CITS 3004
Cybersecurity

THE UNIVERSITY OF WESTERN AUSTRALIA

Lab 7: TCP/IP Attacks

Task 1: TCP/IP Attacks

The learning objective of this lab is for students to gain the first-hand experience on the vulnerabilities of TCP/IP protocols, as well as on attacks against these vulnerabilities. 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. Vulnerabilities of the TCP/IP protocols occur at several layers. The attacks we will carry out in the lab are mostly patched in modern operating systems, but it may still be applicable to some existing ones. Again, it is illegal in most countries to carry out these attacks onto other systems and networks that do not belong to you.

To conduct this lab, you need to have at least 2 virtual machines. One computer is used for attacking, the second computer is used as the victim, and the third computer is used as the observer. Using the CITS3004 USB stick, there is a script to launch 2 VMs (i.e., `Start_SEED_VMs.cmd`). Alternatively, you can set up 3 VMs on your own computer. The Lab computers cannot hold 3 VMs due to resource constraints. If you are doing this on your own computer, make sure to power off the VM when you clone it, and also reinitialise the MAC address with the Full Clone option (on Virtualbox). In addition, make sure you have a NAT configured, otherwise the VMs may not be able to communicate to each other. The setup will look something similar as to shown in Figure 1 (addresses may vary).

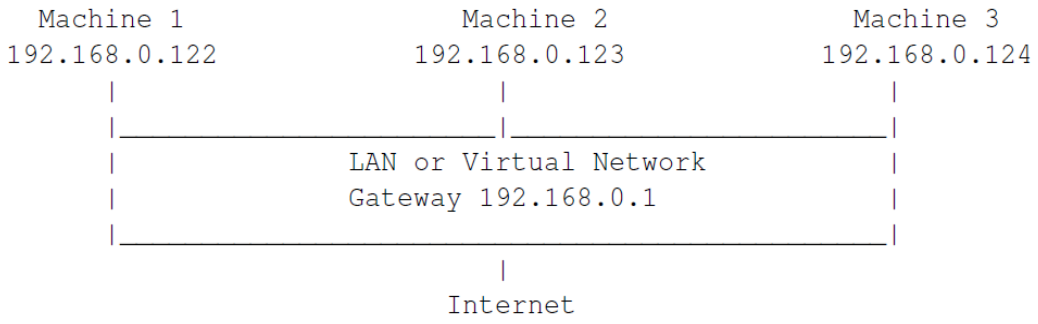


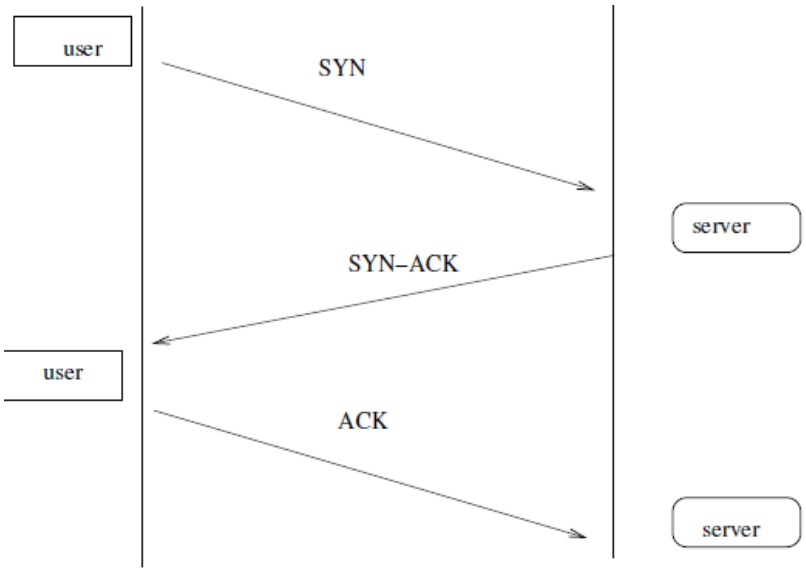
Figure 1: Lab environment configuration example

CITS 3004 Cybersecurity

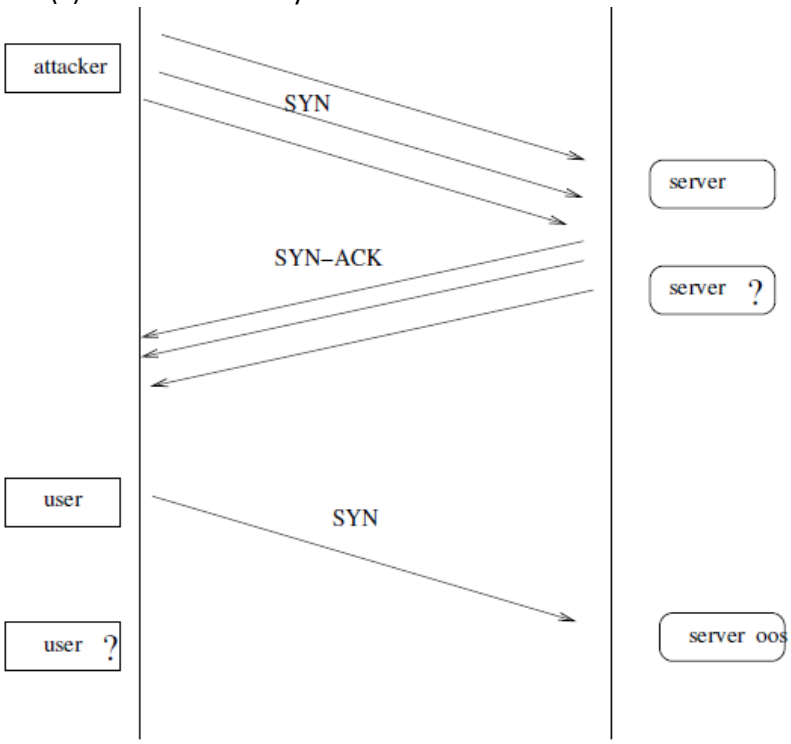
THE UNIVERSITY OF WESTERN AUSTRALIA

1A) 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 got a final ACK back. When this queue is full, the victim cannot take any more connection. Figure 2 illustrates the attack.



(a) Normal TCP 3 way handshake between user and server



(b) SYN flood: attacker sends many SYN to server without ACK.

Figure 2: Normal TCP handshake and SYN Flooding Attack

CITS 3004
Cybersecurity

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

```
# sysctl -q net.ipv4.tcp_max_syn_backlog
```

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 (use tool 76) to conduct the attack, and then use a sniffer tool (e.g., wireshark) to capture the attacking packets. While the attack is ongoing, run the "netstat -na" command on the victim machine, and compare the result with that before the attack. Run your attacks with the SYN cookie mechanism on and off, and compare the results.

Here is a simple help screen for the Netwox tool 76. You can also type netwox 76 --help to get the help information.

```
Title: Synflood
   Usage: netwox 76 -i ip -p port [-s spoofip]
   Parameters:
   -i|--dst-ip ip              destination IP address
   -p|--dst-port port          destination port number
   -s|--spoofip spoofip        IP spoof initialization type
```

You should also experiment with different TCP buffer queue size (default is 1024):

```
# sysctl -q net.ipv4.tcp_max_syn_backlog=\{VALUE\}
```

SYN Cookie Countermeasure: If your attack seems unsuccessful, one thing that you can investigate is whether the SYN cookie mechanism is turned on. SYN cookie is a defense mechanism to counter the SYN flooding attack. The mechanism will kick in if the machine detects that it is under the SYN flooding attack. You can use the sysctl command to turn on/off the SYN cookie mechanism:

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

Answer Q1 ~ Q5 on LMS

CITS 3004
Cybersecurity

THE UNIVERSITY OF WESTERN AUSTRALIA

1B) TCP RST ATTACKS ON TELNET AND SSH 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 B and C, attackers can spoof a RST packet from B to C to break this connection. To succeed in this attack, attackers need to correctly construct the TCP RST packet. In this task, you need to launch a TCP RST attack to break an existing `telnet` connection. After that, try the same attack on an `ssh` connection. To simplify the lab, we assume that the attackers and the victims are on the same LAN, i.e., attackers can observe the TCP traffic between B and C.

If you are using 3 VMs, you can use 2 of them to establish a telnet connection. For instance, we have VMs A (attacker, 10.0.2.4), B (user, 10.0.2.10) and C (user, 10.0.2.11). Then, user B will initiate the netcat with the command `nc -l 1234`. Then, user C runs the command `telnet 10.0.2.10 1234`. If you are only using 2 VMs, the target VM can try `telnet scn.org`. Note, the connection is terminated after 60 seconds.

You will use netwox tool 78 for this task. Here is a simple help screen for this tool. You can also type `netwox 78 –help` to get the help information.

```
Title: Reset every TCP packet
   Usage: netwox 78 [-d device] [-f filter] [-s spoofip]
   Parameters:
   -d|--device device          device name {Eth0}
   -f|--filter filter          pcap filter
   -s|--spoofip spoofip        IP spoof initialization type {linkbraw}
```

TCP RST Attacks on Video Streaming applications

Let us make the TCP RST attack more interesting by experimenting it on the applications that are widely used in nowadays. We choose the video streaming application in this task. For this task, you can choose a video streaming web site that you are familiar with (we will not name any specific web site here). Most of video sharing websites establish a TCP connection with the client for streaming the video content. The attacker's goal is to disrupt the TCP session established between the victim and video streaming machine. To simplify the lab, we assume that the attacker and the victim are on the same LAN. In the following, we describe the common interaction between a user (the victim) and some video-streaming web site:

- The victim browses for a video content from a web site, and selects one of the videos for streaming.
- Normally video contents are hosted by a different machine, where all the video contents are located. After the victim selects a video, a TCP session will be established between the victim machine and the content server for the video streaming. The victim can then view the video he/she has selected.

Your task is to disrupt the video streaming by breaking the TCP connection between the victim and the content server. You can let the victim user browse the video-streaming site from another (virtual) machine or from the same (virtual) machine as the attacker. Please be noted that, to avoid liability issues, any attacking packets should be targeted at the victim machine (which is the machine run by yourself), not the content server machine (which does not belong to you). You only need to use Netwox for this task.

Answer Q6 ~ Q9 on LMS

CITS 3004
Cybersecurity

THE UNIVERSITY OF WESTERN AUSTRALIA

1C) (OPTIONAL) TCP SESSION HIJACKING

The objective of the TCP Session Hijacking attack is to hijack an existing TCP connection (session) between two victims by injecting malicious contents into this session. If this connection is a `telnet` session, attackers can inject malicious commands into this session, causing the victims to execute the malicious commands. We will use `telnet` in this task. We also assume that the attackers and the victims are on the same LAN.

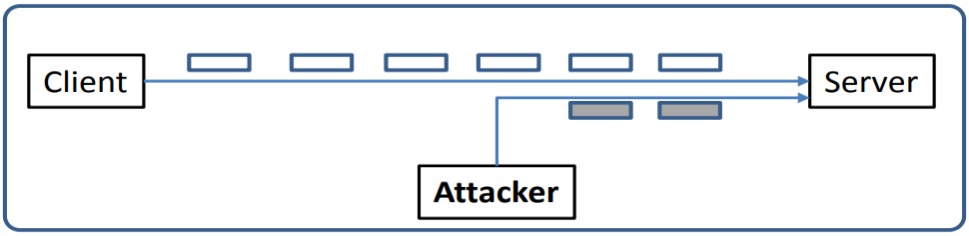


Figure 3: TCP Session Hijacking Attack

You will use netwox tool 40 for this task. Here is part of the manual for this tool. You can also type `netwox 40 –help` to get the help information. In addition, you may also need to use Wireshark to find out the correct parameters for building the spoofed TCP packet. The fields that are not set will use the default value provided by `netwox`. You should note that some of the default values do not work for Ubuntu 16.04. So ensure that you specify all the fields necessary for this attack.

```
Title: Spoof Ip4Tcp packet
   Usage: netwox 40 [parameters …]
   Parameters:
   -l|--ip4-src ip                  Source IP
   -m|--ip4-dst ip                  Destination IP filter
   -j|--ip4-sttl unit32             Time to live
   -o|--tcp-src port                TCP Source port number
   -p|--tcp-dst port                TCP Destination port number
   -q|--tcp-seqnum unit32           TCP sequence number
   -E|--tcp-window unit32           TCP window size
   -r|--tcp-acknum unit32           TCP acknowledge number
   -z|--tcp-ack|+z|--no-tcp-ack     TCP ack bit
   -H|--tcp-data data               TCP data
```

Note1: You should be aware that when Wireshark displays the TCP sequence number, by default, it displays the relative sequence number, which equals to the actual sequence number minus the initial sequence number. If you want to see the actual sequence number in a packet, you need to right click the TCP section of the Wireshark output, and select `"Protocol Preference"`. In the popup window, uncheck the `"Relative Sequence Number and Window Scaling"` option.

Note2: In the `netwox` command above, the `tcp-data` part only takes hex data. If we want to inject a command string, we need to convert it into a hex string (from ASCII). A very simple command in Python can do this for you. For example, we can convert an ASCII string `"Hello World"` to a hex string:

```
>>> "Hello World".encode("Hex")
```