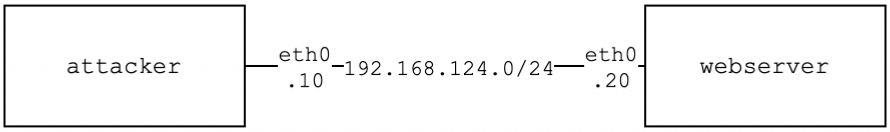
Network Security

Assignment: Transport Layer

SYN-flooding [40 points]

In the topology in the image below, the webserver is vulnerable to the SYN-flooding attack.



Abuse this configured vulnerability to make the web server that's running on port 80 unavailable. Before starting with the assignment, make sure to unzip the `syn_flooding.zip` file and afterward navigate to the unzipped directory. Generate the environment from the image above using the following Docker commands.

```
docker-compose build
docker-compose up -d
```

To connect to the attacker container, execute the command below.

```
docker exec -ti attacker bash
```

After finishing the assignment, you can destroy the environment using the command below. First, make sure to copy your developed files outside the container. Otherwise, your assignment will be lost.

```
docker-compose down
```

Goal

Produce a Python3 script called `syn_flood.py`. The program should accept three arguments: the destination address, the destination port, and the number of half-open connections you want to generate. The script should be invoked as follows:

```
python3 syn_flood.py 192.168.124.20 80 200
```

The script itself shouldn't generate any output. However, the assignment is successful if you don't get any response when executing the command below.

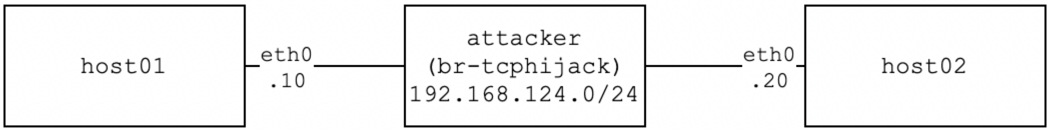
```
curl 192.168.124.20
```

Submission Instructions

Your script should run on the attacker container. After completing the assignment, you should only submit the resulting script. A `requirements.txt` file should be provided if you use any modules from the Python Package Index (pip).

TCP-hijacking [60 points]

Consider the topology in the image below.



We first unzip the `tcp_hijack.zip` archive. Afterward, we navigate to the unzipped directory. Next, we execute the commands below to fire up the environment.

```
docker-compose build
docker-compose up -d
```

To connect to the attacker container, execute the command below.

```
docker exec -ti attacker bash
```

Before starting with the actual assignment, we first have to set up a TCP connection that we will hijack. On host02, we already have a Netcat listener running, which is connected to a bash shell. Connect to the host01 container and execute the command below to connect to host02.

```
nc 192.168.124.20 1337
```

To generate traffic, you can execute shell commands like `ls` from host01 inside the Netcat session.

After finishing the assignment, you can destroy the environment using the command below. First, make sure to copy your developed files outside the container. Otherwise, your assignment will be lost.

```
docker-compose down
```

Goal

Write a Python3 script that's called `hijack_tcp.py`. This script accepts two arguments: the source IP address, and destination IP address TCP connection you're hijacking. The way the script should be called is shown below.

```
python3 hijack_tcp.py 192.168.124.10 192.168.124.20
```

The script should inspect the traffic and find the appropriate connection to hijack by itself. You can generate traffic by executing shell commands from host01 on host02.

```
tcpdump -i br-tcphijack
```

This assignment is considered successful when you can hijack the TCP connection over which shell commands can be executed. As proof of a successful hijack your script should create a directory called `owned` inside the `\root\` directory. Furthermore, you should also setup a reverse shell back to the attacker container. Your script should implement a reverse shell listener. The listener needs to enable interactive execution of shell commands on host02.

Submission Instructions

Your script should run on the attacker container. After completing the assignment, you should only submit the resulting script. A `requirements.txt` file should be provided if you use any modules from the Python Package Index (pip).

Make sure your assignment conforms to the in- and output described in the Goal subsection. Your submission is partly graded through an automated system. Any deviations from the described in- and output can affect your grade.