each ssh session, type the command in both windows and run these commands as close in time as possible. (ssh to a neighboring machine)
2. When opening the trace in wireshark, we will use filters to filter out unnecessary 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 esen-tially 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.

LAB REPORT

1. Explain your design by specifying the exact commands (with options) you will run and in which order. Avoid description unless absolutely necessary.

2. What is the port number used by the remote machine for the first and the second ssh session? Are both sessions connected to the same port number on the remote machine? How do you think the ssh application at remote machine distinguishes between the two sessions?

3. When your machine receives packets from the remote host, 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.

4. Name your trace file as “exercise1.out” and add the file to your roll-number directory.

1. A.

130905628@oslab-23:~$ sudo tcpdump -i any -w exercise5.pcap -n '((src host felicity.netne.net) or (dst host felicity.netne.net))'
tcpdump: listening on any, link-type LINUX_SLL (Linux cooked), capture size 65535 bytes

130905628@oslab-23:~$ sudo ssh -f -w 0:1 felicity.netne.net true

2. A.

Both sessions are not connected to the same port number on the remote machine.

Transmission Control Protocol, Src Port: 52138 (52138), Dst Port: ssh (22), Seq: 0, Len: 0
    Source port: 52138 (52138)
    Destination port: ssh (22)

3. A.

Destination port: ssh (22)

32 bits of the TCP packet spcifies destination port which is 22 for ssh.