CS 433 Computer Networks (2023-24)

Assignment-01

Total Marks: 75 points
Deadline: 08-Sep-2023 11:59PM.

Instructions:

1. The assignment must be done in a group of 2. Only one of the group members needs to submit the assignment.
2. All the programs must work on debian operating system (Ubuntu/ Kali).
3. Submit the link to the GitHub repository, or a Zip file containing the source code. Include a readme.txt containing the information of the team members and how to run your code.
4. Also submit a write-up (PDF) containing all the references. Explain the implementation in brief and include your observations with sufficient screenshots of the terminal output of your code.
5. Discussion regarding the assignment with members of other teams is strictly prohibited.
6. Plagiarism will result in zero marks for the assignment.

Part I: Packet capture statistics: (25 points)

Goal: A program that opens a raw socket and sniffs all the packets going through your network interface. The program should identify the source IP, destination IP, source port and destination port of different TCP flows.

A TCP flow is defined as a sequence of packets between two endpoints (client and server) that can be identified by a 4-tuple (client IP, client port, server IP, server port).

For implementation of the raw socket in C, look into the following header files:

```
sys/socket.h
sys/types.h
netinet/tcp.h
netinet/in.h
netinet/ip.h
```

Calculate the following expression (<roll_no_of_member_1> + <roll_no_of_member_2>) % 3 and use the `<result>.pcap` file for the assignment. The .pcap files are available here.

Run the following command to replay the raw stream of packets that your program will capture.

```
tcpreplay -i <network_interface> <path_to_pcap_file> --mbps <speed> -v
```

Use the speed parameter in tcpreplay to control the packet rate. As your sniffer might miss packets if a large number of packets arrive at the same time. Disconnect from your wired ethernet connection in your VM, if possible, while replaying the packets using tcpreplay.

You can use `ip a` to find the name of your network interface (eth0 in most cases).

You need to share the following:

a. The source code, preferably on GitHub and in C language, but you are not restricted to any language, and means to compile and execute your code (20 points).

b. An analysis of different flows while performing tcpreplay using the provided packet capture (pcap file). The analysis should contain at least the following (5 points):

- i) The number of flows observed by your program and their 4-tuple.
- ii) A reverse DNS lookup of 5 observed IP addresses.

Part II: Capture the Flag (20 points)

Goal: There are some hidden information in the network packets. Identify them using sufficient network programs.

Use tcpreplay to replay the provided pcap file to identify the flags.

```
sudo tcpreplay -i <network_interface> --mbps=<speed>
<path_to_pcap_file>
```

Use the speed parameter in tcpreplay to control the packet rate. As your sniffer might miss packets if a large number of packets arrive at the same time. Disconnect from your wired ethernet connection in your VM, if possible, while replaying the packets using tcpreplay.

Calculate the following expression (<roll_no_of_member_1> + <roll_no_of_member_2>) % 4 and use the `<result>.pcap` file for the assignment. The .pcap files are available here.

Questions for the CTF can be found here

Part III: Link captured packets to the corresponding process : (20 points)

Extend the code from Part I to include the functionality that links the client application TCP port number to the corresponding process ID of that application. You can look into the following linux commands to achieve this:

```
netstat
ss
fuser
lsof
ps
```

Step 1: The code will sniff the packets for a duration of 30 seconds and process them.
Step 2: Prompt the user for a port number.
Step 3: Once the user enters the port number and presses Enter, the program should output the process ID in a new line.
Step 4: Go to Step 2, unless the user presses Ctrl+C.

Part IV: Network Tools (10 points)

1. Run the Wireshark tool and capture the trace of the network packets on your host device. I expect you would be connected to the Internet and perform regular network activities.

   a. List at-least 5 different network protocols that we have not discussed so far in the classroom and describe in 1-2 sentences the operation/usage of protocol and its layer of operation and indicate the associated RFC number if any (5 points).

   b. Identify any one connection and try to estimate the RTT of that connection (2 points).

2. Identify the application layer protocols and their versions used when visiting the following websites:

   github.com

   netflix.com

   google.com

Explain in a few lines the differences and similarities between the protocols. (2 points)

(Hint: Inspect)

3. List the cookies and identify the characteristics of the cookies setup when you visit *eoffice.iitgn.ac.in* (1 points).