TCP/IP Attack Lab[1]

Due Friday, May 6, 2022 @ 11:59pm
(11% of the total course grade)

1. Objective

The learning objective of this lab is for students to gain first-hand experience on vulnerabilities, as well as on attacks against these vulnerabilities. Wise people learn from mistakes. In security education, we study mistakes that lead to software vulnerabilities. Studying mistakes from the past not only help students understand why systems are vulnerable, why a seemly-benign mistake can turn into a disaster, and why many security mechanisms are needed. More importantly, it also helps students learn the common patterns of vulnerabilities, so they can avoid making similar mistakes in the future. Moreover, using vulnerabilities as case studies, students can learn the principles of secure design, secure programming, and security testing.

The vulnerabilities in the TCP/IP protocols represent a special genre of vulnerabilities in protocol designs and implementations; they provide an invaluable lesson as to why security should be designed in from the beginning, rather than being added as an afterthought. Moreover, studying these vulnerabilities help students understand the challenges of network security and why many network security measures are needed. In this lab, students will conduct several attacks on TCP.

This lab will be graded. It has to be completed INDIVIUALLY, but you may want to discuss the lab content with your fellow students.

2. Environment Setup

If you haven't set up your virtual Cyber Security Lab environment using VirtualBox on your laptop, please install and setup VirtualBox and Ubuntu 16.04 virtual machines by following instructions provided in **Lab 1**.

Please note, for Lab 5, you will need to use all three Ubuntu 16.04 VMs to conduct several attacks on TCP. One computer is used for attacking, the second computer is used as the victim, and the third computer is used as the observer.

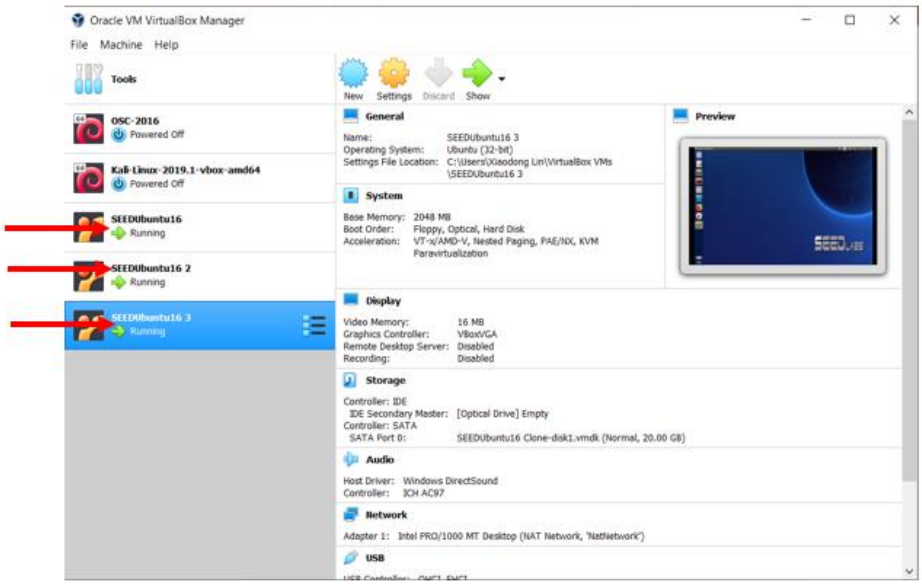

Figure 1. Required Lab Environment Settings

Then, experimental network setup for TCP/IP attacks looks like the following. Note that the IP addresses in your environment may vary.

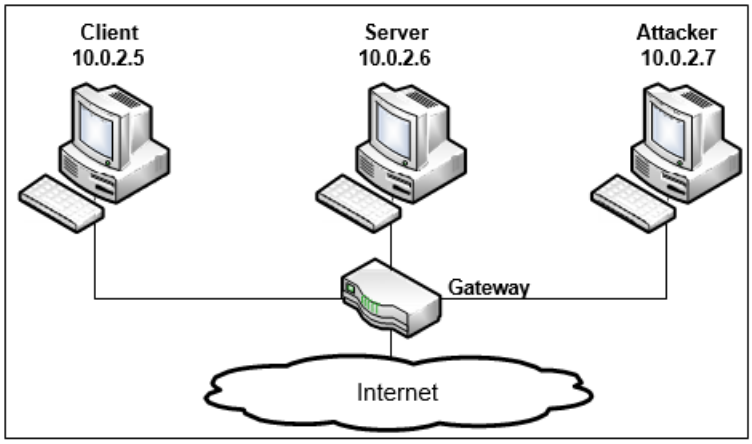

Figure 2. Environment Setup

Netwox Tools. We need tools to send out network packets of different types and with different contents. We can use Netwag to do that. However, the GUI interface of Netwag makes it difficult for us to automate the process. Therefore, we strongly suggest students to use its command-line version, the Netwox command, which is the underlying command invoked by Netwag.

Netwox consists of a suite of tools, each having a specific number. You can run a

command like following (the parameters depend on which tool you are using). For some of the tool, you have to run it with the root privilege:

$ sudo netwox number [parameters ...]

If you are not sure how to set the parameters, you can look at the manual by issuing "netwox number --help". You can also learn the parameter settings by running Netwag: for each command you execute from the graphic interface, Netwag actually invokes a corresponding Netwox command, and it displays the parameter settings. Therefore, you can simply copy and paste the displayed command.

Scapy Tool. Some of the tasks in this lab can also be conducted using Scapy, which is a powerful interactive packet manipulation program. Scapy is very well maintained and is widely used; while Netwox is not being maintained any more. In this lab, you will become familiar with this powerful tool through a simple python program.

3. Lab Exercises

3.1 Task 1: Build Your Own Website

First, you create your personal website on one Ubuntu 16.04 virtual machine, which is the server in Figure 2. Note that an Apache Web Server has already been installed and started on Ubuntu 16.04 virtual machine. For example, if you visit the website on the Ubuntu 16.04 virtual machine, the following default web page shows.

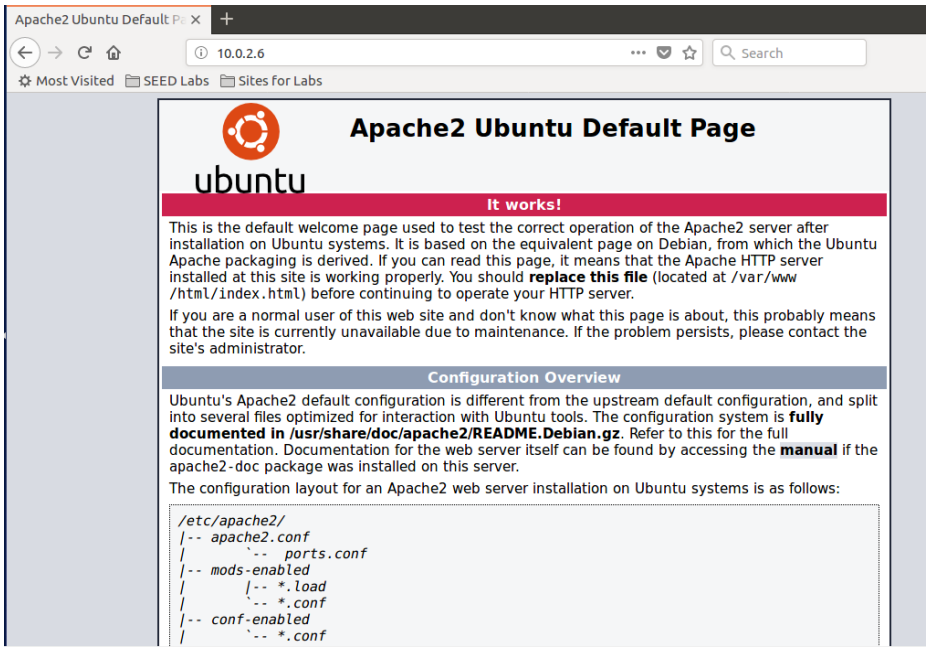

Figure 3. Default web page

Next, add your personal web page, named “XXX.html” into

/var/www/

where XXX is your Southeast University's email ID.

Note that your personal web page should provide a brief personal introduction about you that includes your name, in which year of your study are you, which program you are in. An example of personal web page is shown below

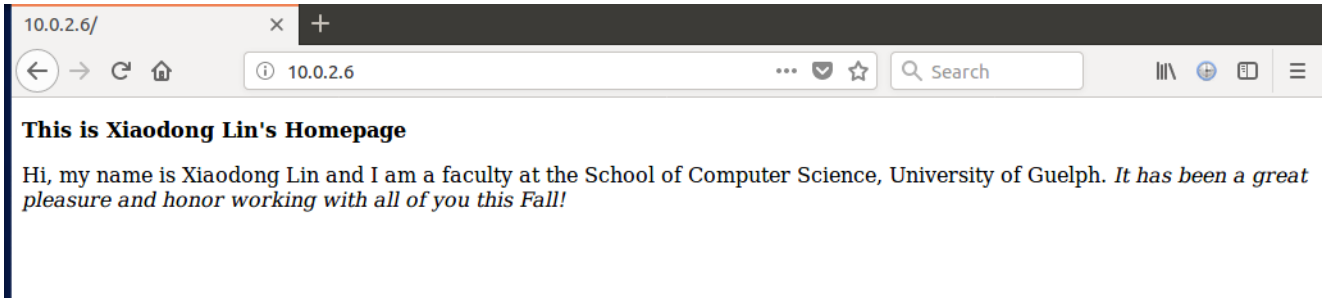

Figure 4. Example of personal web page

Q1: Provide a screenshot of your personal web page. **(0.5 mark)**

3.2 Task 2: SYN Flooding Attack

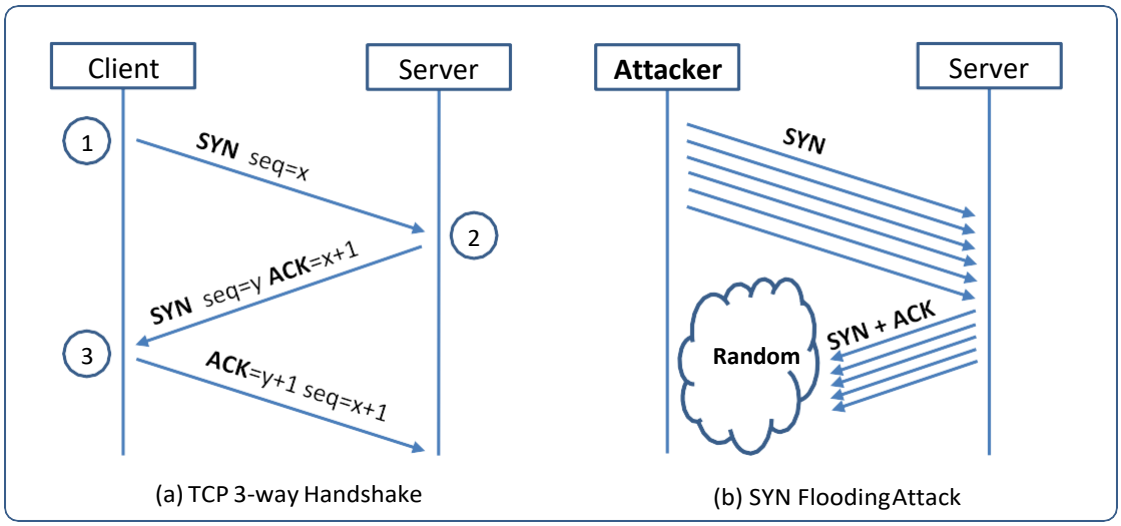

Figure 3: SYN Flooding Attack

SYN flood is a form of DoS attack in which attackers send many SYN requests to a victim's TCP port, but the attackers have no intention to finish the 3-way handshake procedure. Attackers either use spoofed IP address or do not continue the procedure. Through this attack, attackers can flood the victim's queue that is used for half-opened connections, i.e. the connections that has finished SYN, SYN-ACK, but has not yet gotten a final ACK back. When this queue is full, the victim cannot take any more connection. Figure 3 illustrates the attack.

The size of the queue has a system-wide setting. In Linux, we can check the setting using the following command:

```
$ sudo sysctl -q net.ipv4.tcp_max_syn_backlog
```

This value of net.ipv4.tcp_max_syn_backlog (or the size of the queue) indicates the maximum number of queued connection requests which have still not received an acknowledgement from the connecting clients. If this number is exceeded, the kernel will begin dropping requests.

We can use command "netstat -na" to check the usage of the queue, i.e., the number of half- opened connection associated with a listening port. The state for such connections is SYN-RECV. If the 3-way handshake is finished, the state of the connections will be ESTABLISHED.

In this task, you need to demonstrate the SYN flooding attack. You can use the Netwox tool to conduct the attack, and then use a sniffer tool to capture the attacking packets. While the attack is going on, run the "netstat -na" command on the victim machine, and compare the result with that before the attack.

The corresponding Netwox tool for this task is numbered 76. Here is a simple help screen for this tool.

You can also type "netwox 76 --help" to get the help information.

Listing 1: The usage of the Netwox Tool 76

```
[11/12/20]seed@VM:~$ netwox 76 --help
Title: Synflood
Usage: netwox 76 -i ip -p port [-s spoofip]
Parameters:
 -i|--dst-ip ip                 destination IP address {5.6.7.8}
 -p|--dst-port port             destination port number {80}
 -s|--spoofip spoofip           IP spoof initialization type {linkbraw}
 --help2                        display full help
Example: netwox 76 -i "5.6.7.8" -p "80"
Example: netwox 76 --dst-ip "5.6.7.8" --dst-port "80"
```

Q2: In Ubuntu 16.04 VM provided in class, what is the maximum number of queued connection requests which have still not received an acknowledgement from the connecting TCP clients? **(0.5 mark)**

Q3: In your own words, brief explanation how the TCP SYN flood attack works. **(1 mark)**

Q4: Use Netwox to launch TCP SYN flood attack works against the web site on the server (10.0.2.6 in Figure 1). Is the attack successful or not? (Yes/No) Please briefly describe what you have observed. (Note that if you experience long delay when visiting your personal webpage, it means a DoS attack against the web site is successful; otherwise, it is not. It is because the website can become slow to respond to legitimate requests when under DoS attacks. Please wait for a while after you launch TCP SYN flood attack, and then try to visit your personal webpage.) **(0.5 mark)**

Please note that you need super user privilege to use Netwox to launch TCP SYN flood attack works.

There is one Linux kernel configuration related to SYN Flood attacks, called TCP SYN Cookies. **Y**ou can investigate is whether the SYN cookie is enabled or not. You can use the sysctl command to turn on/off the SYN cookie mechanism:

```
$ sudo sysctl -a | grep cookie                    (Display the SYN cookie flag)
$ sudo sysctl -w net.ipv4.tcp_syncookies=0 (turn off SYNcookie)
$ sudo sysctl -w net.ipv4.tcp_syncookies=1(turn on SYN cookie)
```

Be default, the SYN cookie is enabled. Now, you turn off the SYN cookie, and run your TCP SYN flood attack attacks again.

Q5: Is the attack successful or not when the SYN cookie disabled? (Yes/No). Please briefly describe what you have observed. **(0.5 mark)**

Q6: Please explain your observations and what happened to cyber attacks conducted in **Q4** and **Q5**. **(1 mark)**

3.3 Task 3: TCP RST Attacks on `telnet` Connections

The TCP RST Attack can terminate an established TCP connection between two victims. For example, if there is an established telnet connection (TCP) between two users A and B, attackers can spoof a RST packet from A to B, breaking this existing connection. To succeed in this attack, attackers need to correctly construct the TCP RST packet.

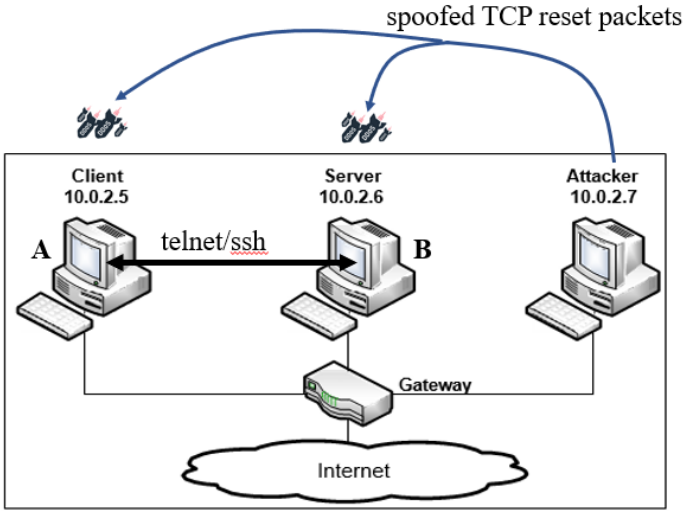

Figure 4: TCP RST Attacks

In this task, you need to launch an TCP RST attack to break an existing telnetconnection between A and B (or the client and the server in Figure above). To simplify the lab, we assume that the attacker and the victim are on the same LAN, i.e., the attacker can observe the TCP traffic between A and B. As such, you can use wireshark to capture telnet traffic on either A or B to analyze the packets.

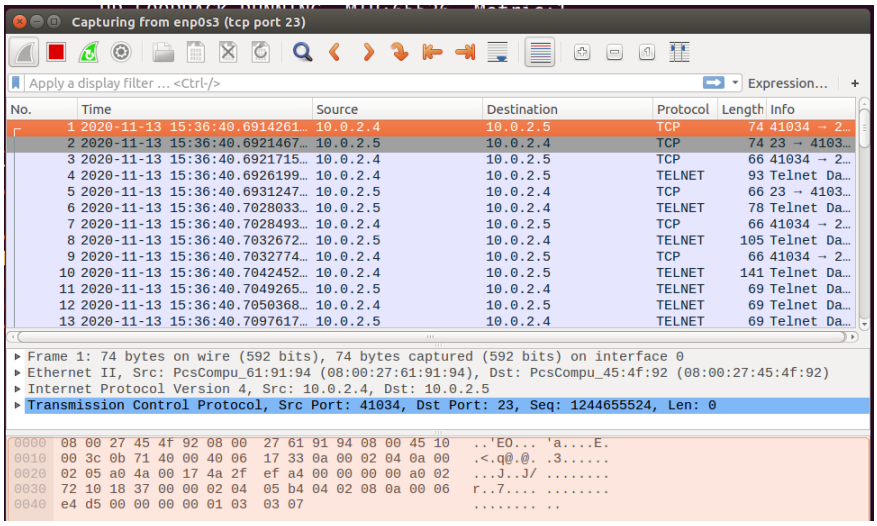

Figure 5: capturing packets with Wireshark

In this task, you need to first use an open source tool, Netwox, to launch an TCP RST attack to break an existing telnet connection between A and B (or the client and the server in Figure 4). As such, before you launch an TCP RST attacks, you must establish a telnet session between A and B.

Using Netwox. A Netwox tool numbered 40 can be used for this task. Here is a simple

help screen for this tool.

Listing 2: The usage of the Netwox Tool 40

```
Title:  Spoof Ip4Tcp packet
    Usage: netwox 40 [-l ip] [-m ip] [-o port] [-p port] [-q uint32]
    [-B]
    Parameters:
    -l|--ip4-src ip               IP4 src {10.0.2.6}
    -m|--ip4-dst ip               IP4 dst {5.6.7.8}
    -o|--tcp-src port             TCP src {1234}
    -p|--tcp-dst port             TCP dst {80}
    -q|--tcp-seqnum uint32        TCP seqnum {rand if unset) {0}
    -B|--tcp-rst|+B|--no-tcp-rst  TCP rst
```

For example,

$ sudo netwox 40 -l 10.0.2.4 -m 10.0.2.5 -o 41040 -p 23 -B -q 319575698

Using netwoxtool 40, we can generate a spoofed RST packet to the client or server. If the attack is successful, you will see a message "**Connection closed by foreign host**" indicating that the connection is broken.

Q7: In your own words, briefly explain how the TCP RST Attack works. **(1 mark)**

Q8: You are required to demonstrate your TCP RST Attacks on telnet connections using Netwox through a step by step MOP (Method of procedure). **(2 marks)**

Note that in order to receive full credit for this question, you must (1) provide screenshots which demonstrate that you have successfully launch an TCP RST attack breaking an existing telnet connection between A and B, and (2) explain how to figure out the important parameters (including the source port number, the source IP address, the destination port, the destination IP address, and sequence number) needed to construct a TCP RST packet for a successful attack, as well as screenshots of Wireshark showing IP and TCP Headers Details for the packet you have analyzed as well as the detailed netwox command).

Next, you will need to develop your own attacking tool using Scapy to launch TCP RST attacks to break an existing telnet connection between A and B. Scapy is a powerful interactive packet manipulation program. It is able to forge or decode packets of a wide number of protocols, send them on the wire, capture them, match requests and replies, and much more [6].

A skeleton python code using Scapy is provided in the following (you need to replace each @@@@ with an actual value):

```
#!/usr/bin/python
from scapy.all import *

print("SENDING RESET PACKET.........")
ip  = IP(src="@@@@", dst="@@@@")
tcp = TCP(sport=@@@@, dport=@@@@,flags="R",seq=@@@@)
pkt = ip/tcp
ls(pkt)
send(pkt,verbose=0)
```

For your convenience, the above python code is provided in a file called reset.py. For example, after you have completed your attacking tool, you can launch a TCP RST attack by typing

$ sudo ./reset.py

Q9: You are required to give the correct values to replace each @@@@ in the skeleton code provided, and submit your modified code as reset_XXX_YYY.py, where XXX is your student ID and YYY is your surname in Pinyin. After you finish the above python program "reset.py", you launch the attack and should be able to disconnect the existing telnet connection. **(3 marks)**

Note that in order to receive full credit for this question, you must: (1) submit your modified exploit code (named reset_XXX_YYY.py) (1 mark); (2) provide screenshots which demonstrate that you have successfully launch an TCP RST attack breaking an existing telnet connection between A and B using your exploit code (1 mark); and (3) explain how to figure out the actual values to replace each @@@@ in the skeleton code provided in order to construct a TCP RST packet for a successful attack (including screenshots of Wireshark showing IP and TCP Headers Details for the packet you have analyzed). (1 mark)

4. Submission

You can submit your answers in a zip file (containing your modified exploit code named reset_XXX_YYY.py, where XXX is your student ID and YYY is your surname in Pinyin and a pdf file named lab6_XXX_YYY.pdf with your answers to the questions in the lab containing the required screenshots). The zip filename must be lab5_XXX_YYY.zip, where XXX is your student ID and YYY is your surname in Pinyin. This naming convention facilitates the tasks of marking for the instructor and course TA. It also helps you in organizing your course work. Failure to follow the requirements will result in mark reduction.

Acknowledgements

This lab has incorporated the materials developed by Dr. Wenliang Du (Syracuse). The copyright of these materials belongs to them.

Reference:

[1] BLOSSOM. Scapy: Performing Network Attacks
https://www.mmu.ac.uk/media/mmuacuk/content/documents/school-of-computing-mathematics-and-digital-technology/blossom/ScapyNetworkAttacks.pdf
[2] S.M. Bellovin. Security Problems in the TCP/IP Protocol Suite.
https://www.cs.columbia.edu/~smb/papers/ipext.pdf
[3] Wenliang Du. Attacks on the TCP Protocol. Computer & Internet Security: A Hands-on Approach, Second Edition, ISBN: 978-1733003926
http://www.cis.syr.edu/~wedu/seed/Book/book_sample_tcp.pdf
[4] How does a TCP Reset Attack work?
https://robertheaton.com/2020/04/27/how-does-a-tcp-reset-attack-work/
[5] CHRIS HOFFMAN. How to Use Wireshark to Capture, Filter and Inspect Packets
https://www.howtogeek.com/104278/how-to-use-wireshark-to-capture-filter-and-inspect-packets/
[6] Scapy. https://scapy.readthedocs.io/en/latest/introduction.html