

## Lab Exercise 7 – Binary Analysis, Firewall, and Intrusion Detection

Due Date: November 11, 2022 11:59pm

Points Possible: 7

NAME: Jingtao Scott Hong jh4ctf

### 1. Overview

This lab exercise will provide some hands-on experience with binary analysis, firewall configuration, and intrusion detection.

### 2. Resources required

This exercise requires Kali Linux VM running in the Virginia Cyber Range.

### 3. Initial Setup

From your Virginia Cyber Range course, select the **Ubuntu with Snort and Other Tools** environment. Click “start” to start your environment and “join” to get to your Linux desktop login. This environment requires authentication. Log in using these credentials:

Username: **student**

Password: **student**

Once you are logged in, click the Terminal Emulator in the bottom menu to open the command line.

### 4. Tasks

#### Task 1: Binary Analysis

Run the file **game** and win the game. You can use the various static and dynamic tools and a little fuzzing to determine the best way to trick the game and win. The file is already located on the Desktop in Cyber Range but you may need to install the analysis tools you want to use. If you want to use your own system you can download the game file here:

[bit.ly/3Cc5QEO](https://bit.ly/3Cc5QEO)

**Question 1: Provide a screenshot of what you entered to win the game. (.5 point)**

```
student@ip-10-1-114-15:~/Desktop$ ./game
Beat the house and win! Can you get to $1M without going bust?
You begin with $1K
Your pot is 1000

Enter integer bet: -99999999
The house rolls 11

Press [enter] to roll
You roll 10
Bad luck. You lose.
You won!
Access Granted!
flag:theresaneasywayandahardway.
student@ip-10-1-114-15:~/Desktop$
```

## Task 2: Firewall Configuration

[Note: be very careful with firewall rule configuration changes on your Cyber Range virtual machine. If you set the rules improperly you could break your network connection to the range VM. Fortunately, this can almost always be fixed by restarting your VM. If that happens, go to the Virginia Cyber Range page and select the “Stop Exercise” button for this lab, then restart the exercise and re-join.]

Use the following command to set the host-based firewall on your Linux system to a default policy that we have specified:

```
$ sudo /etc/default_firewall.sh
```

[When using **sudo** you may need to enter your student password: **student**]

Linux host-based firewalls are configured using the **iptables** command. There is a pretty good (and short) tutorial at <http://fideloper.com/iptables-tutorial>. To review the firewall rules set by the default policy, use the following command (you may want to use the mouse to drag the terminal screen wider first to read the output more clearly with no linebreaks):

```
$ sudo iptables -L -n
```

Simple packet filtering firewalls usually have a default policy to DROP (deny) packets and only to accept traffic that meets specific criteria. When a packet arrives on a host, the firewall tries to match firewall rules starting with the first rule in the chain. The firewall will apply the first rule that matches and the default rule is applied last, so if there is a rule that “ACCEPTs” a packet before the default DROP, the packet will be accepted. In general, if a specific input or output IP address, port, or protocol is not specified, the rule applies to 'any' IP address, port, or protocol.

Review the default firewall configuration (**\$ sudo iptables -L -n**) and answer questions 1 – 3.

**Question 2: What is the default policy on the INPUT, FORWARD, and OUTPUT chains in the default firewall configuration and what does this mean for each? (.5 point)**



```
student@ip-10-1-114-15:~$ sudo iptables -L -n
[sudo] password for student:
Chain INPUT (policy DROP)
target     prot opt source                destination
ACCEPT     icmp -- 0.0.0.0/0              0.0.0.0/0
ACCEPT     udp  -- 0.0.0.0/0              0.0.0.0/0
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0          tcp dpt:22 state N
NEW,ESTABLISHED
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0          tcp dpt:3389 state
NEW,ESTABLISHED
ACCEPT     all  -- 0.0.0.0/0              0.0.0.0/0

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0          tcp dpt:22 state E
ESTABLISHED
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0          tcp dpt:3389 state
ESTABLISHED
ACCEPT     all  -- 0.0.0.0/0              0.0.0.0/0
```

For INPUT, policy is DROP, for FORWARD and OUTPUT is ACCEPT. For DROP policy, it means that the system will check if there are any matching rules that fit to the existing rules. If not, then disregard the traffic and leave with no responses. For ACCEPT policy, it means that system will check if there are any matching rules that fit to the existing rules and allow the traffic to go through.

**Question 3:** What specific firewall rules are in place on the INPUT chain? Specify protocols and ports for which packets are allowed by the rules provided, and under what conditions those packets are allowed. (Hint: there are 5 rules) (.5 point)

For INPUT chain, protocol ICMP and UDP, all traffic is allowed regardless of the source or destination. For protocol TCP, destination 22 and 3389 will be allowed with the packet of NEW ESTABLISHED. And for all protocols will be accepted regardless of the source and destination.

**Question 4:** You notice a big problem with the firewall rules on the INPUT chain. What is it? (.5 point)

Because the last rule of INPUT is to allow all Protocols through traffic regardless of the source and destination, there is no point of specifying all the rules above for the traffics and protocols.

**Question 5:** What firewall rules are in place on the OUTPUT chain? Specify protocols and ports for which packets are allowed by the rules provided, and under what conditions those packets are allowed. (.5 point)

For protocol TCP, dpt 22 and 3389 will be ACCEPT with the packet of ESTABLISHED. And for all protocols will be accepted regardless of the source and destination.



We will use two shell scripts to modify the firewall configuration. A script called '/etc/extingui.sh' will clear all firewall rules and set the default policy on the INPUT, OUTPUT, and FORWARD chains to ALLOW all traffic in and out of your server. Execute this script as follows.

```
$ sudo /etc/extingui.sh
```

Perform the following command again to see that the firewall rules are cleared:

```
$ sudo iptables -L -n
```

Once the firewall rules are cleared, you will modify the script '/home/student/lab2/firewall.sh' to add firewall configuration commands. This file is not a blank file, it already has a firewall rules template in it. Use the text editor of your choice to edit this script (one option is "mousepad").

```
$ mousepad /home/student/lab2/firewall.sh
```

Iptables commands are of the following form:

```
iptables [command-type] [pattern-match options] -j [action]
```

Where [command-type] specifies whether the rule will be added or deleted on a specified chain, [pattern-match-options] specifies the port, interface, address, etc. to match, and [action] specifies what action to take if the packet matches the pattern (DROP, REJECT, ACCEPT, LOG).

In our simple packet filtering firewall, all of our rules will be added to the INPUT or OUTPUT chains and our actions will either be ACCEPT or DROP; so in this exercise, all of your rules will be of the form:

```
iptables -A INPUT [pattern-match options] -j [ACCEPT or DROP]
```

Pattern match options that you will use include:

-s	source IP address or address range (can use CIDR addressing)
-d	destination IP address or address range
-p	transport layer protocol (tcp, udp, or icmp)
-m	match a specific property (such as 'state')
--dport	destination port number (must be used with a protocol specified by the -p option)
--sport	source port number (must be used with a protocol specified by the -p option)
--state	connection state (NEW, ESTABLISHED, etc.)

An example rule using the above options is here:

```
# Allow inbound packets to TCP port 20 from subnet 192.168.1.0/24  
iptables -A INPUT -s 192.168.1.0/24 -p tcp --dport 20 -j ACCEPT
```



**Add rules to your `/home/student/lab2/firewall.sh` script that will allow outbound connection attempts on port 80 and the return traffic.** Be very specific with your rules and include state, don't just allow all.

Once you have edited and saved the `firewall.sh` file, apply at the command line as follows:

```
$ sudo /home/student/lab2/firewall.sh
```

Perform the following command again to see that the firewall rules are cleared:

```
$ sudo iptables -L -n
```

**Question 6: List the rule(s) that you added to the `firewall.sh` to allow outbound HTTP requests (port 80) and responses. (1 point)**

```
iptables -A INPUT -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p tcp --dport 80 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
student@ip-10-1-114-15:~$ sudo iptables -L -n
Chain INPUT (policy DROP)
target     prot opt source                destination
ACCEPT     icmp -- 0.0.0.0/0              0.0.0.0/0
ACCEPT     udp  -- 0.0.0.0/0              0.0.0.0/0
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0      tcp dpt:3389 state
NEW,ESTABLISHED
ACCEPT     all  -- 0.0.0.0/0              0.0.0.0/0
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0      tcp spt:80 state E
STABLISHED

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0      tcp dpt:3389 state
ESTABLISHED
ACCEPT     all  -- 0.0.0.0/0              0.0.0.0/0
ACCEPT     tcp  -- 0.0.0.0/0              0.0.0.0/0      tcp dpt:80 state N
EW,ESTABLISHED
student@ip-10-1-114-15:~$
```

## Task 2: Intrusion Detection

Your Virginia Cyber Range virtual machine has Snort software installed for intrusion detection. Instead of observing traffic from a network interface, we will use Snort to process packet capture files (.pcap files) from previously captured traffic.

Here is a great Snort reference from the Snort creator:

[https://paginas.fe.up.pt/~mgi98020/pgr/writing\\_snort\\_rules.htm](https://paginas.fe.up.pt/~mgi98020/pgr/writing_snort_rules.htm)



Before we run Snort against captured packets, we'll take a look at some snort rules (signatures) in the `/etc/snort/rules` directory. To do this, open a terminal window and change to the appropriate directory as follows.

```
$ cd /etc/snort/rules
$ ls
```

← this will list all the rule files

Examine the file **shellcode.rules** using the text editor of your choice (your Linux VM includes *vi* and *nano*, as well as a GUI text editor called *mousepad* as shown in the command below. You could also use the *cat* or *more* command).

```
$ mousepad shellcode.rules &
```

Each rule has a unique Snort ID number (sid), which is included in the signature. In *shellcode.rules*, find **sid:648** amongst the rules in that file and answer the following questions.

**Question 7: What is the specific signature (content) that sid:648 tries to match? (.5 point)**

```
|ORTS -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"[90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90]"; depth:128; reference:arachnids,181; classtype:shellcode-detect; sid:648; rev:7;)
```

|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|

**Question 8: What action is Snort supposed to take if the signature contained in sid:648 is matched? (.5 point)**

It will alert the user with its message: "SHELLCODE x86 NOOP".

```
|ORTS -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"[90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90]"; depth:128; reference:arachnids,181; classtype:shellcode-detect; sid:648; rev:7;)
```

Change directories to `/home/student/lab2` and run **snort** against the packet capture file called **theft.pcap** in that directory as shown here.

```
$ sudo snort -c /etc/snort/snort.conf -r theft.pcap
```

[When using **sudo** you may need to enter your student password: **student**]

You will see Snort processing on your screen, let it complete and return on the command prompt \$.

When Snort finishes processing, open a web browser on your Cyber Range virtual machine (on the menu bar at the bottom of the screen) It will automatically open the following URL: North wall (detail), Diego Rivera, Detroit Industry murals, 1932-33

BASE is the Basic Analysis and Security Engine, which is installed on your virtual machine along with Snort. It allows you to view Snort alerts in a nice graphical format. Log in to BASE using the username: **john** and the password: **secret3**. The homepage should show the results of the scan we just processed using Snort, with a little over 27,000 Total Number of Alerts. [If you aren't seeing "Total Number of Alerts: 27081, It takes a few minutes for the back-end alert processing to complete, so you might have to refresh the page a few times.] NOTE: If you run the command with *theft.pcap* more than once you will have double or triple the alerts, **so only run it once**.



Review the alerts (you might have to do some filtering) and answer the following questions.

**Question 9:** An attacker was trying to steal a specific, and very sensitive, file from the target system. What file was she after? (Hint: Click on the Total Number of Alerts, then click on Unique Alerts to filter the information) (.5 point)

**/etc/passwd**

**Question 10:** What is the technique that the attacker was trying to use to steal the file? (1 point)

**They used http directory traversal**

**Question 11:** What is the source IP address of the attacker that is trying to steal the file from the system? (1 point)

**92.60.73.14**

*By submitting this assignment you are digitally signing the honor code, "I pledge that I have neither given nor received help on this assignment".*

**END OF EXERCISE**

---

## References

- iptables man page: <https://linux.die.net/man/8/iptables>
- iptables tutorial: <https://fideloper.com/iptables-tutorial>
- Snort: <https://www.snort.org/>
- Writing Snort Rules: [https://paginas.fe.up.pt/~mgi98020/pgr/writing\\_snort\\_rules.htm](https://paginas.fe.up.pt/~mgi98020/pgr/writing_snort_rules.htm)
- BASE: <https://lwn.net/Articles/112548/>

