<u>**Exercise 11 – Communication Protocols**</u>

Important: In order to open a raw socket on linux (for raw data sending and receiving) you have to be **root**

1. Sniffing:
    a. Use a display filter to filter only “telnet” data.
    b. Look at telnet.pcap using wireshark, what is the password for the user student ?
    c. Write a script using pcap, that sniffs for passwords automatically.
        i. Open a fresh python terminal
        ii. Run: from scapy.all import *
        iii. Run: all_packets = rdpcap(“telnet.pcap”)
        iv. Run: all_packets.summary() – Record the output to a file.
        v. Create a function called handle_incoming_packet that handles the each packet uses globals to maintain the state.
        vi. Run handle_incoming_packet on all packets consecutively
        vii. See that it works.
        viii. Use the mode sniff(lfilter=handle_incoming_packet) to run it automatically on all interfaces, make sure sniff() doesn’t collect data for no reason by returning False from within handle_incoming_packet. Code must handle simultaneous incoming connections.

<u>**PLEASE PROVIDE THE FOLLOWING FILES WITH THE FOLLOWING NAMES IN THE ROOT DIRECTORY**</u>:

1. ex11.q1.txt –full details of the work you did in your own words for all of q1 [incl. (a) – (c), (i)-(viii)].
2. ex11.q1.vii.py – sniffer that reads from a pcap file. Write proper code, document in your own words!
3. ex11.q2.viii.py - sniffer that reads from interface directly. Write proper code, document in your own words.

2. Stealth open port network scan.
    a. Create a python script that sends TCP SYN requests to a given IP address on all ports (1-65535).
        i. Use the scapy command to send()
    b. Create another script that listens to incoming SYN+ACK response, and marks the ports as open. The script will generate the following lines <u>**ONLY**</u>:
    FOUND OPEN PORT: [0-9]*
    eg.:
    FOUND OPEN PORT: 80
    c. Why is this mode of scanning called stealth ?

<u>**PLEASE PROVIDE THE FOLLOWING FILES WITH THE FOLLOWING NAMES IN THE ROOT DIRECTORY**</u>:

1. ex11.q2.txt –full details of the work you did in your own words for all of q2 (a)-(c)
2. ex11.q2.a.py – sniffer that reads from a pcap file.
3. ex11.q2.b.py - sniffer that reads from interface directly.

3. ICMP shell
    a. Write a script that sends an ICMP echo reply with the content "Hello, World!"
        i. Use wireshark and the "ping" shell command to understand ICMP.
    b. Write a script that sniffs for ICMP echo reply, executes commands based on those commands, and sends the output back (you can use subprocess.Popen).
        i. This must work locally!: It means that you have to differentiate between sent packets and received packets.
        ii. The sender will be called like this send.py <remote_ip> <file_to_run> [arg1] [arg2] …

<u>PLEASE PROVIDE THE FOLLOWING FILES WITH THE FOLLOWING NAMES IN THE ROOT DIRECTORY</u>:

1. ex11.q3.txt –full details of the work you did in your own words for all of q3 (a)-(b)
2. ex11.q3.a.py – sniffer that reads from a pcap file.
3. ex11.q3.b_send.py - send commands via ICMP echo.
4. ex11.q3.b_recv.py - receives commands(plural) and executes.

4. Basic Intrusion Detection/Prevention System.
    a. Write a script that looks for TCP port scanning
        i. Sniff for incoming TCP traffic, and look for incoming SYN requests.
        ii. Count the amount of SYNs/min, and check the result.
        iii. If it there are more than 10 SYNs/min from one source IP print **<u>only</u>** an alert:
        *** PORT SCANNING ALERT *** ([0-9]*.[0-9]*.[0-9]*.[0-9]*)
        Eg.:
        *** PORT SCANNING ALERT *** (127.0.0.1)
    b. SQL injection alert.
        i. Sniff TCP packets to see if there are any packets incoming to port 80 containing the words: "SELECT", "UPDATE", "INSERT" (in lower, upper or mixed case).
        ii. If such packets are received run iptables to block the sending IP
        hint: iptables -I INPUT -j DROP -p tcp
        iii. BONUS: How would you bypass your IDS ? [**3pts**]
        iv. BONUS: How can you abuse this system to create a DoS [**3pts**]

<u>PLEASE PROVIDE THE FOLLOWING FILES WITH THE FOLLOWING NAMES IN THE ROOT DIRECTORY</u>:

1. ex11.q4.txt –full details of the work you did in your own words for all of q4 (a)-(b), (i)-(iv)
2. ex11.q4.a.py – sniffer that looks for TCP port scanning.
3. ex11.q4.b.py - sniffer that looks for SQL injection and blocks it.