VI Semester B.Tech. (CSE) | Winter Semester (2023-2024)

Computer Networks Lab (CSC307)

List of experiments for 12th March 2024:

9.1: Hands-on exercises on network analysis using 'tcpdump' on command-line.

1. **Packet Capture**
    a. Display the distinct types of interfaces available on your system.
    b. Capture and display traffic of only IPv4 network.
    c. Capture the packets on your primary network interface at least for a minute or so. Display the number of packets captured, packets received, and packets dropped.
    d. Limit the number of packets captured to '50' while using the same interface as in the previous question, display the number of packets captured, number of packets received, and the number of packets dropped.
    e. Limit the number of packets captured to '30' while using 'any' interface.
    f. Capture 'ICMP' packets using the interface of your choice, ping www.google.com in the background in another terminal, limit the number of packets to '100'.
    g. Capture the packets of only HTTP traffic irrespective of the interface.
    h. Capture and display the payload (data) of TCP & UDP packets on port '5000'.
    i. Limit the capture of packets to '30' related to one of the specific hosts shown in your terminal and save those captured packets into a text file named "output.txt".
    j. Monitor and display the SYN packets for a specific IP range and save that captured packets into a traffic file named "capture.pcap".

2. **Protocol Analysis**
    a. Capture a single TCP type packet using interface of your choice and display the contents of that packet in a hexadecimal format.
    b. Capture and display TCP traffic with a window size greater than '1000'.
    c. Capture and display all TCP packets using HTTPS connection. Identify the number of HTTPS connections established during the capture window.
    d. Describe the Host header in HTTP requests and demonstrate how tcpdump can be used to extract and analyze this header from captured traffic.
    e. Describe the User-Agent header in HTTP requests and demonstrate how tcpdump can be used to extract and analyze User-Agent information from network traffic.
    f. Write a command that captures all network traffic in a very verbose mode, save those packets in 'analysis.pcap' file regardless of their size and then filters to output to only show lines containing keywords related to 'HTTP cookies and

the Host header which will allow one to easily identify and analyze cookie exchanges and server communication within the captured traffic. Terminate the execution after 15 seconds.

9.2: Hands-on exercises of network analysis using Wireshark software.

This assignment will help you to test your familiarity in using wireshark for packet capture and analysis. Start packet capture in wireshark application and then open your web browser and type in an URL of websites: www.iitism.ac.in, www.microsoft.com, www.google.com . Answer the following questions, based on your experimentation:

1. Show the different protocols that appear in the protocol column in the unfiltered packet-listing window in wireshark GUI.

2. Show the Internet (IP) address of the URLs you visited and what is the Internet address of your computer?

3. Measure how long did it take from when the HTTP GET message was sent until the HTTP OK reply was received for the webpage you visited in your web browser?
   Hint: (By default, the value of the Time column in the packet listing window is the amount of time, in seconds, since Wireshark tracing began. To display the Time field in time-of-day format, select the Wireshark View pull down menu, then select Time Display Format, then select Time-of-day.)

4. Show the two HTTP messages displayed in wireshark GUI after you had visited above URL through your web browser.
   Hint: To do so, select *Print* from the Wireshark *File* command menu, and select "Selected Packet Only", "Displayed" and then ok.

5. Draw Flow Graph by using Wireshark.
   Hint: You can achieve this by choosing Wireshark → Statistics → Flow Graph.

6. Draw a graph that compares the number of captured TCP and UDP packets. What conclusion can you get from this graph e.g. which protocol has more packets?
   Hint: You can do this by using Statistics → I/O Graph.