CSSE 490 -- Network Security

Rose-Hulman Institute of Technology

Lab 4: Port Knocking

Learning Objectives

At the end of this lab, you should be able to:

- Define `nftables` sets and how they can manipulated.
- Define port knocking as a way to hide certain ports behind a firewall.
- Implement a simple port knocking firewall.
- Implement a more involved sequence of port knocking that mixes up TCP and UDP ports.

Name: ______________________________

Question	Points	Score
Question 1	5	
Question 2	5	
Question 3	5	
Question 4	5	
Question 5	5	
Question 6	10	
Question 7	5	
Question 8	5	
Question 9	5	
Question 10	5	
Question 11	5	
Question 12	10	
Question 13	10	
Question 14	15	
Question 15	5	
Total:	100	

1 Experiment 0

The questions below to `Experiment 0` in the lab instructions.

Question 1. (5 points) Why is the firewall rule preventing the client from successfully pinging the server?

Hint: If you're struggling with this one, you might find it useful to start a packet capture session on the server and the firewall and see where the packets are being dropped.

Question 2. (5 points) Explain by referencing the logs what seems to be the bug in the current set of rules in the `firewall` chain.

Question 3. (5 points) Suggest a way to fix the rules in the `firewall` chain so that the two-way communication between the client and the server can complete.

2 Experiment 1

The questions below to `Experiment 1` in the lab instructions.

Question 4. (5 points) Does the ping packet get delivered to the server?

Question 5. (5 points) Does the ping packet get added to the counter in the `icmp_chain`?

Question 6. (10 points) Explain the difference between a `goto` to a chain and `jump` to a chain.

No, it is not that `goto` drops the packets and `jump` accepts them.

Hint: There are two ways for you to answer this question:

1. Trace the rules in this table using the debugging techniques from above and understand where each packet travels.
2. Add a counter to the second rule (`ip protocol icmp accept`) and then check which counters get updated with `jump` vs with `goto`. Then, change the `firewall` chain's default policy to `drop` and try again and report on your observations.

2.1 Experiment 1: Back to set updates

The questions below refer to the last step of experiment 1.

Install your table in the firewall and then attempt to start a `telnet` connection from the client to the server (`telnet server` from the client container).

Question 7. (5 points) Should you be able to establish a `telnet` connection between the client and the server?

Question 8. (5 points) If your answer to the question above is no, what would you need to do to allow the client to talk to the server over `telnet`?

After you are able to allow the client to talk to the server, establish the `telnet` connection and answer the following questions:

Question 9. (5 points) How long do you expect the `telnet` connection to last? In other words, what will happen to the `telnet` connection after 30 seconds?

To help in answering that question, have the client container issue an ICMP echo request every 5 seconds to the sever. You can do so using the `-i` flag of `ping` as follows: `ping -i 5 server`. During this time, monitor the content of the `allowed_ip` set in the table using `nft list table e1s1`.

Question 10. (5 points) What do you notice about the entry for the client's IP address in the `allowed_ip` set? What does that tell you about the behavior of the `add` operation in the `add_to_set` chain?

Now replace the `add @allowed_ip { ip saddr timeout 30s }` with `update @allowed_ip { ip saddr timeout 30s }` and then rerun the above exercise.

Question 11. (5 points) What do you notice about the behavior of `update` vs that of `add`?

Finally, answer the following conceptual questions:

Question 12. (10 points) What would happen if we had replaced the `jump add_to_set` action with `goto add_to_set` in the `firewall` chain? Explain your answer.

Question 13. (10 points) What would happen if we swap the order of the last two rules in the `firewall` chain? i.e., our chain would look like:

```
  ip saddr @allowed_ip counter accept
  ip protocol icmp jump add_to_set

```

3 Experiment 2

The question below refers to the conceptual question in the last step of `experiment 2`.

Question 14. (15 points) Before you write down the script for your rules, on your question sheet, please draw a *finite state machine* that represents the possible states that your firewall might be in when receiving packets.

4 Reflection

In this lab, we have used port knocking as a way to make sure that our users can authenticate to the firewall so that the firewall can unlock certain ports for them on the protected network.

Question 15. (5 points) In the space below, think about possible ways in which this approach can be broken down. There are two major limitations with this approach that we'd like to tackle in the next set of concept labs and labs.