HW3

Due Oct 29, 2020 by 11:59pm **Points** 100 **Submitting** a file upload
Available Oct 15, 2020 at 11:59pm - Dec 9, 2020 at 11:59pm about 2 months

This assignment was locked Dec 9, 2020 at 11:59pm.

Click here to download HW3 files: **HW3.zip** **(https://canvas.wisc.edu/courses/219039/files/13217730/download?download_frd=1)** .

Part1 (40 pts)

In this part, you will find four packet traces (pcap files) that can be read by the WireShark tool (among other tools). You will need to investigate these traces to answer the questions below. To get started you will want to understand how to use WireShark's filtering capabilities. Your solution will be a file `solutions.txt` with answers to the questions below.

Trace 1: HTTP (10 pts)

1. Give three websites (domain name and IP addresses) visited from source IP address 192.168.0.100.
2. Give three search queries and the domain of the site for each query made from source IP address 192.168.0.100.

Trace 2: FTP (10 pts)

FTP is the file transport protocol. There is a lot of information about it on the internet.

1. What is the username and password used to connect to the FTP server?
2. List any (and all) files that were downloaded from the FTP server.
3. List the full path for two files (in different directories) on the FTP server that were NOT downloaded.

Trace 3: Traceroute (10pts)

Traceroute is a tool used to determine the route between two IP addresses. You can find information about it on the internet.

1. Briefly describe how the traceroute tool works including which network protocols are in use.
2. Give the source IP address that issued the traceroute command and the destination IP address.
3. List the IP addresses on the route between source and destination.

Trace 4: POP (10 pts)

The post-office protocol (POP) is used for email.

1. What is the POP username and password?
2. How many emails are in the user's mailbox?
3. Give the contents of from, to, subject, and date for one email message.
4. What email client (application) and operating system is this person using to send and receive email?

Part 2 (60 pts)

In this part, you will write a simple intrusion detection system to detect potential attacks or dangerous behavior in network activity.

Here are three pcaps with example attacks in folder 2:

1. `arpspoofing.pcap` includes ARP spoof attacks.
2. `portscan.pcap` includes TCP SYN port scans.
3. `synflood.pcap` includes TCP SYN floods.

Your job is to write a software IDS (a Python script named **scanner.py**) that takes as input a pcap trace and looks for all the above malicious behaviors. The local network you are protecting is configured with two machines (192.168.0.100 with MAC address 7c:d1:c3:94:9e:b8 and 192.168.0.103 with MAC address d8:96:95:01:a5:c9) and a router (192.168.0.1 with MAC address f8:1a:67:cd:57:6e). Your scanner should:

1. Detect ARP Spoofing attempts. (20 pts)

Output a warning including the content of the spoofing packet. The format of your output should be:

```
ARP spoofing!
Src MAC: XX:XX:XX:XX:XX:XX
Dst MAC: XX:XX:XX:XX:XX:XX
Packet number: XX
```

Packet number should **respect the default packet order** in pcap file and **start from 0.**

Please print the MAC address in hexadecimal format with small letters.

Your program should generate the above message every time it detects a spoofing packet.

No empty line between two successive ARP spoofing messages.

2. Detect Port Scans. (20 pts)

A port scan is defined to occur whenever TCP SYNs or UDP packets are sent to a 100 or more different ports on a target system. The scanner should output a warning including the victim destination IP address and the offending packet numbers. The format of your output should be:

```
Port scan!
Dst IP: XX.XX.XX.XX
Packet number: XX, XX, XX, XX
```

Packet number should **respect the default packet order** in pcap file and **start from 0**.

Your program should generate one above message per IP. For each victim port, you only need to ouput the smallest offending packet number corresponding to that port (i.e suppose port 53 received packet number 100 and 150, you **only** need to consider packet number 100). Also, make sure the reported packet numbers per message are in ascending order.

No empty line between two messages.

3. Detect TCP SYN floods. (20 pts)

Your tool should detect when the number of TCP SYNs to a particular destination (that are not associated with completed handshakes) exceeds 100 per second. The scanner should output a warning including the victim destination IP address, and the offending packet numbers. The format of your output should be:

```
SYN floods!
Dst IP: XX.XX.XX.XX
Dst Port: XX
Packet number: XX, XX, XX, XX
```

Packet number should **respect the default packet order** in pcap file and **start from 0**.

Your program should generate one above message per IP and port. For every victim port, you only need to report the first 101 packets within a second which are detected as a SYN flood attack. Make sure the reported packet numbers per message are in ascending order.

No empty line between two messages.

Program Details:

Your program should take as input the filename of a pcap file that contains captured network packets, for example:

```
python scanner.py example.pcap
```

The output of your program will be the warning messages as described above. You should first output all the messages related to ARP spoofing, then messages related to port scanning, finally SYN flooding (If all three attacks are detectable in a single pcap file).

Please also write a **README** to explain *how to run your code* and *give one line of description of each kind of your scanners*.

We will test your program on new pcap files other than the three we provide. **Please make sure the output of your program matches *exactly* as described to avoid any unnecessary marks reduction during grading.**

HW3-output.zip (https://canvas.wisc.edu/courses/219039/files/15541609/download?download_frd=1) contains sample output files to help you check the desired output style.

Notes:

You are required to use **dpkt (v. 1.9.2)** library for reading pcap files and scanning through different packet headers. Your program should be compatible with Python3.6 or Python3.8. You can simply run `pip install dpkt==1.9.2` to install the dpkt library in the virtual environment used for HW1.

Deliverables:

Submit 3 seperate files:

1. **solutions.txt** wrt the1st part,
2. **scanner.py** wrt the 2nd part,
3. **README** explaining *how to run your code and giving one line of description of each kind of your scanners.*

HW3 rubric

Criteria	Ratings					Pts
Part 1: Trace 1	**10 pts** **Full Marks** All answers correct		**5 pts** **Partial credit** Single correct answer		**0 pts** **No Marks** No correct answers	10 pts
Part 1: Trace 2	**10 pts** **Full Marks** All answers correct	**7 pts** **Partial credit** 2 correct answers	**3 pts** **Partial credit** Single correct answer		**0 pts** **No Marks** No correct answers	10 pts
Part 1: Trace 3	**10 pts** **Full Marks** All answers correct	**7 pts** **Partial credit** 2 correct answers	**3 pts** **Partial credit** Single correct answer		**0 pts** **No Marks** No correct answers	10 pts
Part 1: Trace 4	**10 pts** **Full Marks** All answers correct	**7.5 pts** **Partial credit** 3 correct answers	**5 pts** **Partial credit** 2 correct answers	**2.5 pts** **Partial credit** 1 correct answer	**0 pts** **No Marks** No correct answers	10 pts
Part 2: ARP spoofing README	**5 pts** **Full Marks**				**0 pts** **No Marks**	5 pts
Part 2: ARP spoofing implementation	**15 to >10.0 pts** **Successful implementation**				**10 to >0 pts** **Unsuccesful implementation**	15 pts
Part 2: Port scan detection README	**5 pts** **Full Marks**				**0 pts** **No Marks**	5 pts
Part 2: Port scan detection implementation	**15 to >10.0 pts** **Successful implementation**				**10 to >0 pts** **Unsuccesful implementation**	15 pts
Part 2: SYN flood detection README	**5 pts** **Full Marks**				**0 pts** **No Marks**	5 pts

Criteria	Ratings		Pts
Part 2: SYN flood detection implementation	**15 to >10.0 pts** **Successful implementation**	**10 to >0 pts** **Unsuccesful implementation**	15 pts
			Total Points: 100