Report on
CTF: Code Crusade - Conquer the Digital Realm

Network Security (CS6903) - Assignment 9
Dept. of Computer Science And Engineering
Indian Institute of Technology, Hyderabad

Submitted By Team: Halloween hackers

APPROACH FOR CAPTURING FLAG #1

We began by using open-source network scanning software, Nmap (Network Mapper)[1], to identify open ports on the target system. Nmap allowed us to efficiently enumerate the network and discover potential entry points for further exploration.

Nmap revealed several open ports as shown in the *Figure 1* below including *22, 3890, 3900, 3910, 3920, 3930, 3940.*

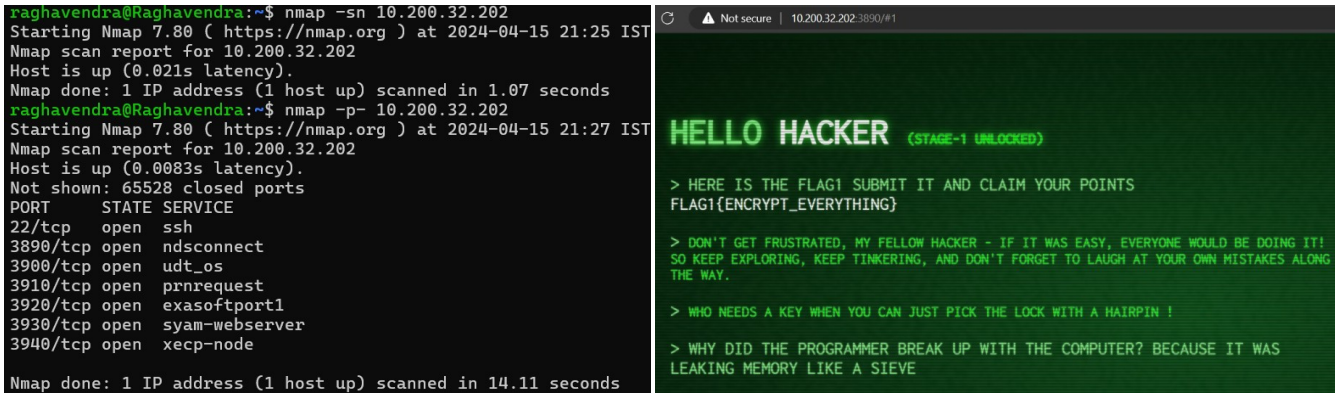

Figure 1. Nmap scan results for open port and the first flag

We used the following command with `nmap` to identify the hosts that were up and running:

```
$ nmap -sn 10.200.32.202
```

This command allowed us to perform a ping scan on the specified IP address `10.200.32.202` to determine which hosts were reachable on the network.
We used the following command with `nmap` to conduct a comprehensive port scan on the designated target VM whose IP address is 10.200.32.202, aiming to look for all open ports and active services:

```
$ nmap -p- 10.200.32.202
```

This command facilitated the scanning of all available ports on the specified IP address, allowing us to identify active services and assess potential security risks.

After individually examining the open ports, we discovered our initial flag at `http://10.200.32.202:3890/#1`. Additionally, the first flag page provided us with clues hinting at the locations of other flags to conquer.

APPROACH FOR CAPTURING FLAG #2

We initiated our reconnaissance phase by employing DIRB[2], an open-source web content scanner, to identify accessible directories and files on the target web server. DIRB[2] proved invaluable in swiftly enumerating potential entry points, thereby facilitating further exploration of the target environment.

Utilizing DIRB yielded a comprehensive list of directories and files, as illustrated in *Figure 2* below, resulting in the identification of the `'/granted'` directory with a HTTP response code of 200 and a size of 4371 bytes. We found our flag at `http://10.200.32.202:3890/granted`

We used the following command which utilizes a common list of words often employed by pentesters provided by DIRB.

```
$ dirb http://10.200.32.202:3890
```

This allowed us to conduct a directory brute-force attack on the specified URL, seeking potential hidden directories or files.

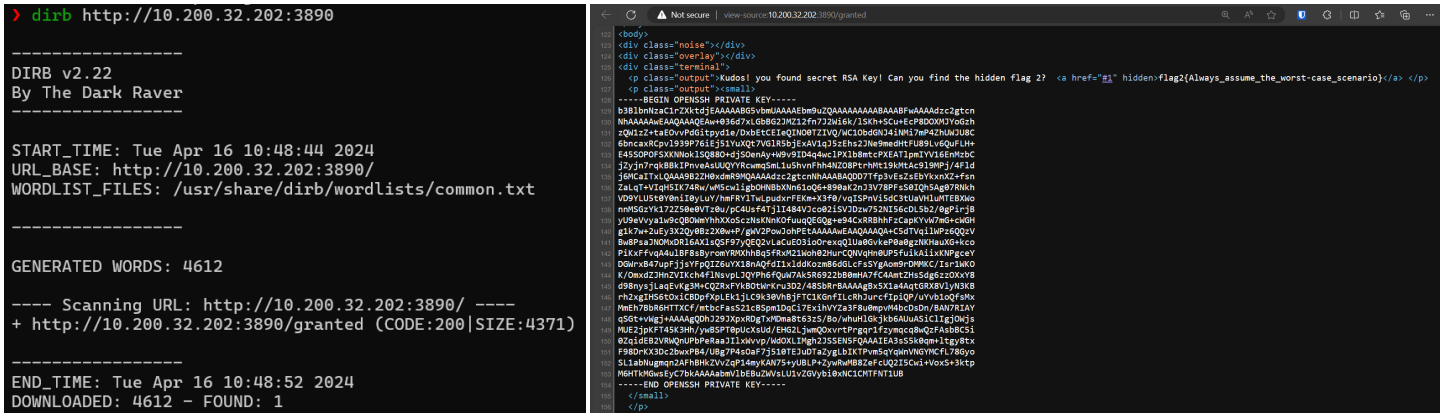

Figure 2. DIRB scan results showing the discovery of directories on the target VM and the second flag

After identifying the OPENSSH RSA PRIVATE KEY in the same site source, we securely saved it into a file named `ctf_halloweenhackers_private_key.key` for further analysis and potential exploitation.

APPROACH FOR CAPTURING FLAG #3

We utilized the OPENSSH RSA PRIVATE KEY (Shown in Figure 3 below) saved in the second capture to gain access to target VM with the help of the following command:

```
$ ssh -i ctf_halloweenhackers_private_key.key ns@10.200.32.202
```

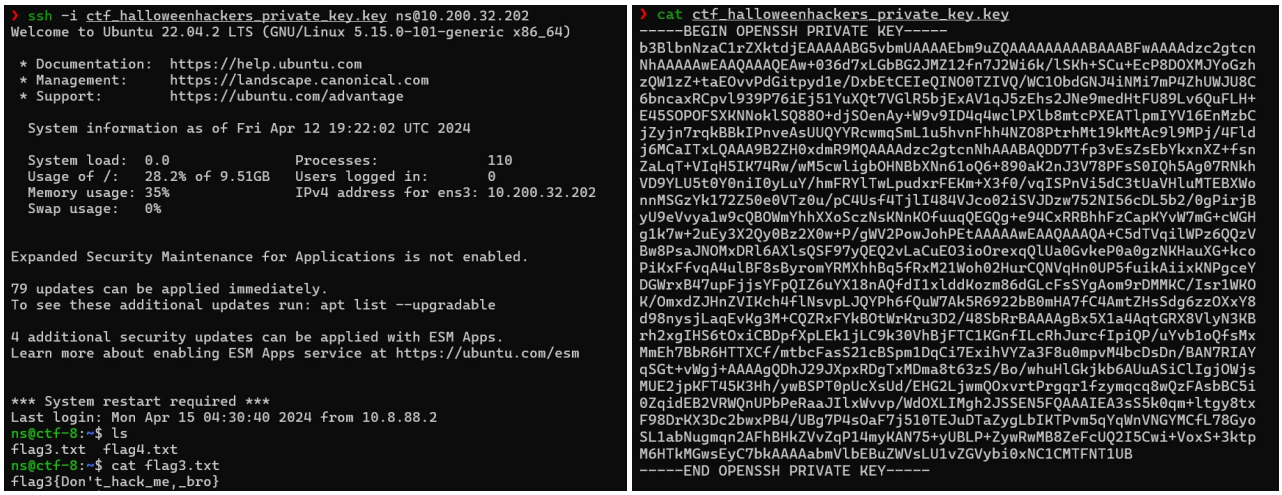

Figure 3. The third flag and Private Key File

We can see two files inside the VM `flag3.txt` and `flag4.txt`. We access the file `flag3.txt` using:

```
$ cat flag3.txt
```

But, wait! Why can't I access the `flag4.txt` just like I got to `flag3.txt`? Ohhh, its owner's username is "`hacker`" and we are logged in as "`ns`". We will need to gain access to VM with user id "`hacker`" instead of "`ns`".

"WHO NEEDS A KEY WHEN YOU CAN JUST PICK THE LOCK WITH A HAIRPIN !"

APPROACH FOR CAPTURING FLAG #4

"WHY DID THE PROGRAMMER BREAK UP WITH THE COMPUTER? BECAUSE IT WAS LEAKING MEMORY LIKE A SIEVE"

From the above hint, it reminded me of *Openssl Heartbleed* [5] attack which does buffer overflow for leaking memory.

We utilize the following command to initiate a Metasploit Framework console to carry out the Heartbleed Attack[5].

```
$ msfconsole
```

Then, we first search for anything related to heartbleed using the following command and type use command accordingly:

```
msf6 > search heartbleed
msf6 > use auxiliary/scanner/ssl/openssl_heartbleed
```

Now, let's use the Remote Host and Port → Vulnerable Machine IP and Port using the following command:

```
msf6 auxiliary(scanner/ssl/openssl_heartbleed) > set RHOST 10.200.32.202
msf6 auxiliary(scanner/ssl/openssl_heartbleed) > set RPORT 3900
```

Now, we enable the verbose and initiate the attack using the following command:

```
msf6 auxiliary(scanner/ssl/openssl_heartbleed) > set verbose true
msf6 auxiliary(scanner/ssl/openssl_heartbleed) > run
```

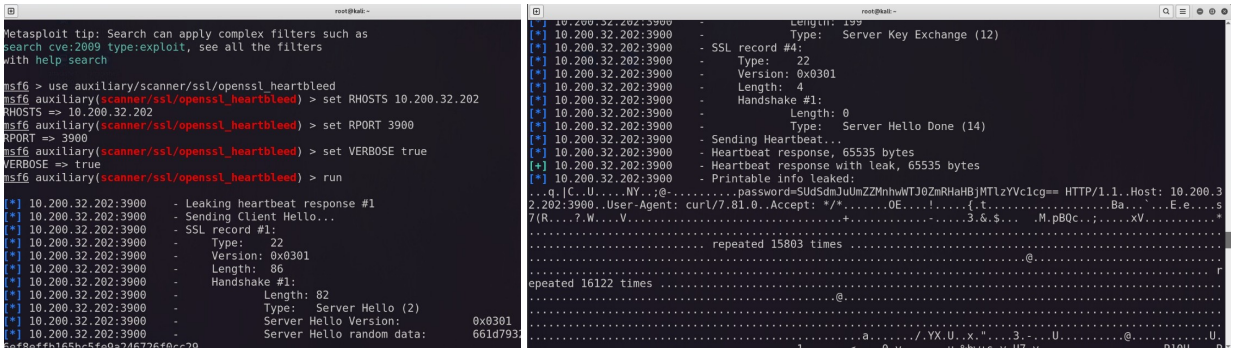

Figure 4. Openssl Heartbleed Attack in action in metasploit console

We carried out a Heartbleed attack to exploit an OpenSSL vulnerability, obtaining an encrypted password. After two decoding steps from Base64, we decrypted the password associated with the user "`hacker`".
Now, we sign in to the VM using the following command and get the flag:

```
$ ssh hacker@10.200.32.202
Password: dont_click_this_link
$ cat /home/ns/flag4.txt
```

```
hacker@ctf-8:/$ cat /home/ns/flag4.txt
flag4{I've_got_99_problems,_but_a_breach_ain't_one}
hacker@ctf-8:/$
```

Figure 5. The fourth flag

APPROACH FOR CAPTURING FLAG #5

We inspected the source code of the site with port number `3910`. It revealed the Username and Password check hidden in Base64 Encoded Plain Text. By decoding it using Base64 Decoder, we got the original ID as `cyber_ninja` and Password as `stealth_mode_on` . It's a kind of XSS or CSRF type attack.

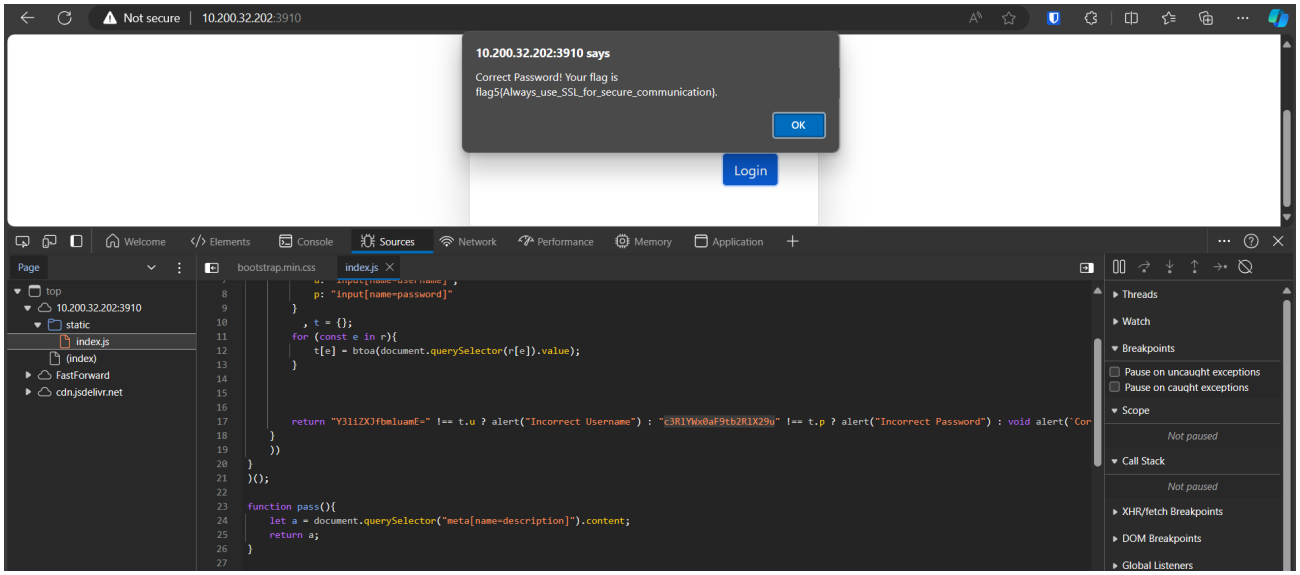

Figure 6. The fifth flag

APPROACH FOR CAPTURING FLAG #6

We inspected the source code of the site with port number `3920` and found nothing special. Then we created a random user by using the Sign up option available. Once signed up, we went back to login page and started the Burp Suite Proxy Interceptor. It revealed the cookie check for Admin which was set to false. We just set it to true and forwarded the packet to server and got access to the admin account and our sixth flag as shown in Figure 7 below.

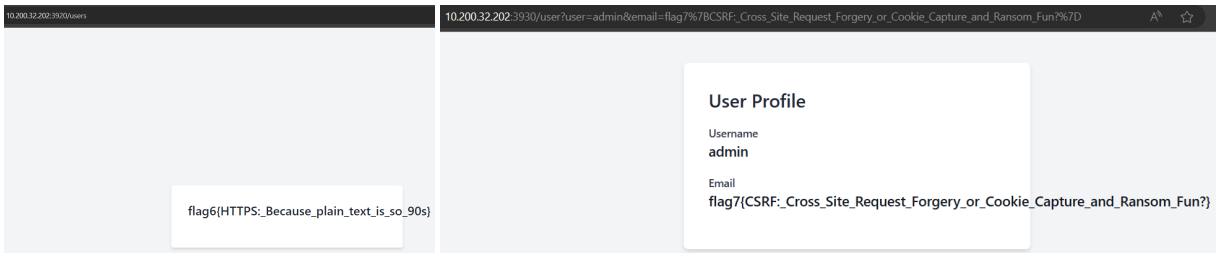

Figure 7. The sixth and seventh flag

APPROACH FOR CAPTURING FLAG #7

When we tried to enter wrong ID and password on the site with port number `3930`, we were able to see some SQL queries. So, we tried to perform SQL injection as taught in on of the hands-on session, we used the following paramters to perform it:

```
Username: admin' --
Password: #
```

And we were inside admin. The flag was visible as shown in Figure 7 above.

APPROACH FOR CAPTURING FLAG #8

When we accessed the site via port number `3940`, we encountered a message prompting us to click a button to retrieve the flag. However, clicking the button only displayed a cat image. Using Burp Suite Proxy Interceptor, we intercepted the request and noted that it was sending `cat+img` when the button was clicked. Suspecting manipulation, we modified the request to `cat+flag` and forwarded it. This alteration successfully revealed the flag, as depicted in Figure 8.

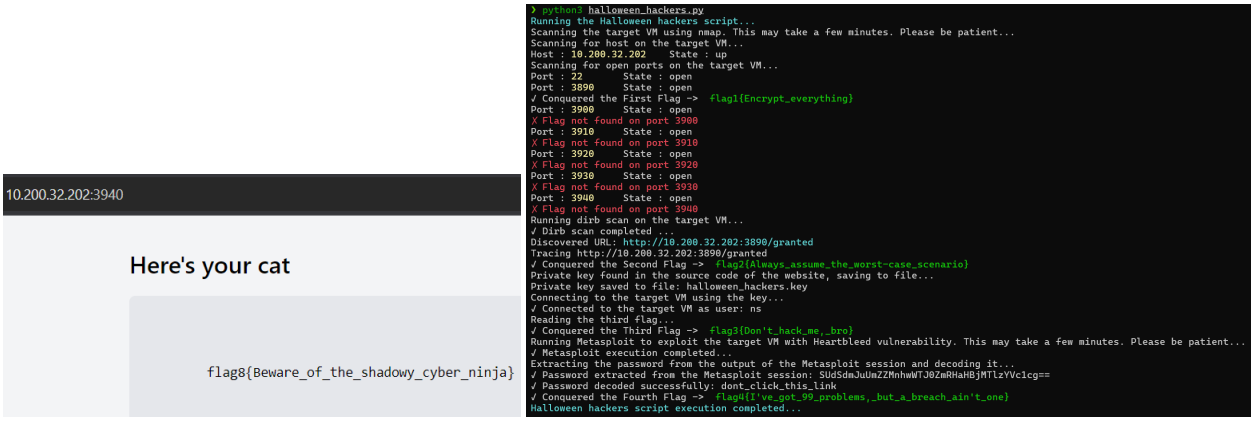

Figure 8. The eight flag and Automated Python Script Result

CREDIT STATEMENT

- **Kritik Agarwal**: `Captured the flags 3,4,6, Helped prepare the report and script.`
- **Raghavendra Kulkarni**: `Captured the flags 1,2,5, Helped prepare the script.`
- **Rohit Sutrave**: `Captured the flags 7,8 and prepared the report and helped with the script preparation.`

ANTI-PLAGIARISM STATEMENT

We certify that this assignment/report is our own work, based on our personal study and/or research and that we have acknowledged all material and sources used in its preparation, whether they be books, articles, ChatGPT tips, packages, datasets, reports, lecture notes, and any other kind of document, electronic or personal communication. We also certify that this assignment/report has not previously been submitted for assessment/project in any other course lab, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that we have not copied in part or whole or otherwise plagiarized the work of other students in this group. We pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, We understand my responsibility to report honor violations by other students if we become aware of it.

Names: `Kritik Agarwal, Raghavendra Kulkarni, Rohit Sutrave`
Date: `17th April, 2024`
Signature: `K.A., R.G.K., R.S.`

REFERENCES

1. Nmap
2. DIRB
3. Burp Suite
4. Base64 Decoder
5. Heartbleed Attack using Metasploit
6. Classroom materials shared by Dr. Bheemarjuna Reddy Tamma