Network Security Lab 2

Jordi Nadeu Ferran

1 Introduction

In this laboratory involves setting up a controlled network security environment to prevent an attacker from reaching a web service.

In summary:

1. **Honeypot** container:Simulates an SSH service using Cowrie to capture attacker activity and log attempts.
2. **Web service with IDS** container: A container running a web server and SSH service, equipped with Snort to detect SSH brute force attacks.
3. **Attacker** container: A Debian container used to perform reconnaissance with Nmap and brute force attacks with Hydra and passwords list.

The structure of folders and files is as follows:

```
web-service/
    +-- Dockerfile
attacker/
    +-- Dockerfile
    +-- passwords.txt
```

2 Build and run Docker containers

First of all we will create a docker network to interconnect the containers, to do this we need to execute the following command:

```
docker network create lab-net
```

With the following command create and run the Honeypot container:

```
docker run -d --network lab-net --name honeypot -p 2222:2222/tcp cowrie/cowrie
```

To check if all goes correctly, execute:

```
docker logs honeypot
```

The output if all was well, would be:

```
[-] Ready to accept SSH connections
```

Then for build and run the web-service container do the following commands executed inside the project folder:

```
docker build -t ubuntu-server ./web-service
```

```
docker run -d --network lab-net --name web-service --cap-add=NET_ADMIN
-p 22:22 ubuntu-server
```

```
docker exec -it web-service bash
```

Once we are inside the web service container, we can start the IDS (snort) running the following command:

```
snort -A console -q -i eth0 -c /etc/snort/snort.conf
```

And finally the last container we will open another terminal and follow the commands to create and run the attacker container where we will perform the attack later:

```
docker build -t debian-attacker ./attacker
```

```
docker run -it --rm --network lab-net --name attacker debian-attacker
```

(Extra) To identify the IP addresses of the web services and the honeypot containers, we can use the following command:

```
docker network inspect lab-net
```

Figure 1: Output of docker network inspect command

3 How to block the attacker

To protect the web service container from the attacker container we can use a firewall to block the IP address of the attacker.

We can do that with the following command inside the web service container:

```
ufw deny from <attacker-ip> && ufw enable
```

Remember to conect with the web service container and can execute commands inside we can do:

```
docker exec -it web-service bash
```

Also we can use anothers tools, like Fail2ban for example.

4 How would you warn others?

To warn peers in another organization about a threat, such as a detected brute force SSH attack, follow a structured and professional approach.

First of all, before warning others, compile detailed and accurate information about the threat. The details of the threat would be:

- The type of the attack, in this case a SSH brute force attack.
- The targeted service or application, in this case the SSH on port 22.
- Indicators of Compromise (IoCs) such as attacker IP addresses (172.18.0.4) and other information that we have about the attacker.

We also have to report the impact if the attack succeeds or not. And also the detection in this case we have detected for the Snort and honeypot logs.