ECE404 Introduction to Computer Security: Homework 08

Spring 2024
Due Date: 5:59pm, March 21, 2024

1 Introduction

This assignment marks the start of the system/protocol side of ECE404. The goal of this assignment is to give you a deeper understanding of the transport control protocol (TCP) and its vulnerabilities to denial-of-service (DoS) attacks.

As always, please read the homework document in its entirety before coming to office hours with your questions. The teaching staff have spent a long time writing the assignment to cover many common questions you might have.

2 Problem 1

Write a Python object-oriented program that scans a specific target IP for open ports, and subsequently performs a SYN Flood attack.

2.1 Starter Code

```
class TcpAttack():
    def __init__(self, spoofIP:str, targetIP:str)->None:
        # spoofIP: String containing the IP address to spoof

    def scanTarget(self,rangeStart:int,rangeEnd:int)->None:
        # rangeStart: Integer designating the first port in the
                                range of ports being scanned
        # rangeEnd: Integer designating the last port in the
                                range of ports being scanned
        # return value: no return value, however, writes open
                                ports to openports.txt

    def attackTarget(self,port:int,numSyn:int)->int:
        # port: integer designating the port that the attack
                                will use
        # numSyn: Integer of Syn packets to send to target IP
                                address at the given port
        # If the port is open, perform a DoS attack and return
                                1. Otherwise return 0

```

```
15 if __name__ == "__main__":
16     # Construct an instance of the TcpAttack class and perform
                                    scanning and SYN Flood Attack
```

2.2 Program Requirements

Construct a class called TcpAttack that implements both open port scanning and the SYN flood attack. A breakdown of how you might use the starter code to accomplish this is as follows:

1. Define the constructor of the TcpAttack class:
   - The constructor is an inbuilt function of the class that gets executed when creating new instances of that class.
   - Every instance of the TcpAttack class has two instance variables, `spoofIP` and `targetIP`. Thus the constructor of this class accepts two strings as arguments.
     (a) `spoofIP`: Any IP that is not your own machine's
     (b) `targetIP`: The target of the scan and the SYN Flood attack
     (c) Note that there is a flexibility in how you express the IPs. They can either be expressed as symbolic hostnames or in the corresponding dotted decimal notation.
2. Define the `scanTarget` class method:
   - The method accepts two integer arguments:
     – `rangeStart`: The first port in the range of ports to be scanned
     – `rangeEnd`: The last port in the range of ports to be scanned
   - This method scans the target machine for open ports in the range [`rangeStart`, `rangeEnd`] and writes all open ports detected into an output file called `openports.txt`
   - The format of `openports.txt` should be one open port per line in ascending order.
3. Define the `attackTarget` class method
   - This method accepts two integer arguments:
     – `port`: The port number on which the attack will be mounted on

  - `numSyn`: The number of SYN packets to be sent to the target on the specified port
- This method first verifies if the specified port is open. If so, perform the DoS attack and return 1. Otherwise return 0

2.3 Program Dependencies

For this assignment, you will need to use a combination of functions from the `socket` [2] and `scapy` [1] libraries. Feel free to consult the official documentation for these modules, as well as Prof. Kak's implemenation in Lecture 16.15.

- `socket`: a module that allows you to set up a socket connection
- `scapy`: a module that allows you to create and send network packets

Please note that you will need to install `scapy` in order to use its defined methods and objects. If you elected to create a conda environment at the beginning of the semester, installing `scapy` is as easy as running the following command in your `ece404` conda environment.

```
pip install scapy
```

2.4 Implementation Details for SYN Flood Attack

Note that SYN flood attacks have become more difficult to mount over the years. As shown in Lecture 16.14 of the lecture notes, most ISP's now use BCP 38 ingress filtering to prevent spoofing over a router. Therefore you would have to do the spoofing attack between two computers on the same LAN where the packets would not go through a router.

For this assignment, it is totally acceptable if you do not actually manage to cause a DoS outside your LAN or do not have the means to do it with another computer on the same LAN. We are simply looking to see that a theoretical attack is implemented correctly.

2.5 How to Tell that Your Program is Working

To test that the target machine is actually receiving packets, you should run `tcpdump` (or some equivalent program) while your script is running to see that you are actually sending packets to the target IP address (i.e. start

tcpdump and then run your program). If you are using Windows, you can use **Wireshark** instead of `tcpdump` to look at the packets.

In the event that you are on a busy network, you can use `tcpdump` to selectively sniff packets as outlined in Lecture 16. To further avoid clutter, you can optionally turn off all other applications connecting to the internet. As mentioned below, you will include output from these programs in your homework submission.

If you do not have access to another computer to test on, you can use Prof. Kak's machine in RVL whose symbolic hostname is `moonshine.ecn.purdue.edu`.

2.6 How Your Code Will Be Tested

Your source code will be tested with a script similar to the one below:

```
from TcpAttack import *

spoofIP = '10.10.10.10'
targetIP = 'moonshine.ecn.purdue.edu'

rangeStart = 1000
rangeEnd = 4000

port = 1716
numSyn = 100

tcp = TcpAttack(spoofIP, targetIP)
tcp.scanTarget(rangeStart, rangeEnd)

if tcp.attackTarget(port, numSyn):
    print(f"Port {port} was open, and flooded with {numSyn} SYN
                                    packets")
```

3 Submission Instructions

- For this homework you will be submitting a zip file titled `hw08_<last name>_<first name>.zip`, which consists of:
  - A pdf titled `hw08_<last name>_<first name>.pdf` containing:
    - Output (e.g. screenshots) from **tcpdump** (or equivalent program) of both the port scanning and syn flood attack. **Your**

PDF should indicate in the tcpdump output (e.g. highlight, circle, etc.) which packets were sent as a result of the program you wrote.

  - Example screenshots have been provided below in section 4

– The file `TcpAttack.py` containing your code for the programming problem.

4 Example Screenshots From tcpdump

```
14:20:04.715494 IP 128.46.144.64.64523 > 128.46.144.123.3998: Flags [S], seq 3907085255, win 65535, options [mss 1
460,nop,wscale 6,nop,nop,TS val 2978516925 ecr 0,sackOK,eol], length 0
        0x0000:  4500 0040 0000 4000 4006 19a0 802e 9040  E..@..@.@......@
        0x0010:  802e 907b fc0b 0f9e e8e1 63c7 0000 0000  ...{......c.....
        0x0020:  b002 ffff 7c4a 0000 0204 05b4 0103 0306  ....|J..........
        0x0030:  0101 080a b188 8fbd 0000 0000 0402 0000  ................
14:20:04.716065 IP 128.46.144.64.64524 > 128.46.144.123.3999: Flags [S], seq 1608770264, win 65535, options [mss 1
460,nop,wscale 6,nop,nop,TS val 1706869457 ecr 0,sackOK,eol], length 0
        0x0000:  4500 0040 0000 4000 4006 19a0 802e 9040  E..@..@.@......@
        0x0010:  802e 907b fc0c 0f9f 5fe3 e2d8 0000 0000  ...{...._.......
        0x0020:  b002 ffff 9eed 0000 0204 05b4 0103 0306  ................
        0x0030:  0101 080a 65bc c2d1 0000 0000 0402 0000  ....e...........
14:20:04.716652 IP 128.46.144.64.64525 > 128.46.144.123.4000: Flags [S], seq 1337699075, win 65535, options [mss 1
460,nop,wscale 6,nop,nop,TS val 1518739943 ecr 0,sackOK,eol], length 0
        0x0000:  4500 0040 0000 4000 4006 19a0 802e 9040  E..@..@.@......@
        0x0010:  802e 907b fc0d 0fa0 4fbb ab03 0000 0000  ...{....O.......
        0x0020:  b002 ffff 9309 0000 0204 05b4 0103 0306  ................
        0x0030:  0101 080a 5a86 21e7 0000 0000 0402 0000  ....Z.!.........
```

Figure 1: tcpdump output indicating port scanning

```
14:20:04.886063 IP 10.10.10.10.50548 > 128.46.144.123.1716: Flags [S], seq 0, win 8192, length 0
        0x0000:  4500 0028 0001 0000 4006 5612 0a0a 0a0a  E..(....@.V.....
        0x0010:  802e 907b c574 06b4 0000 0000 0000 0000  ...{.t..........
        0x0020:  5002 2000 9efc 0000                      P.......
14:20:04.892312 IP 10.10.10.10.16857 > 128.46.144.123.1716: Flags [S], seq 0, win 8192, length 0
        0x0000:  4500 0028 0001 0000 4006 5612 0a0a 0a0a  E..(....@.V.....
        0x0010:  802e 907b 41d9 06b4 0000 0000 0000 0000  ...{A...........
        0x0020:  5002 2000 2298 0000                      P..."...
14:20:04.899050 IP 10.10.10.10.10196 > 128.46.144.123.1716: Flags [S], seq 0, win 8192, length 0
        0x0000:  4500 0028 0001 0000 4006 5612 0a0a 0a0a  E..(....@.V.....
        0x0010:  802e 907b 27d4 06b4 0000 0000 0000 0000  ...{'...........
        0x0020:  5002 2000 3c9d 0000                      P...<...
```

Figure 2: tcpdump output indicating SYN flood attack on port 1716

References

[1] Scapy: interactive packet manipulation tool. URL https://pypi.org/project/scapy/.

[2] Socket: Low-level networking interface. URL https://docs.python.org/3/library/socket.html.