Practical: 1

Date: 14/12/2022

Aim: Perform 5 different types of (port) scanning using nmap on a single port and capture the packets using wireshark and analyze the output.

Theory:

1. Nmap:

- Nmap is a free and open-source network scanner created by Gordon Lyon. Nmap is used to discover hosts and services on a computer network by sending packets and analyzing the responses.
- Nmap provides a number of features for probing computer networks, including host discovery and service and operating system detection.
- These features are extensible by scripts that provide more advanced service detection, vulnerability detection, and other features.
- Nmap can adapt to network conditions including latency and congestion during a scan.
- Nmap started as a Linux utility and was ported to other systems including Windows, macOS, and BSD. It is most popular on Linux, followed by Windows.

2. Wireshark:

- Wireshark is a free and open-source packet analyser.
- It is used for network troubleshooting, analysis, software and communications protocol
- development, and education.
- Originally named Ethereal, the project was renamed Wireshark in May 2006 due to
- trademark issues.
- Wireshark is cross-platform, using the Qt widget toolkit in current releases to implement
- its user interface, and using pcap to capture packets; it runs on Linux, macOS, BSD, Solaris, some other Unix-like operating systems, and Microsoft Windows.
- There is also a terminal-based (non-GUI) version called TShark. Wireshark, and the other programs distributed with it such as TShark, are free software, released under the terms of version 2 of the GNU General Public License.

3. open:

- An application is actively accepting TCP connections, UDP datagrams or SCTP associations on this port.
- Finding these is often the primary goal of port scanning. Security-minded people know that each open port is an avenue for attack.

- Attackers and pen-testers want to exploit the open ports, while administrators try to close or protect them with firewalls without thwarting legitimate users.
- Open ports are also interesting for non-security scans because they show services available for use on the network.

4. closed

- A closed port is accessible (it receives and responds to Nmap probe packets), but there is no application listening on it.
- They can be helpful in showing that a host is up on an IP address (host discovery, or ping scanning), and as part of OS detection. Because closed ports are reachable, it may be worth scanning later in case some open up.
- Administrators may want to consider blocking such ports with a firewall. Then they would appear in the filtered state, discussed next.

5. filtered

- Nmap cannot determine whether the port is open because packet filtering prevents its probes from reaching the port.
- The filtering could be from a dedicated firewall device, router rules, or host-based
- firewall software.
- These ports frustrate attackers because they provide so little information. Sometimes
- they respond with ICMP error messages such as type 3 code 13 (destination unreachable: communication administratively prohibited), but filters that simply drop probes without responding are far more common.
- This forces Nmap to retry several times just in case the probe was dropped due to network congestion rather than filtering. This slows down the scan dramatically.

6. Unfiltered

- The unfiltered state means that a port is accessible, but Nmap is unable to determine whether it is open or closed.
- Only the ACK scan, which is used to map firewall rulesets, classifies ports into this state.
- Scanning unfiltered ports with other scan types such as Window scan, SYN scan, or FIN scan, may help resolve whether the port is open.

7. open|filtered

- Nmap places ports in this state when it is unable to determine whether a port is open or filtered. This occurs for scan types in which open ports give no response.

- The lack of response could also mean that a packet filter dropped the probe or any response it elicited.
- So Nmap does not know for sure whether the port is open or being filtered. The UDP, IP protocol, FIN, NULL, and Xmas scans classify ports this way.

8. closed|filtered

- This state is used when Nmap is unable to determine whether a port is closed or filtered.
- It is only used for the IP ID idle scan.

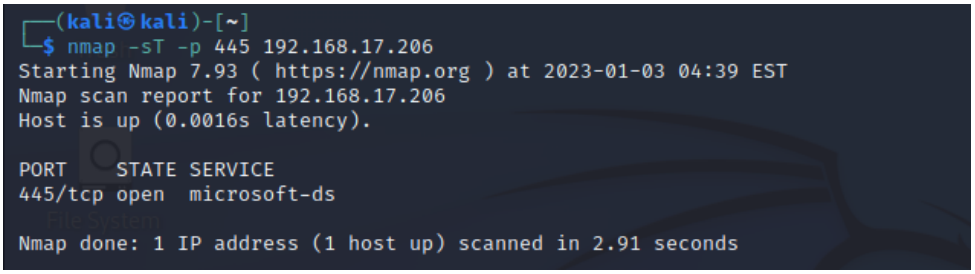
Implementation:

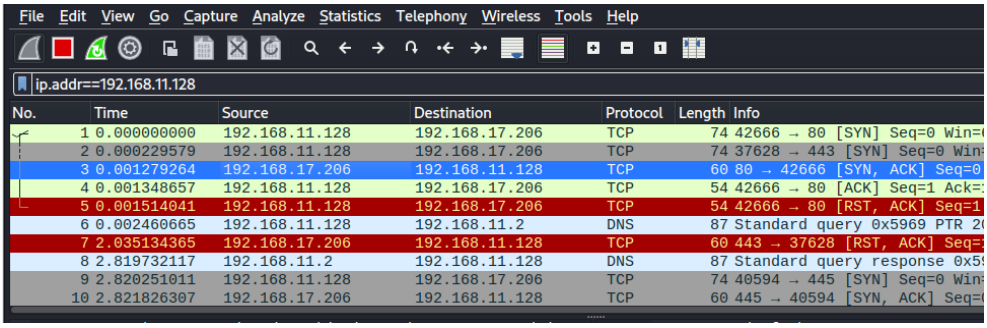
- There are two kinds of ports on each computer – TCP, and UDP – and 65,536 of each.
- The first 1024 TCP ports are the well-known ports like FTP(21), HTTP(80), or SSH(22).
- Anything above 1024 is available for use by services or applications.
- To scan Nmap ports on a remote system, enter the following in the terminal:

1. TCP Scan :

TCP scan will scan for TCP port like port 22, 21, 23, 445 etc and ensure for listening port through 3-way handshake connection between the source and destination port. If the port is open then source made request with SYN packet, a response destination sent SYN, ACK packet and then source sent ACK packets, at last source again sent RST, ACK packets.

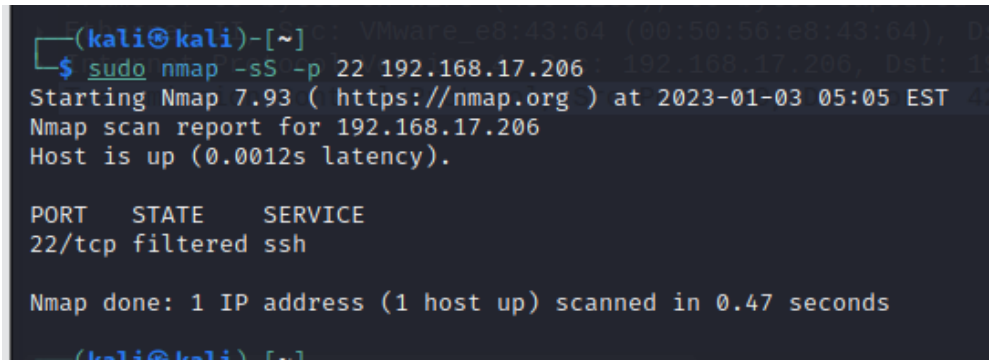
Type following NMAP command for TCP scan as well as start Wireshark on another hand to capture the sent Packet.

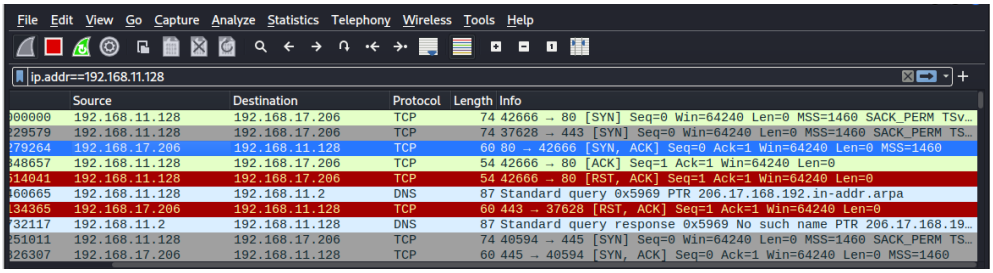




2. Stealth Scan:

SYN scan is the default and most popular scan option for good reasons. It can be performed quickly, scanning thousands of ports per second on a fast network not hampered by restrictive firewalls. It is also relatively typical and stealthy since it never completes TCP connections.



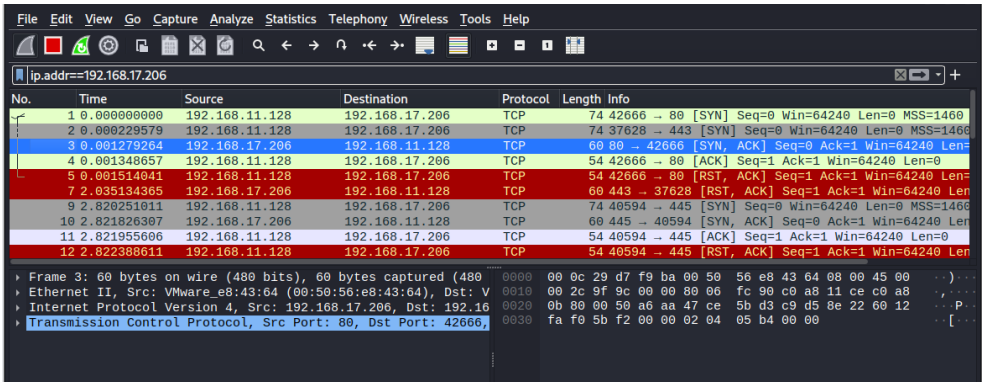


3. Fin Scan:

FIN packet is used to terminate the TCP connection between the source and destination port typically after the data transfer is complete. In the place of an SYN packet, Nmap starts a FIN scan by using a FIN packet. If the port is open then no response will come from destination port when FIN packet is sent through source port.

Type following NMAP command for TCP scan as well as start Wireshark on another hand to capture the sent Packet.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sF -p 22 192.168.17.206
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-03 05:09 EST
Nmap scan report for 192.168.17.206
Host is up (0.00094s latency).

PORT   STATE         SERVICE
22/tcp open|filtered ssh

Nmap done: 1 IP address (1 host up) scanned in 0.67 seconds
```

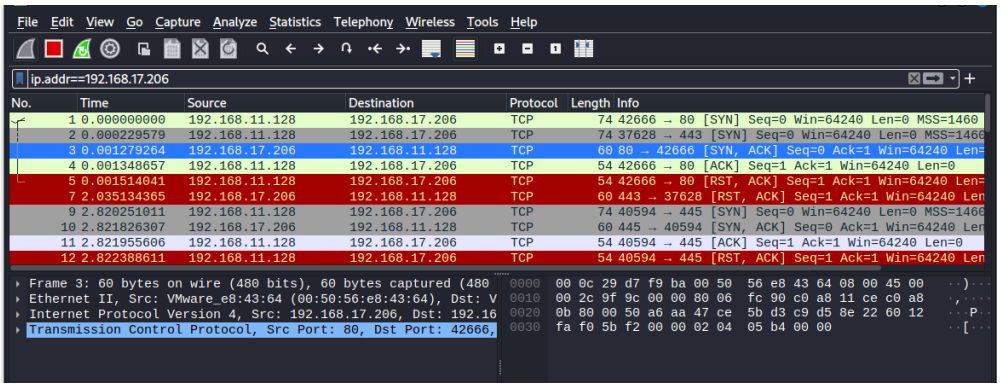


4. Null Scan:

Null Scan is a series of TCP packets which hold a sequence number of “zeros” (0000000) and since there are none flags set, the destination will not know how to reply the request. It will discard the packet and no reply will be sent, which indicate that the port is open.

Type following NMAP command for TCP scan as well as start Wireshark on another hand to capture the sent Packet.

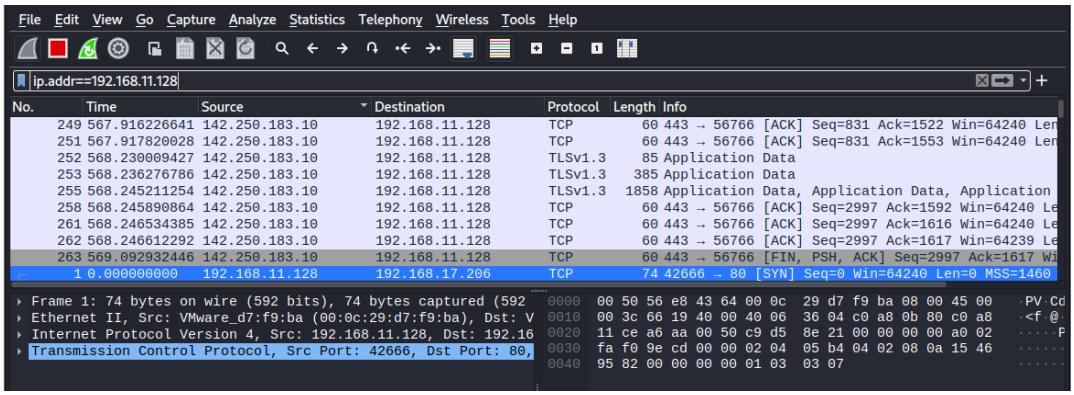
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sN -p 22 192.168.17.206
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-03 05:07 EST
Nmap scan report for 192.168.17.206
Host is up (0.0011s latency).

PORT   STATE         SERVICE
22/tcp open|filtered ssh

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```



5. UDP Scan :

UDP scan works by sending a UDP packet to every destination port; it is a connectionless protocol. For some common ports such as 53 and 161, a protocol-specific payload is sent to increase the response rate, a service will respond with a UDP packet, proving that it is open. If no response is received after retransmissions, the port is classified as open|filtered. This means that the port could be open, or perhaps packet filters are blocking the communication.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sU -p 161 192.168.11.128
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-03 05:13 EST
Nmap scan report for 192.168.11.128
Host is up (0.00022s latency).

PORT    STATE  SERVICE
161/udp closed snmp

Nmap done: 1 IP address (1 host up) scanned in 0.37 seconds
```



Conclusion/Summary:

Student Signature & Date	Marks	Evaluator Signature & Date

Practical 2

Date: 21/12/2022

Aim: Perform a Vulnerability Scan on a system within the Local Area Network and Submit the report.

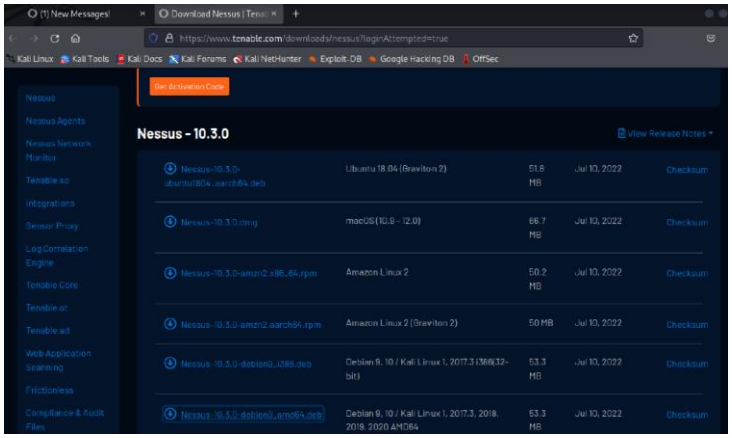
Theory:

Nessus Essentials:

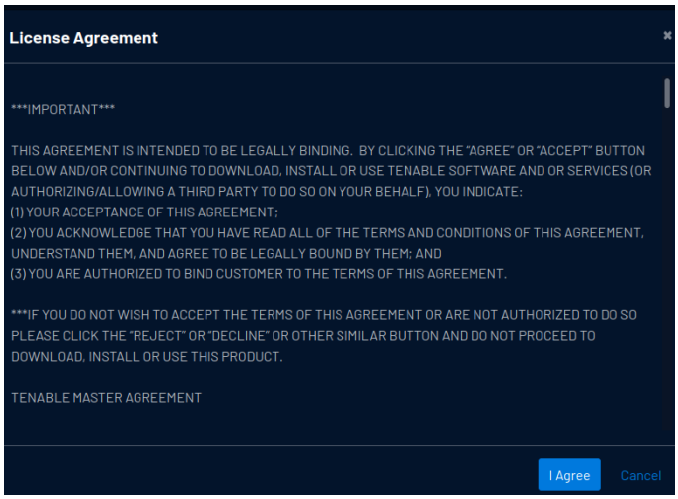
- Nessus Essentials is a free vulnerability assessment solution for up to 16 IPs that provides an entry point into the Tenable ecosystem.
- Backed by market leading functionality from Nessus Professional, Nessus Essentials gives you the accuracy and speed you need to discover, prioritize and remediate vulnerabilities.

Implementation:

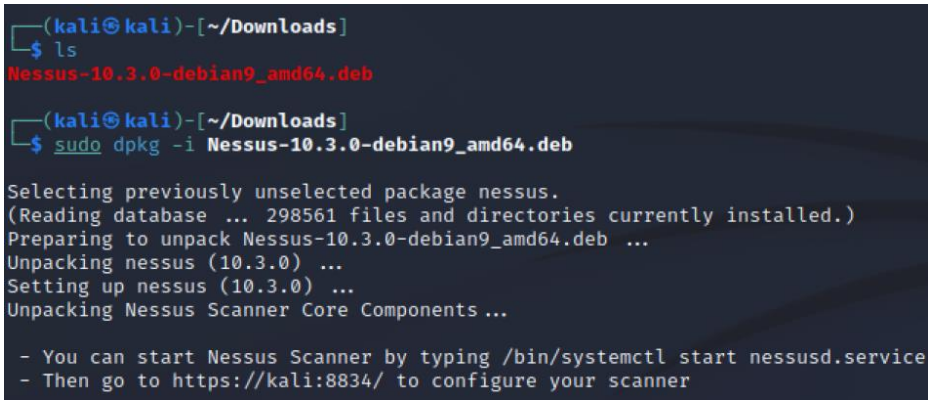
- Firstly, Nessus Essential is not pre-installed. Hence, we need to download it.



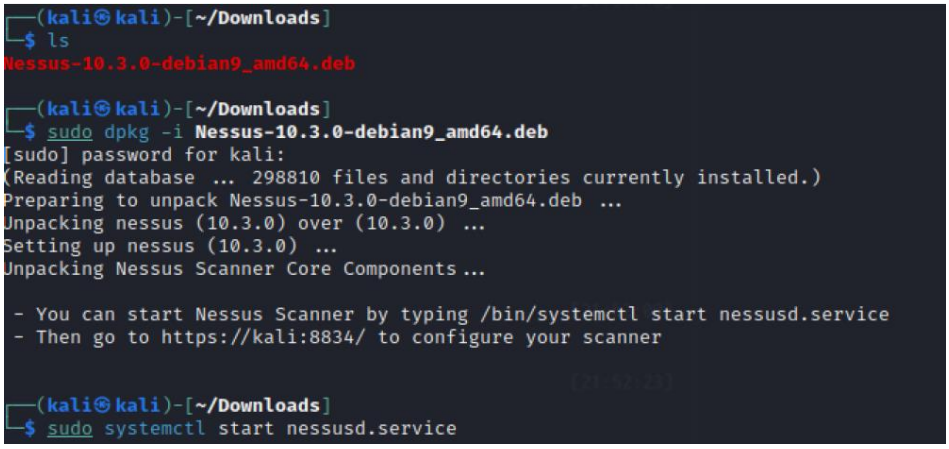
- Accept the Agreement



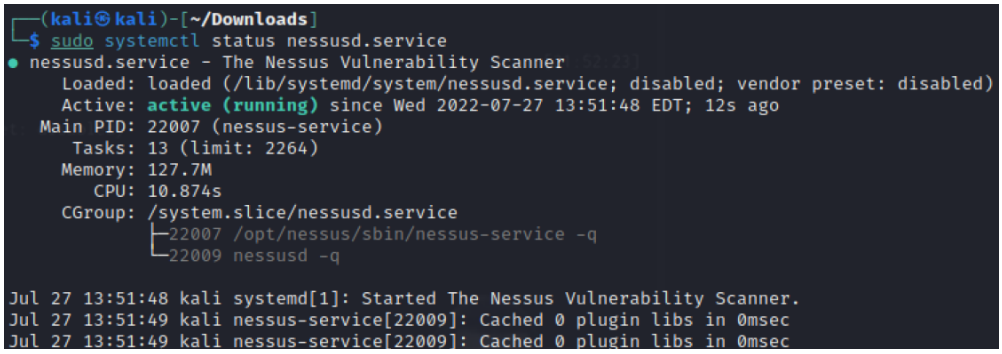
- Now, unpack the package



- Now, enter the following commands



- Check the status

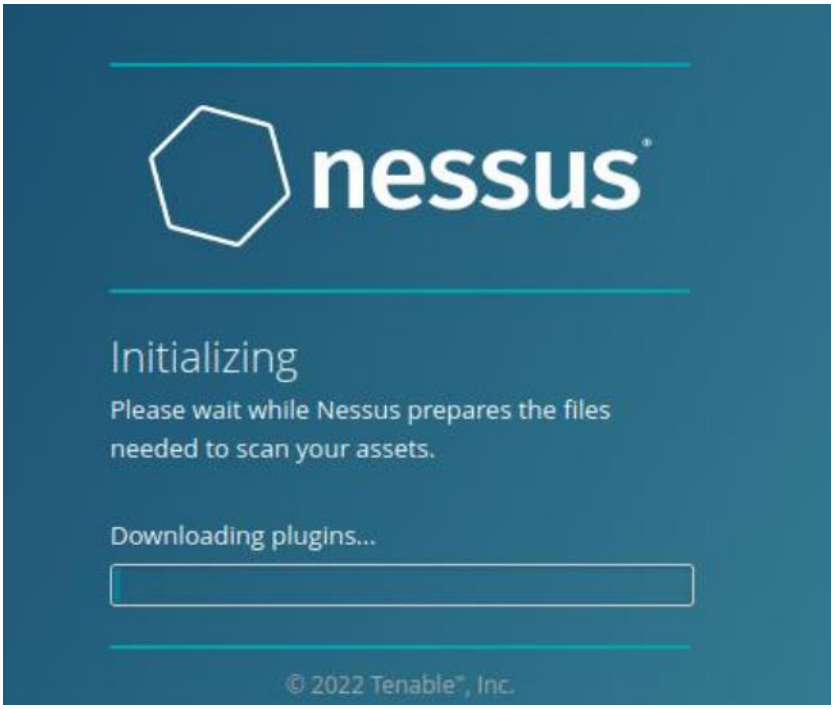


- Go to the link provided and proceed further. (https://kali:8834)

- The installation page will arrive

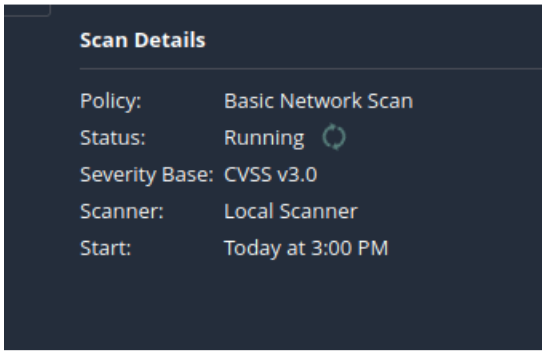
- Apply the activation code.

- After providing the user_name and password, download process will begin.

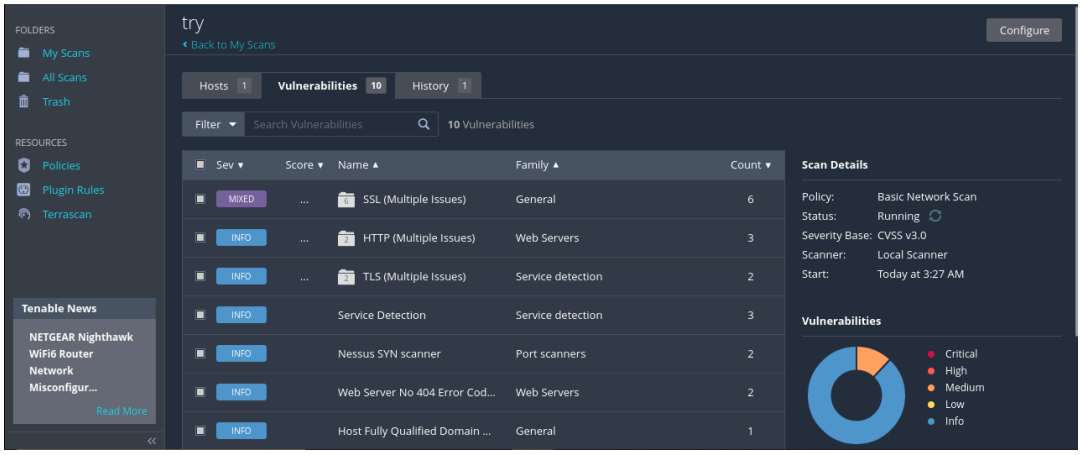


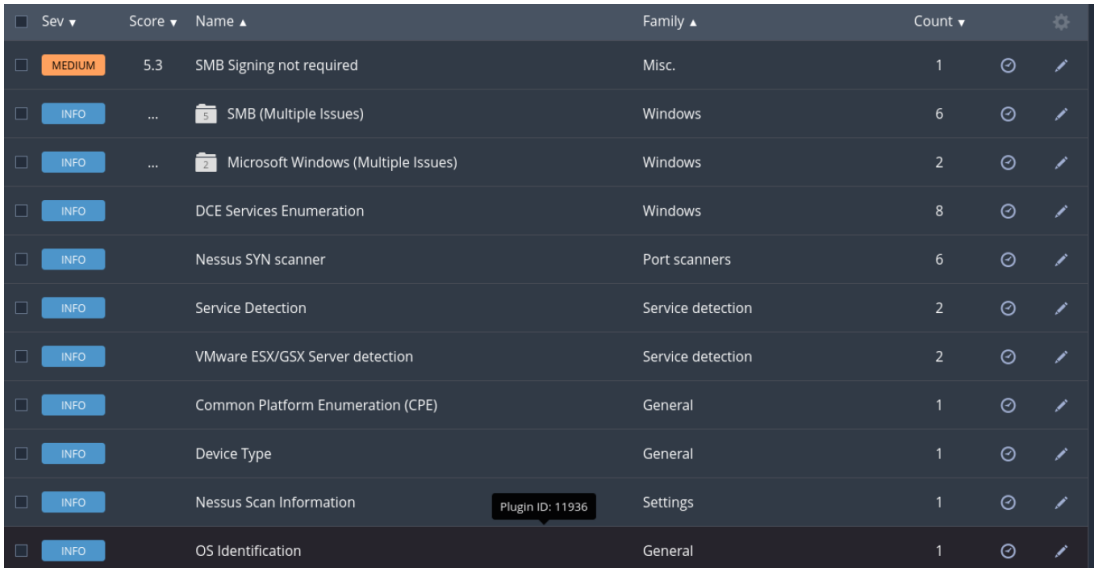
- Once, all plugins are installed, it will prompt you to enter the details of hosts that you want to check for.

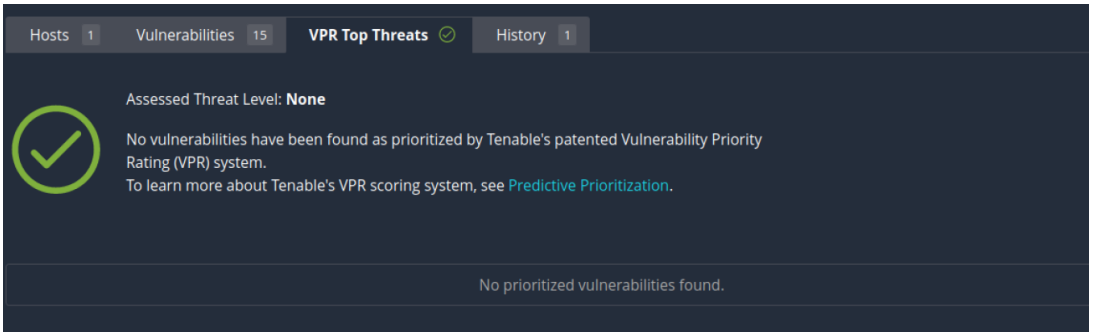
- After that, it will start the scanning.



- Once completed, following information will be shown.







Conclusion/Summary:		
Student Signature & Date	**Marks**	**Evaluator Signature & Date**

Practical: 3

Date: 28/12/2022

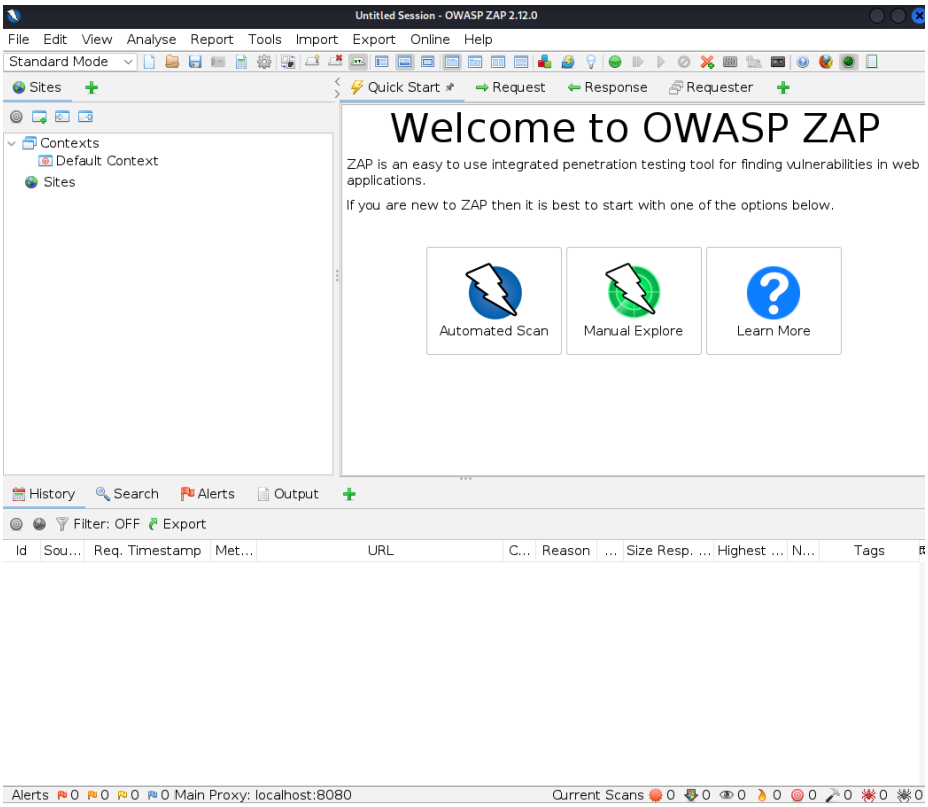
Aim: Implementation to identify web vulnerabilities, using OWASP project

Theory:

- OWASP stands for "Open Web Application Security Project".
- It is an open, online community that creates methodologies, tools, technologies and guidance on how to deliver secure web applications.
- OWASP ZAP (ZAP) is one of the world's most popular free security tools and is actively maintained by hundreds of international volunteers. It can help to find security vulnerabilities in web applications. It's also a great tool for experienced pen testers and beginners.
- ZAP is what is known as a "man-in-the-middle proxy." It stands between the browser and the web application. While you navigate through all the features of the website, it captures all actions. Then it attacks the website with known techniques to find security vulnerabilities.
- It is one of the most active Open Web Application Security Project (OWASP) projects and has been given Flagship status.
- When used as a proxy server it allows the user to manipulate all of the traffic that passes through it, including traffic using https.
- It can also run in a daemon mode which is then controlled via a REST API.
- ZAP was added to the ThoughtWorks Technology Radar in May 2015 in the Trial ring.
- ZAP was originally forked from Paros, another pentesting proxy. Simon Bennetts, the project lead, stated in 2014 that only 20% of ZAP's source code was still from Paros.

Implementation:

- Starting ZAP.
- Once setup you can start ZAP by clicking the ZAP icon on your Windows desktop or from the start menu.

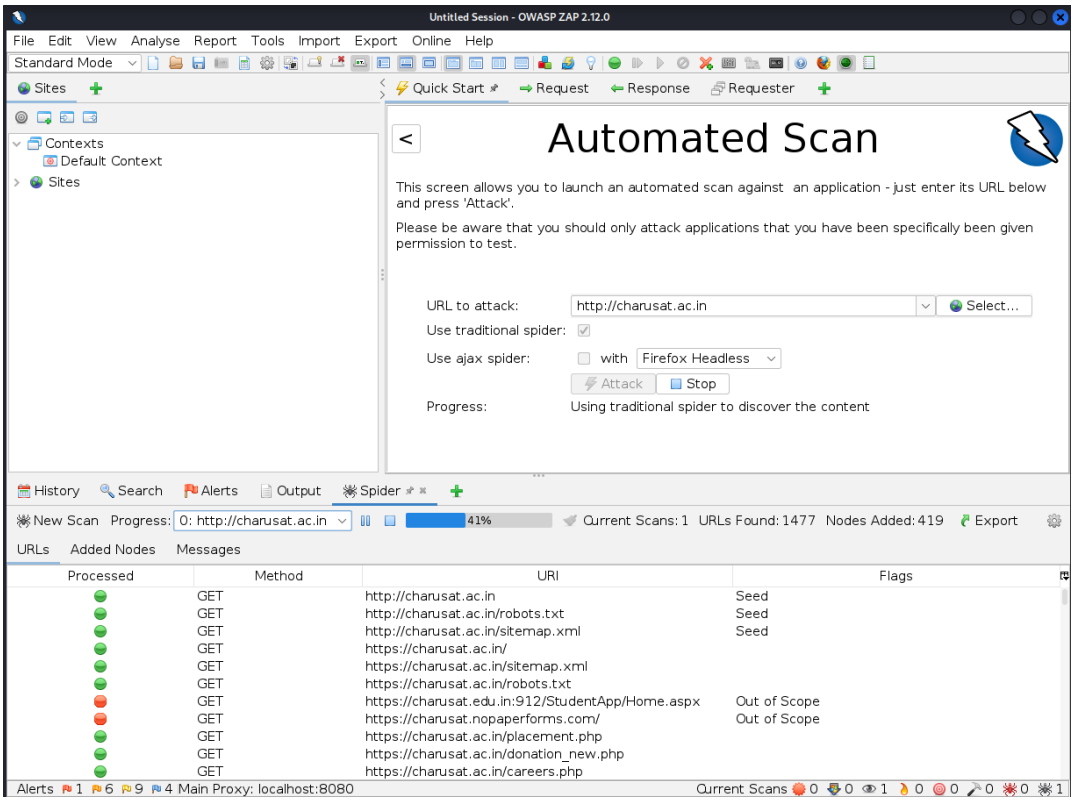


Glimpses Of ZAP Console

- Spidering the web application
- Spidering a web application means crawling all the links and getting the structure of the application. ZAP provides two spiders for crawling web applications;
- The traditional ZAP spider discovers links by examining the HTML in responses from the web application. This spider is fast, but it is not always effective when exploring an AJAX web application.
- This is more likely to be effective for AJAX applications. This spider explores the web application by invoking browsers which then follow the links that have been generated. The AJAX spider is slower than the traditional spider.

Automated Scan:

- This option allows you to launch an automated scan against an application just by entering the URL. If you are new to ZAP, it is best to start with Automated Scan mode.
- To run a Quick Start Automated Scan:
  1. Start Zap and click the large ‘Automated Scan’ button in the ‘Quick Start’ tab.
  2. Enter the full URL of the web application you want to attack in the ‘URL to attack’ text box.
  3. Click the ‘Attack’ button.

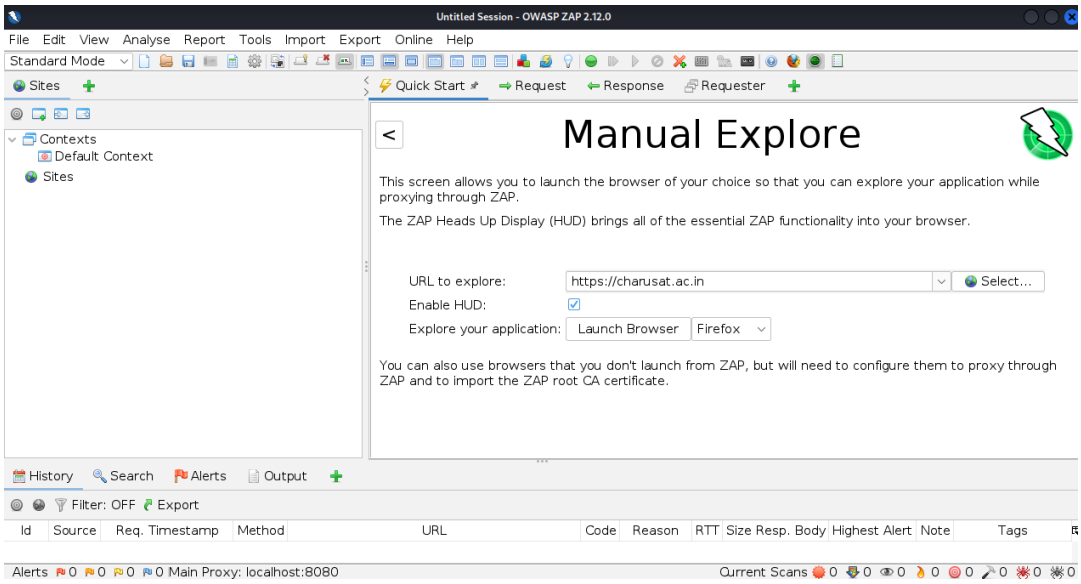


Crawling started

- Once you click the ‘Attack’ button, ZAP will start crawling the web application with its spider and passively scan each page it finds. Then ZAP will use the active scanner to attack all of the discovered pages, functionality and parameters.
- Exploring the web application manually
- Spiders are a great way to explore the basic site, but they should be combined with manual exploration to be more effective. This functionality is very useful when your web application needs a login or contains things like registration forms, etc.

- You can launch browsers that are pre-configured to proxy through ZAP via the Quick Start tab. Browsers launched in this way will also ignore any certificate validation warnings that would otherwise be reported.

Manual Explore:



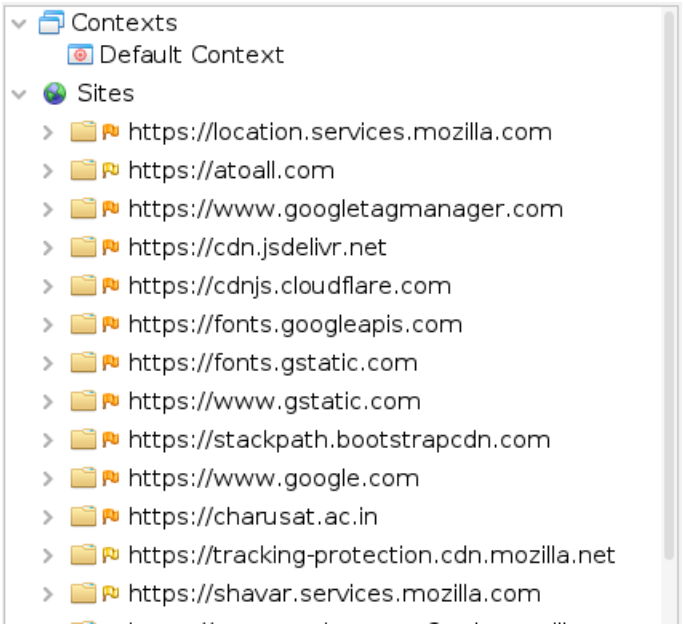
- To Manually Explore the web application.
- Start ZAP and click on the large “Manual Explore” button in the Quick Start tab.
- Enter the full URL of the web application to be explored in the ‘URL to explore’ text box.
- Select the browser you would like to use and click the ‘Launch Browser’ button.
- This will launch the selected browser with a new profile. Now explore all of the targeted web applications through this browser.
- ZAP passively scans all the requests and responses made during your exploration for vulnerabilities, continues to build the site tree, and records alerts for potential vulnerabilities found during the exploration.

What is passive scanning?

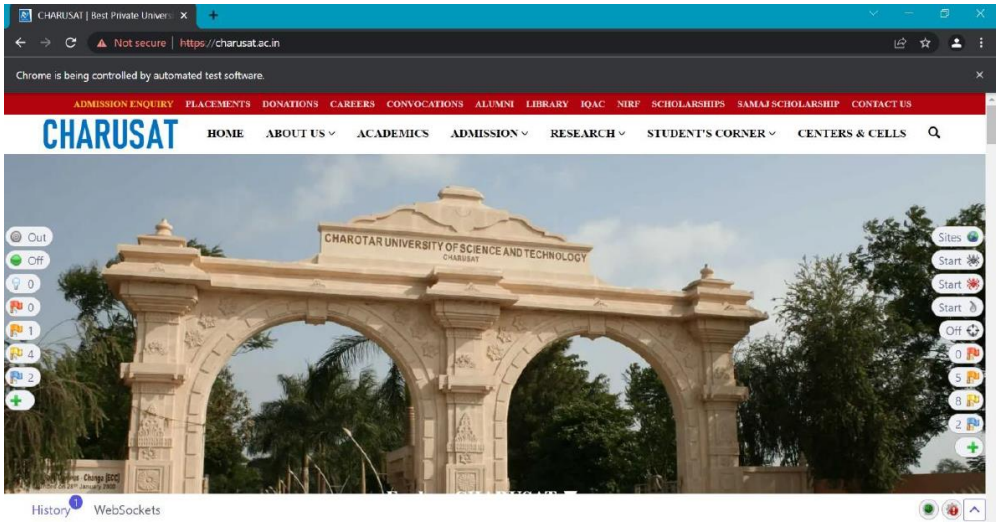
- Passive Scans only scan the web application responses without altering them.
- It does not attack or insert malicious scripts to the web application, so this is a safe scan; you can use it if you are new to security testing.
- Passive scanning is good at finding some vulnerabilities and as a way to get a feel for the basic security of a web application.

What is active scanning?

- Active scan attacks the web application using known techniques to find vulnerabilities. This is a real attack that attempts to modify data and insert malicious scripts in the web application.



Sites Updates



Manual scanning initiated

- We will not proceed as we are not allowed to scan the website.

Conclusion/Summary:		
Student Signature & Date	**Marks**	**Evaluator Signature & Date**

Practical 4

Date: / /2023

Aim: Perform log analysis of machine data using Splunk software in windows/Linux. This machine data can come from web applications, sensors, devices, or any data created by the user.

Theory:

Splunk

Splunk (the product) captures, indexes, and correlates real-time data in a searchable repository from which it can generate graphs, reports, alerts, dashboards, and visualizations.

Features offered by Splunk Enterprise: -

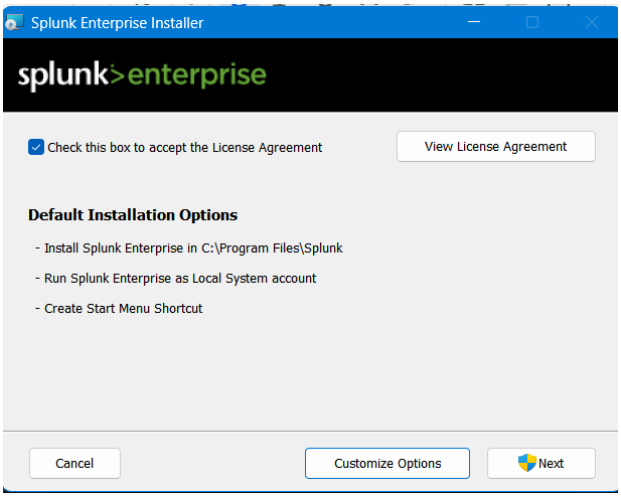
1. Data searching**: –** Searching in Splunk involves the pattern of creating metrics or indexes on Dashboards.
2. Data ingestion**: –** Splunk ingest data in various formats like XML, JSON, and unstructured machine data such as logs of CPU running on web servers.
3. Data Indexing: **–** Splunk auto index the ingested data of various machines for the faster searching on various conditions
4. Alerts: – Splunk alert used for triggering emails or other feeds when some unusual suspicious activity found in data is being analyzed.
5. Dashboards: **–** It shows the search results in the form of pivots, area mapping, pie charts, reports, etc.

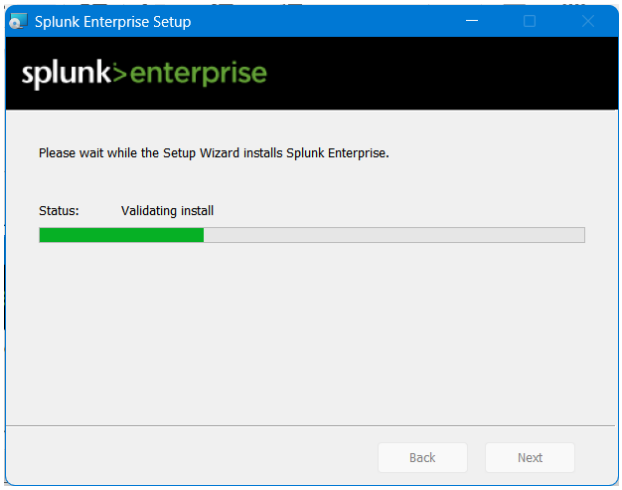
Uses of Splunk: -

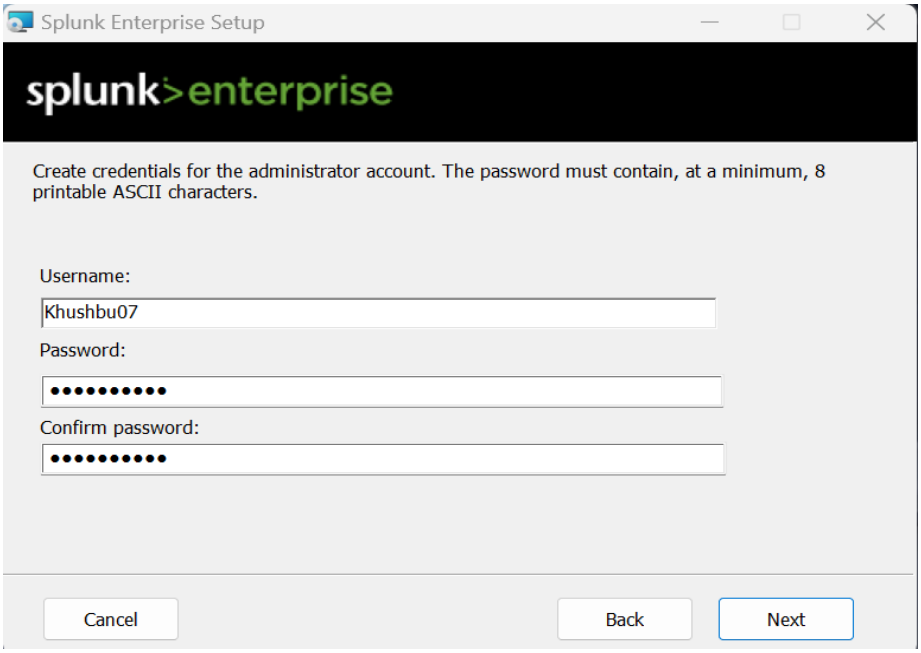
- Splunk is used for monitoring and searching through big data.
- It indexes and correlates information in a container that makes it searchable, and makes it possible to generate alerts, reports and visualizations.
- In the context of security, Splunk is essentially used as a log analysis engine. “It is used to correlate security events, which allows you to identify where your breaches are coming from,”

Implementation:

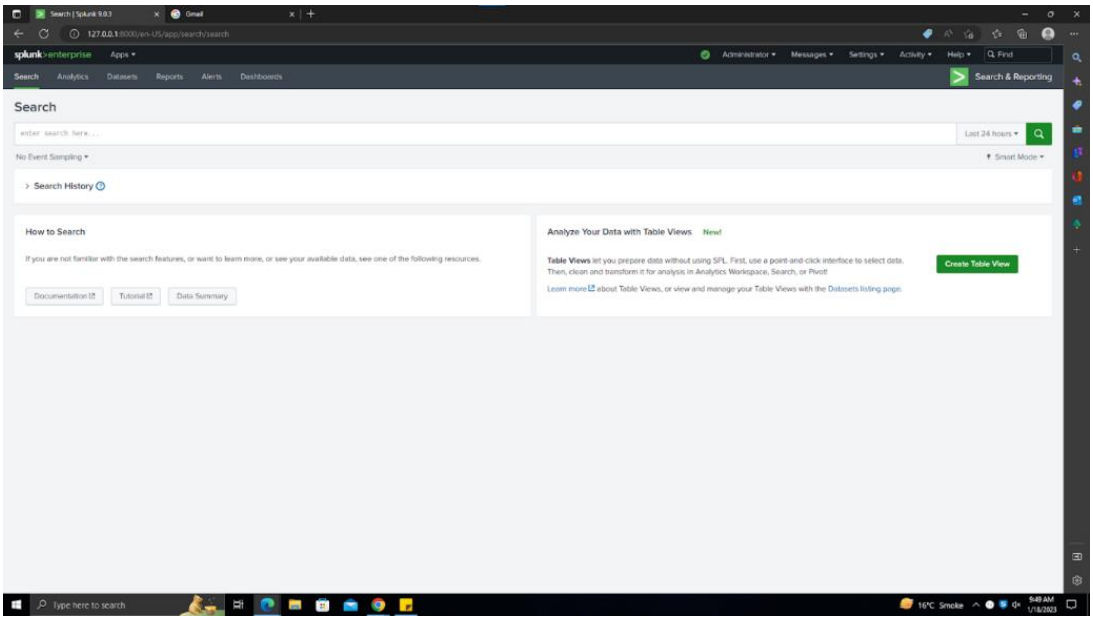
Installation: -







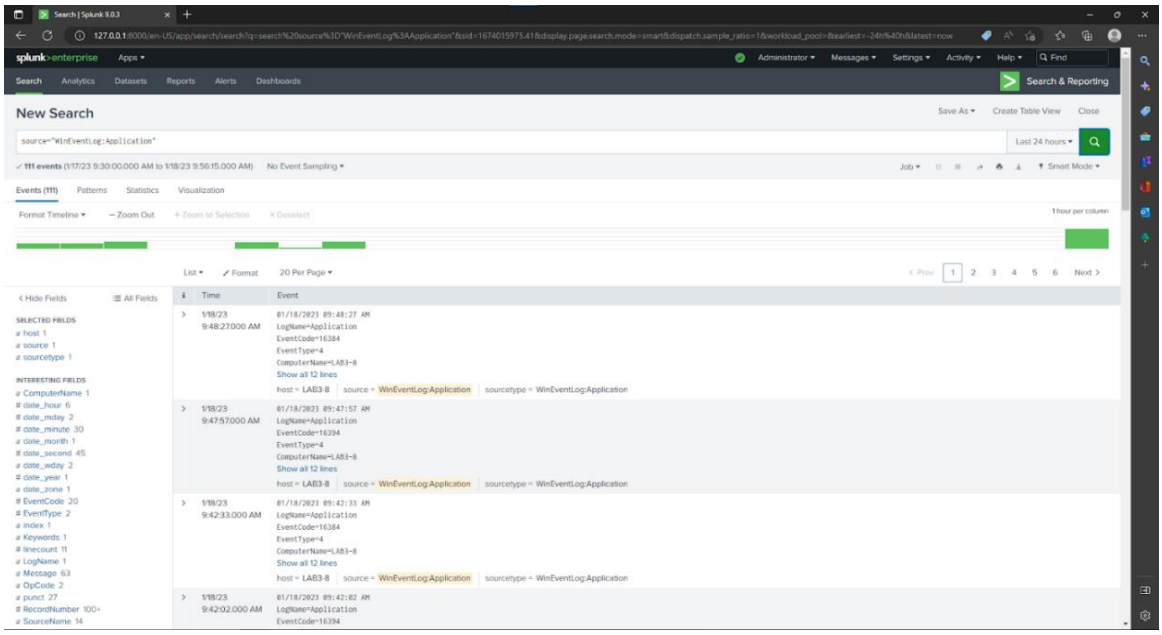
Step 1: Launch the Splunk Enterprise Application



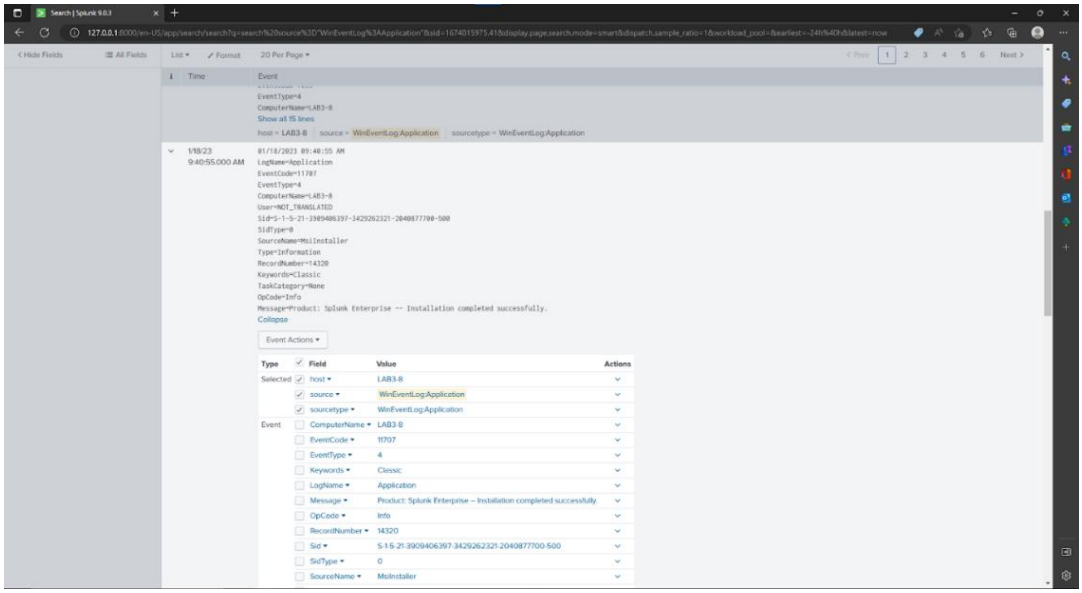
Step 2: Searching a Windows event

Type the following command in the filter or search section

source='WinEventLog:Application'



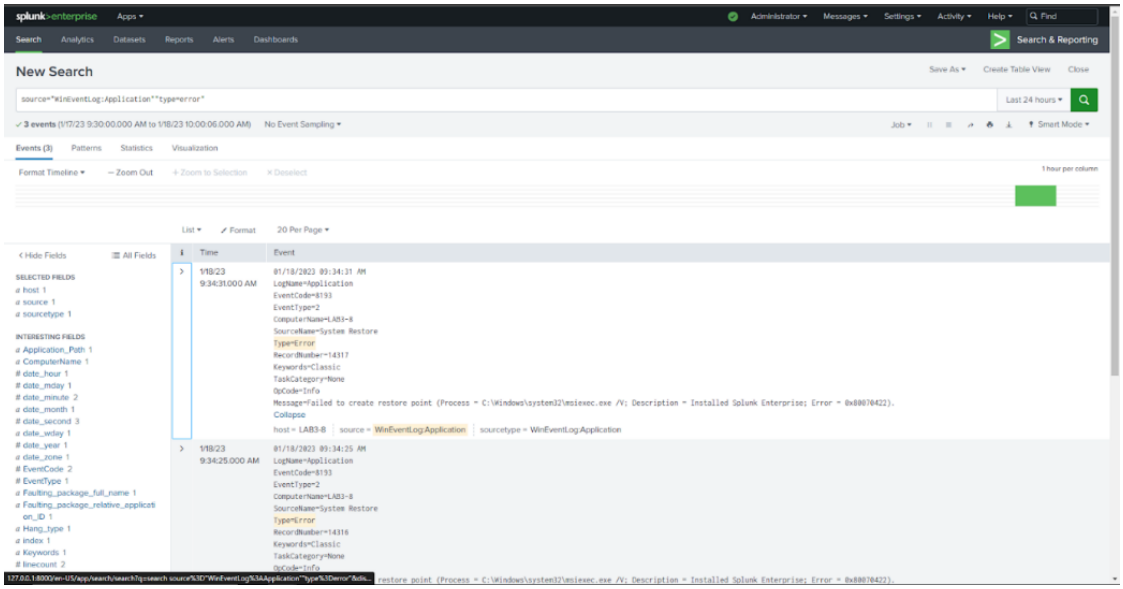
Step 3: Now expand any one Windows Event and observe the details



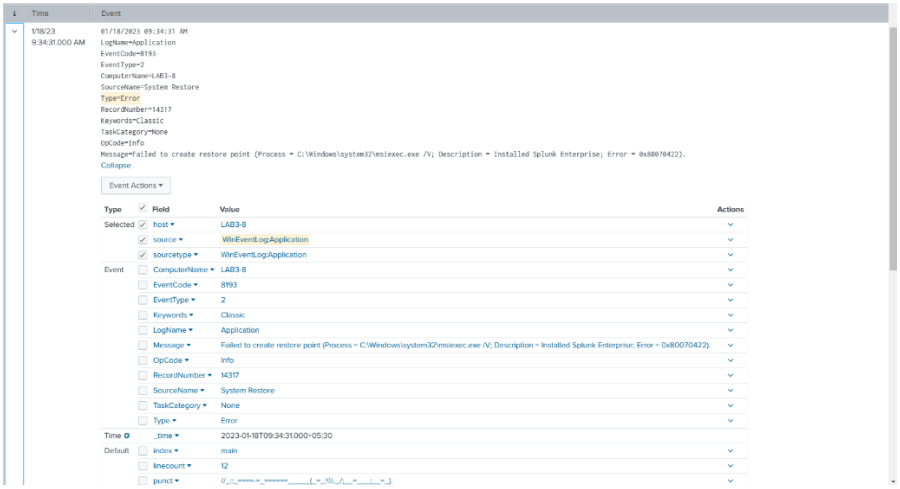
Step 4: Searching for any kind of Failure/error in any Windows Event

Type the following command in the filter or search section

source=’WinEventLog:Application’’type=error’

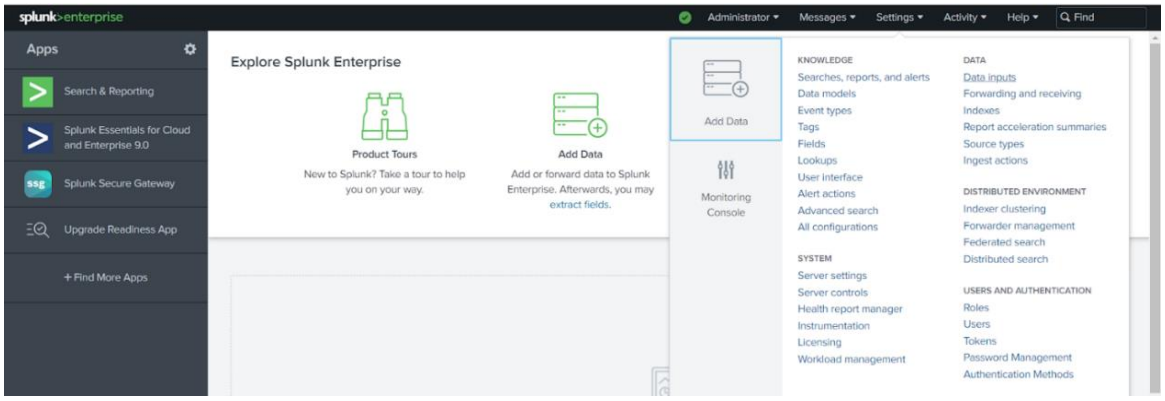


Step 5: Expand any one event of your choice and observe the details

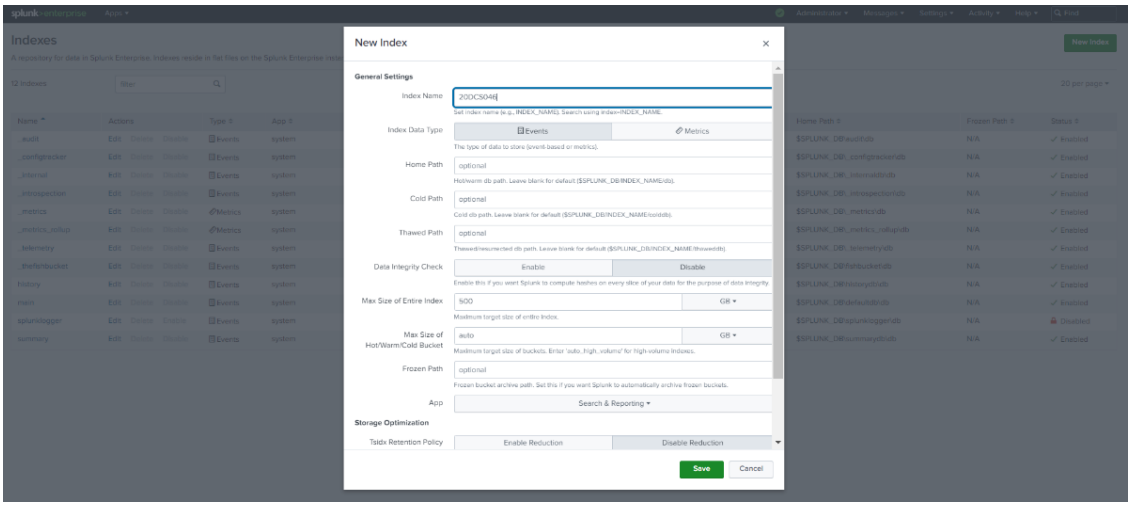


Step 6: Creating an Index

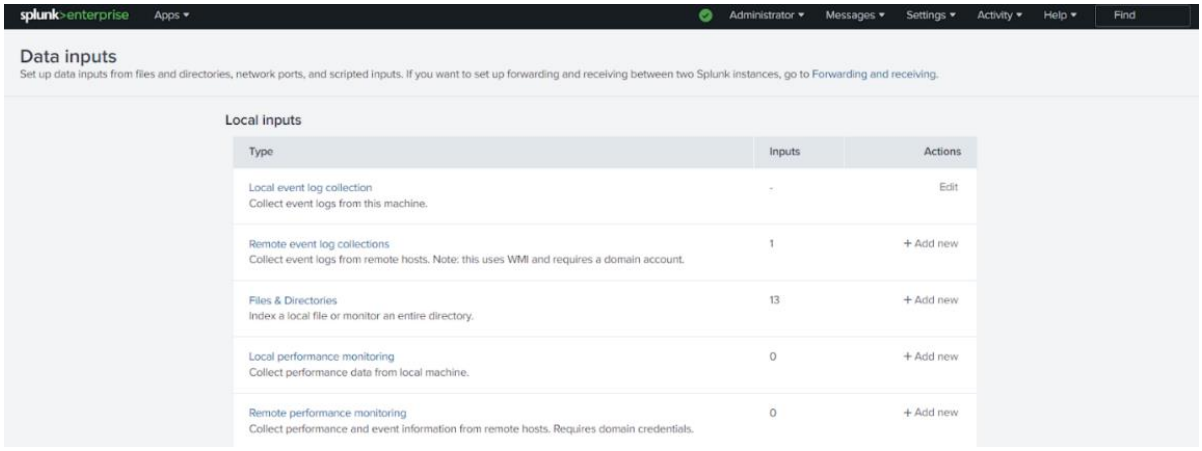
Hover on setting then under Data column click on Indexes

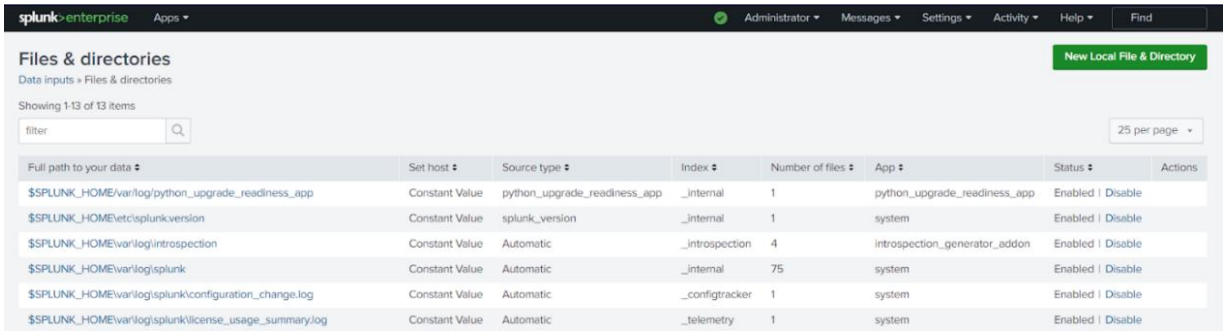


Now name the index as per your wish

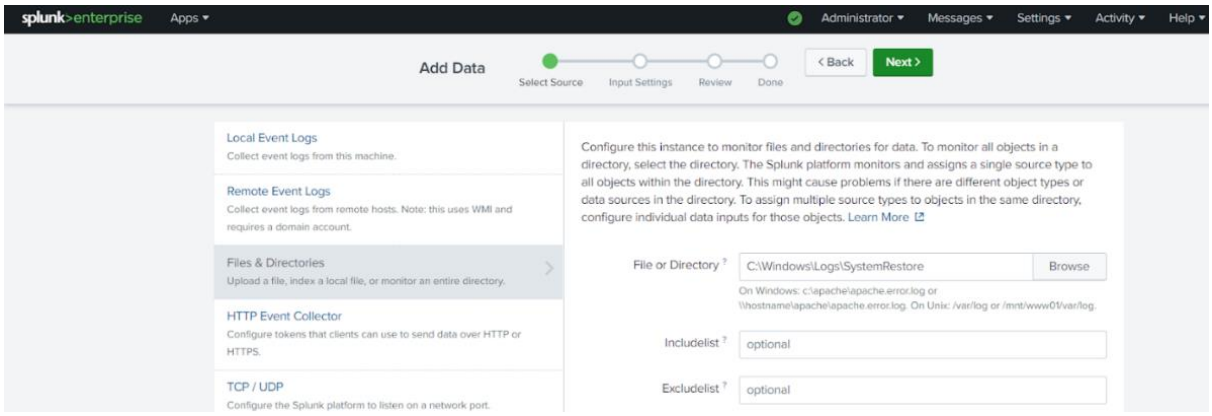


Step 2: In Data Inputs select Files & Directories option.

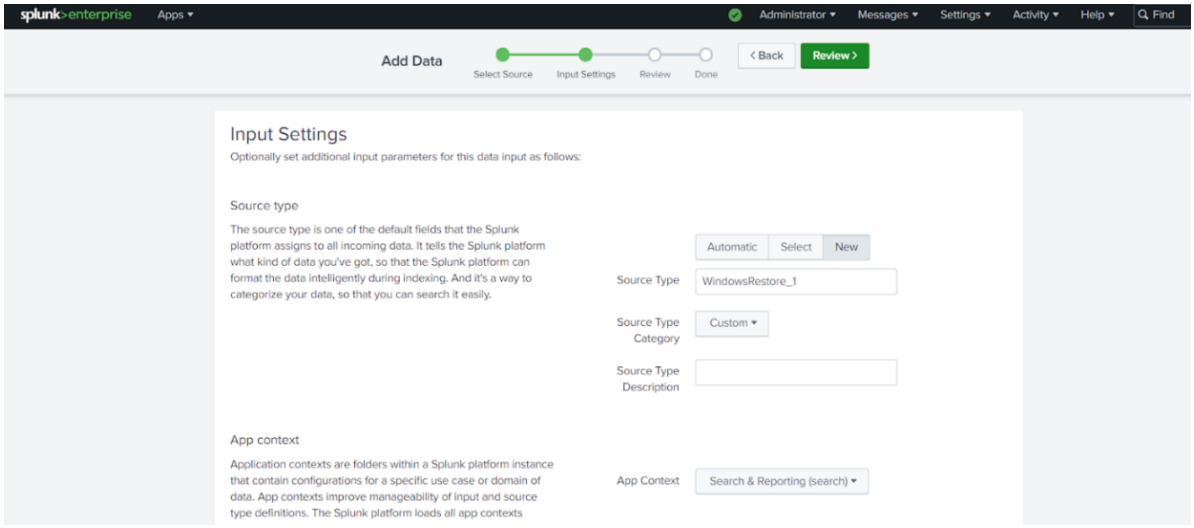




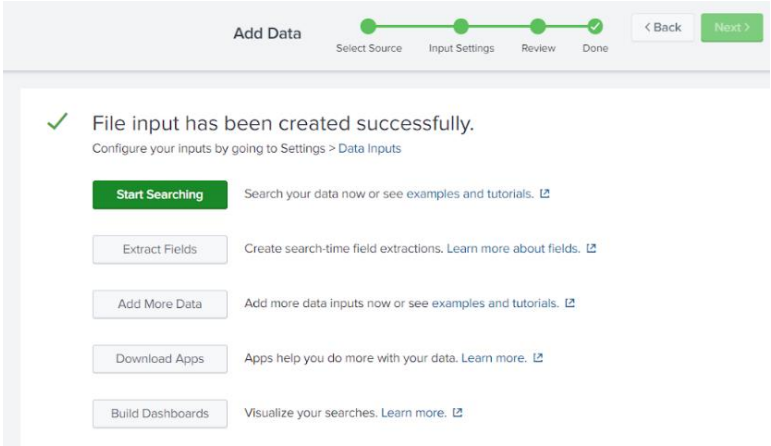
Step 3: Select the Source as shown below.



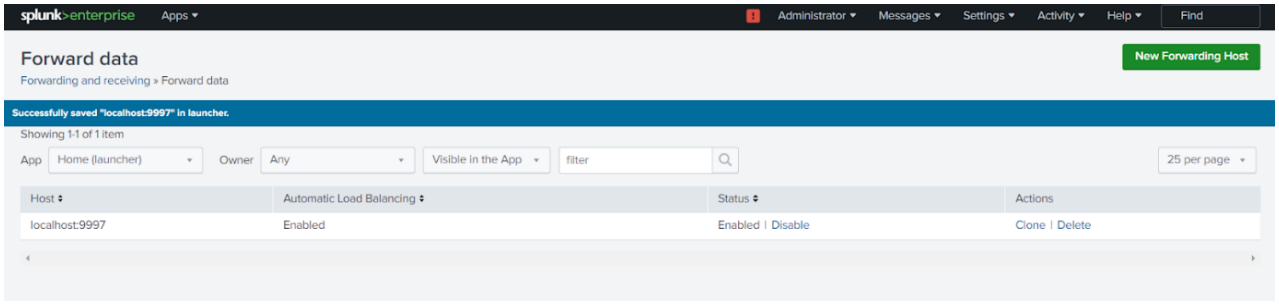
Step 4: Select the Input Settings as shown below.



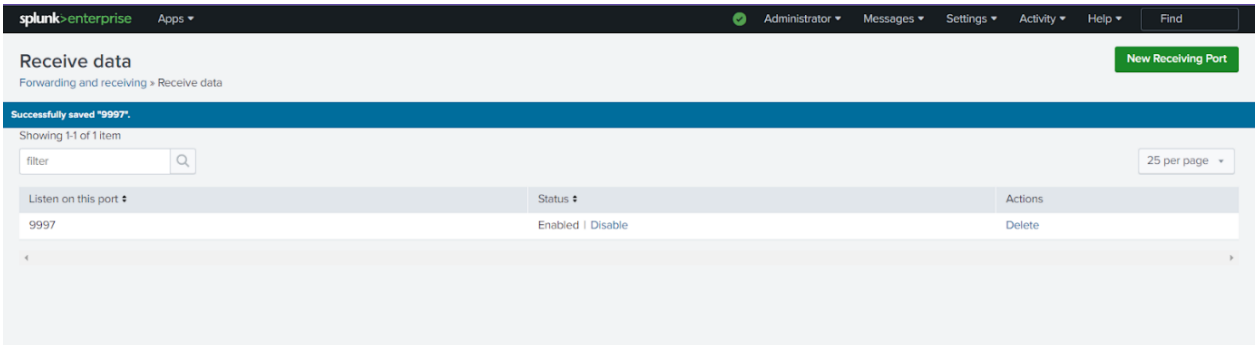
Step 5: Click on Submit Button.



Forward Data

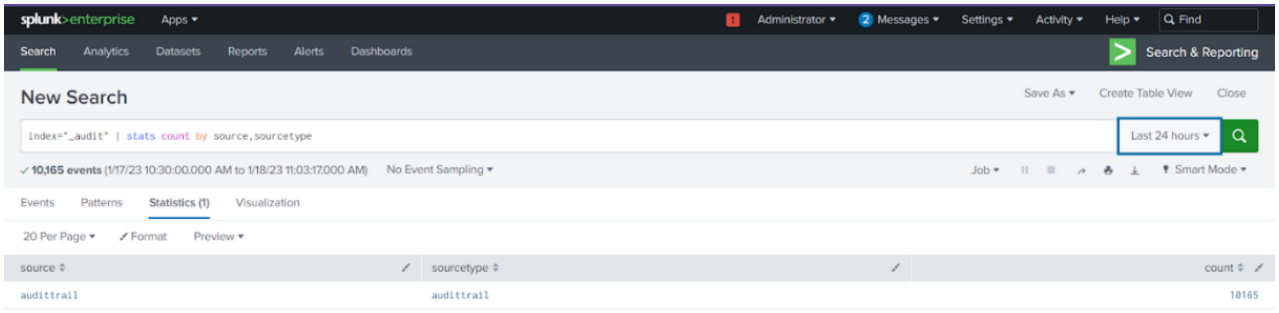


Receive Data

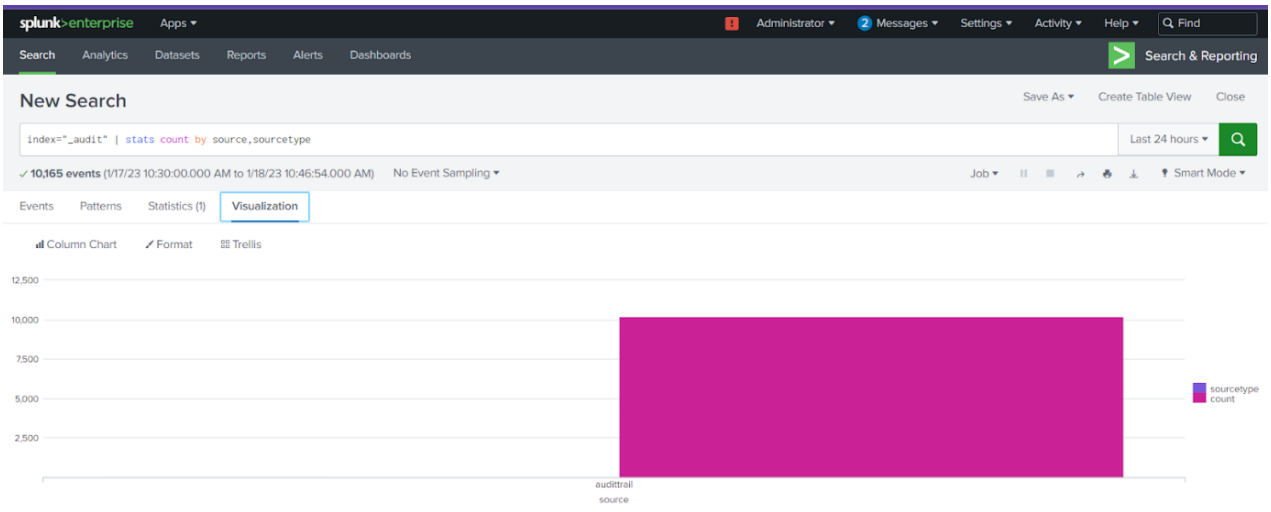


Step 7: Applying Search on the data by writing:

index=”_audit” | stats count by source, sourcetype.



Step 8: Visualization of logs of the selected file.



Conclusion/Summary:		
Student Signature & Date	**Marks:**	**Evaluator Signature & Date**

Practical 5

Date: / /2023

Aim: Monitor the traffic in real time and issue alerts to users when it discovers potentially malicious packets or threats on Internet Protocol network using SNORT.

Theory:

SNORT is an open-source, rule-based Network Intrusion Detection and Prevention System. It was developed and still maintained by Martin Rocher, open-source contributors, and the CISCO talos team.

Snort is the foremost open-source Intrusion Prevention System in the world. Snort IPS uses a series of rules that helps define malicious network activity and uses those rules to find packets that match against them and generate alerts for users.

Intrusion Detection System (IDS)

IDS is a passive monitoring solution for detecting possible malicious activity/patterns, abnormal incidents and policy violations. It is responsible for generating alerts for each suspicious event.

There are two main types of IDS systems:

- **Network-based IDS (NIDS) –** NIDS monitors the traffic flow from various areas of the network. The main aim is to investigate the traffic on the entire subnet. If a signature is identified, an alert is created.
- **Host-based IDS (HIDS) –** HIDS monitors the traffic flow form a single endpoint device. The aim is to investigate the traffic on a particular device. If a signature is identified, an alert is created.

Intrusion Prevention System (IPS)

IPS is an active protecting solution for preventing possible malicious activity/patterns, abnormal incidents and policy violations. It is responsible for stopping/preventing/terminating the suspicious event as soon as the detection is performed.

There are four main types of IPS systems:

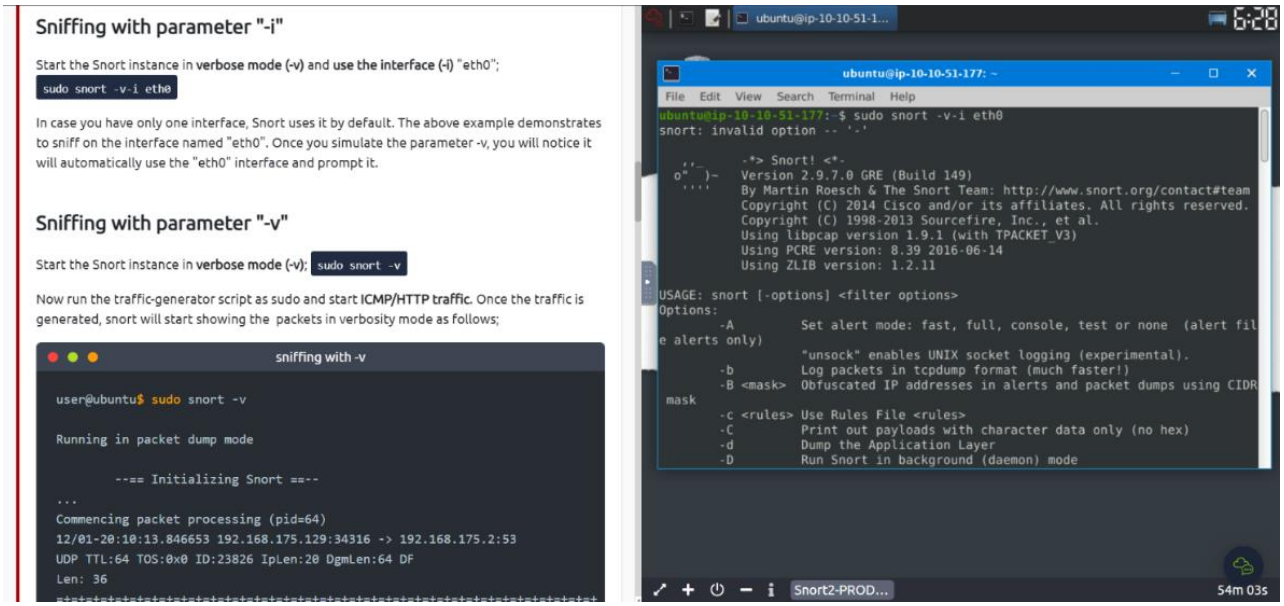
- **Network IPS (NIPS) –** NIPS monitors the traffic flow from various areas of the network. The aim is to protect the traffic on the entire subnet. If a signature is identified, the connection is terminated.
- **Network Behaviour-based IPS (NBA) –** It monitors the traffic flow from various area of the network. The aim is to protect the traffic on the entire subnet. If a signature is identified, the connection is terminated.

- **Wireless IPS (WIPS) –** It monitors the traffic flow from wireless network. The aim is to protect the wireless traffic and stop possible attacks launched from there. If a signature is identified, the connection is terminated.
- **Host-based IPS (HIPS) –** It actively protects the traffic flow from a single endpoint device. The aim is to investigate the traffic on a particular device. If a signature is identified, the connection is terminated.

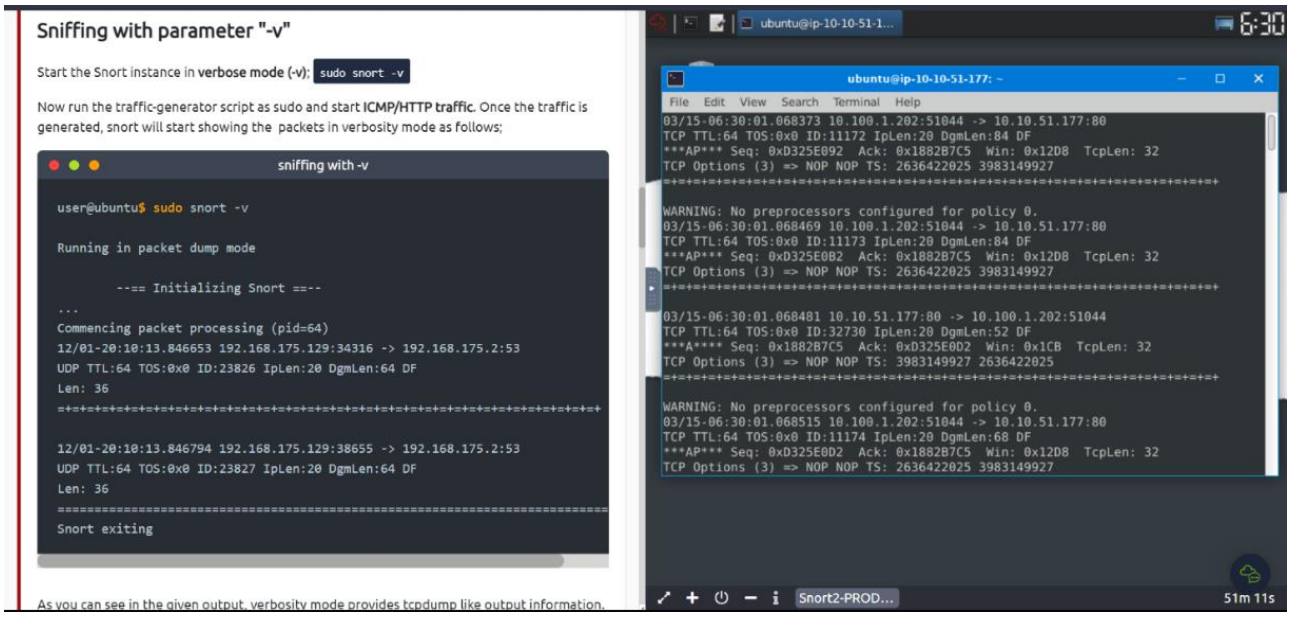
Implementation:

Step-1: Running Snort in sniffer mode

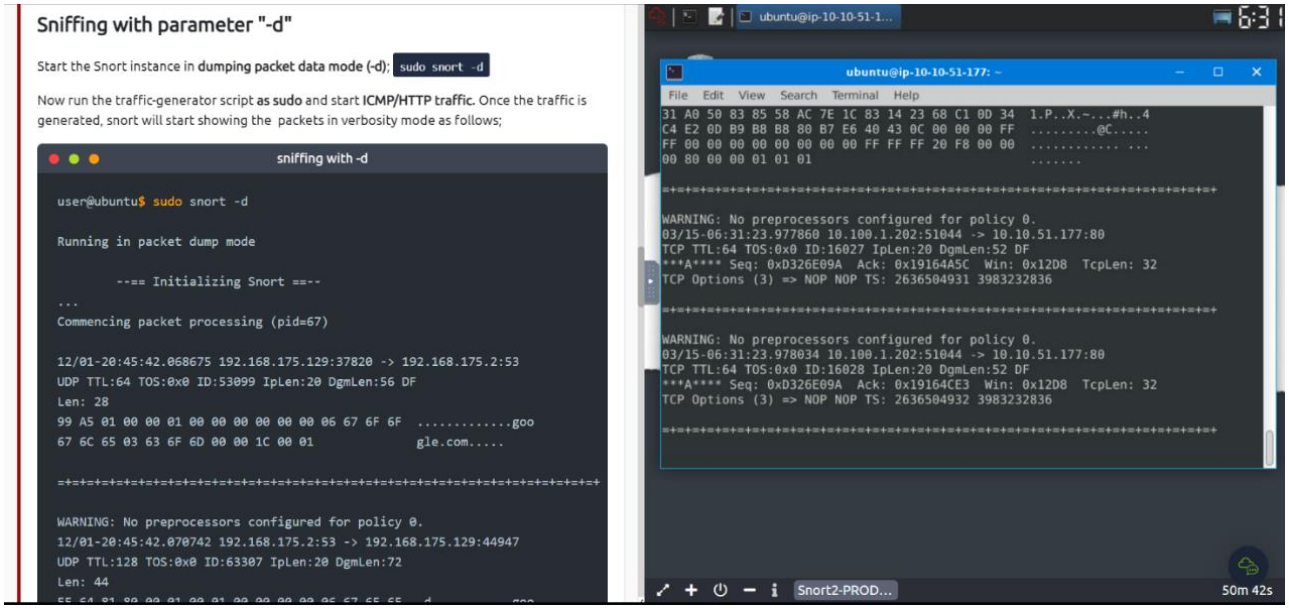
Sniffing with parameters "-i" verbose mode (-v) and the interface (-i).



Step 2: Sniffing with parameter "-v" verbose mode (-v)

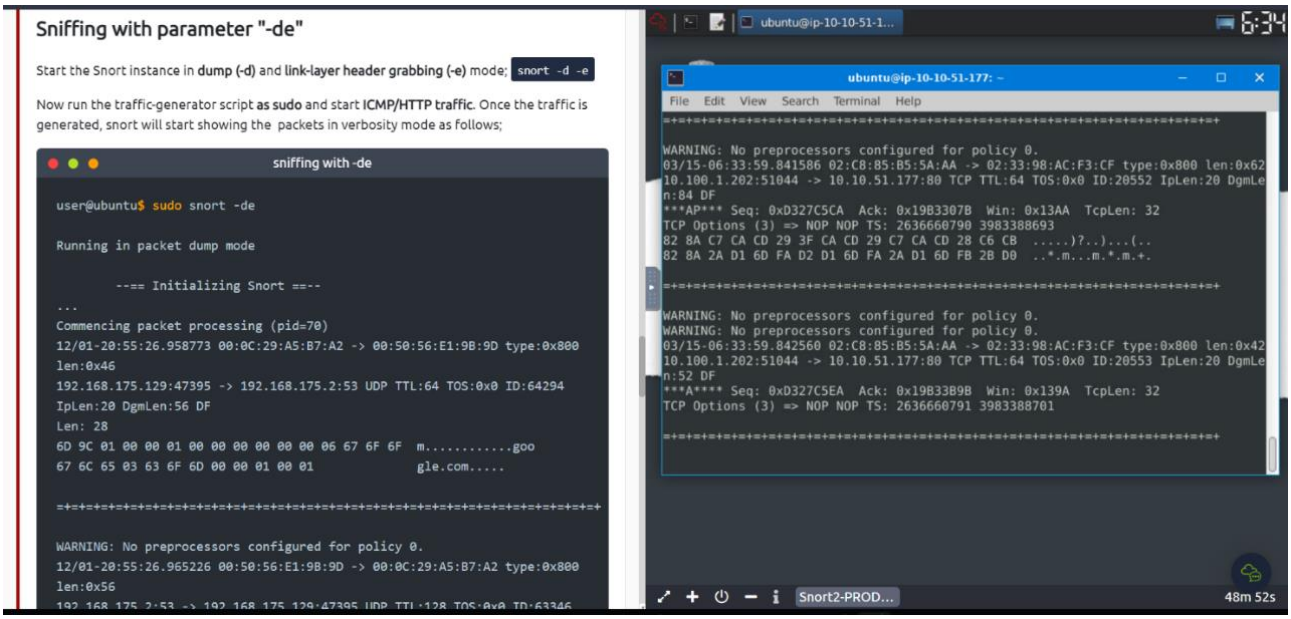


Step 3: Sniffing with parameter "-d" dumping packet data mode (-d)

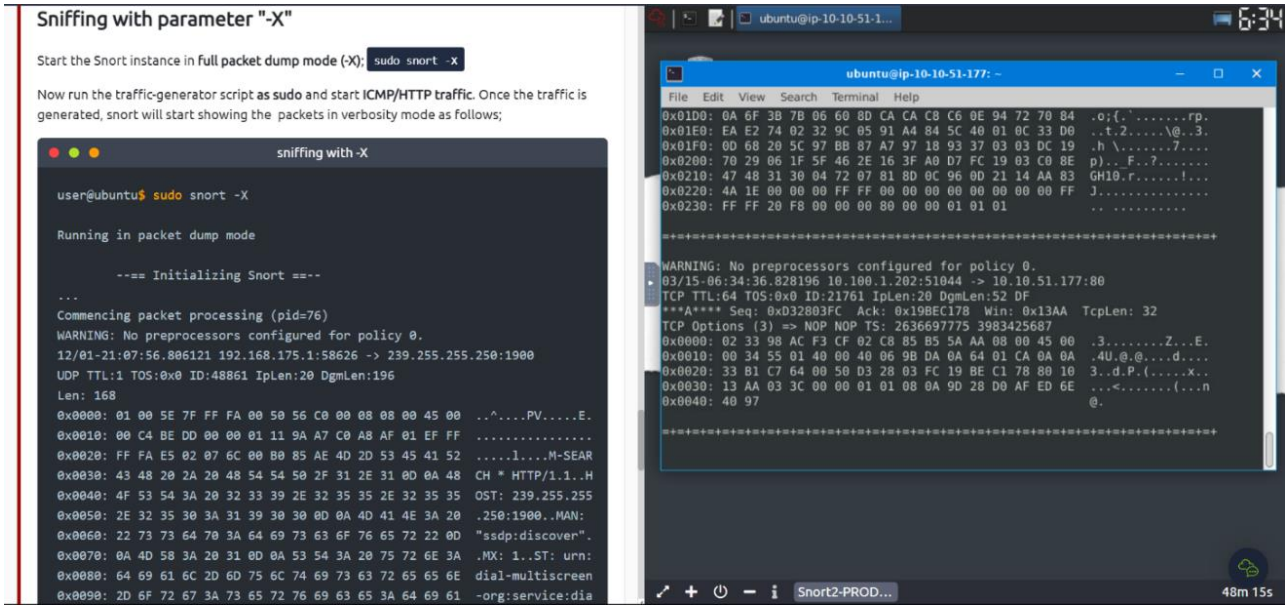


Step 4: Sniffing with parameter "-de"

dump (-d) and link-layer header grabbing (-e)

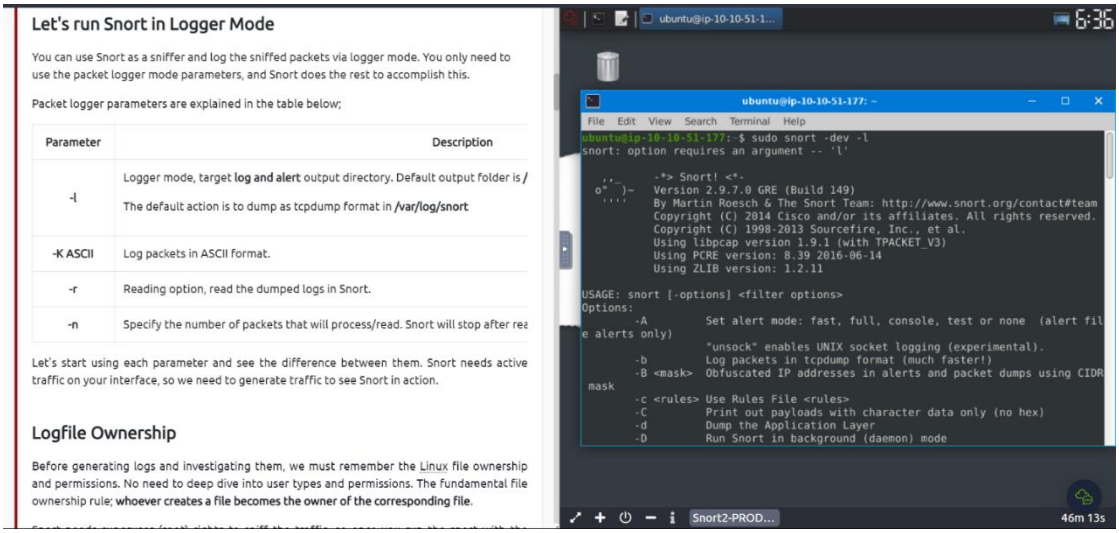


Step 5: Sniffing with parameter "-X" full packet dump mode (-X)



Packet Logger Mode

Step 6: Logging with parameter "-l"



Let's run Snort in Logger Mode

You can use Snort as a sniffer and log the sniffed packets via logger mode. You only need to use the packet logger mode parameters, and Snort does the rest to accomplish this.

Packet logger parameters are explained in the table below;

Parameter	Description
-l	Logger mode, target **log and alert** output directory. Default output folder is / The default action is to dump as tcpdump format in **/var/log/snort**
-K ASCII	Log packets in ASCII format.
-r	Reading option, read the dumped logs in Snort.
-n	Specify the number of packets that will process/read. Snort will stop after rea

Let's start using each parameter and see the difference between them. Snort needs active traffic on your interface, so we need to generate traffic to see Snort in action.

Logfile Ownership

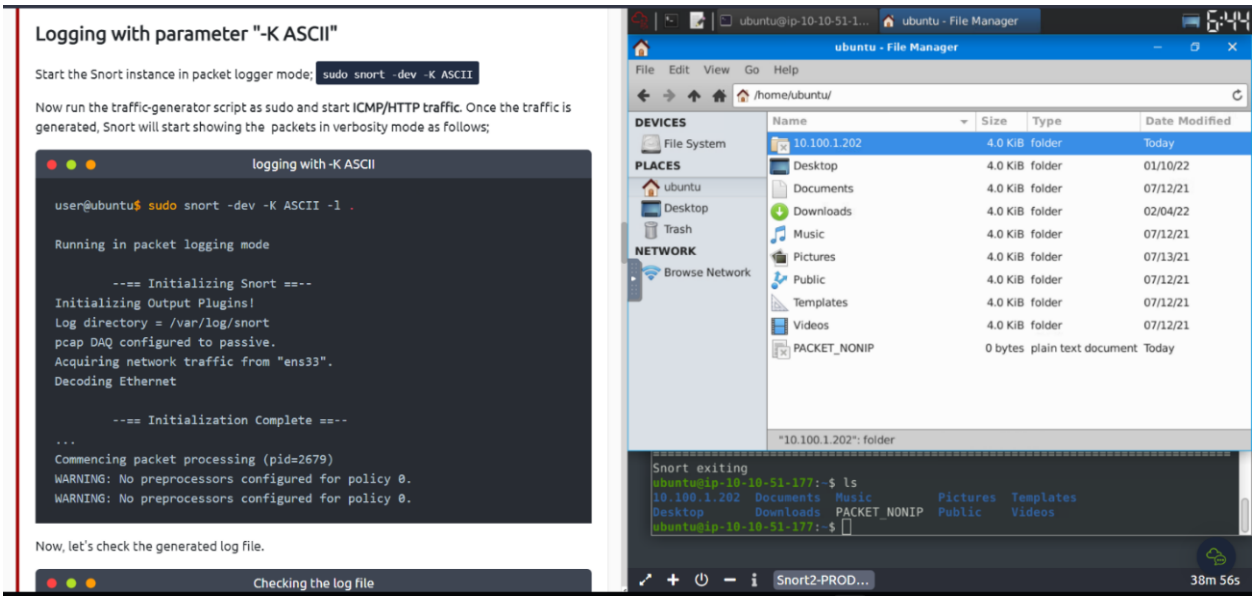
Before generating logs and investigating them, we must remember the Linux file ownership and permissions. No need to deep dive into user types and permissions. The fundamental file ownership rule; **whoever creates a file becomes the owner of the corresponding file.**

```
ubuntu@ip-10-10-51-177: ~
File  Edit  View  Search  Terminal  Help
ubuntu@ip-10-10-51-177:~$ sudo snort -dev -l
snort: option requires an argument -- 'l'

        -*> Snort! <*-
 o"  )~   Version 2.9.7.0 GRE (Build 149)
  ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
          Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
          Copyright (C) 1998-2013 Sourcefire, Inc., et al.
          Using libpcap version 1.9.1 (with TPACKET_V3)
          Using PCRE version: 8.39 2016-06-14
          Using ZLIB version: 1.2.11

USAGE: snort [-options] <filter options>
Options:
        -A          Set alert mode: fast, full, console, test or none  (alert fil
e alerts only)
                    "unsock" enables UNIX socket logging (experimental).
        -b          Log packets in tcpdump format (much faster!)
        -B <mask>   Obfuscated IP addresses in alerts and packet dumps using CIDR
 mask
        -c <rules>  Use Rules File <rules>
        -C          Print out payloads with character data only (no hex)
        -d          Dump the Application Layer
        -D          Run Snort in background (daemon) mode
```

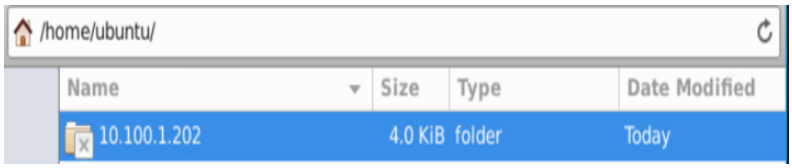
Step 7: Reading log file

```
 Total allocated space (uordblks):      678144
 Total free space (fordblks):           108288
 Topmost releasable block (keepcost):   102304
===============================================================================
Packet I/O Totals:
   Received:         4327
   Analyzed:         4327 (100.000%)
    Dropped:            0 (  0.000%)
   Filtered:            0 (  0.000%)
Outstanding:            0 (  0.000%)
   Injected:            0
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:         4327 (100.000%)
       VLAN:            0 (  0.000%)
        IP4:         4325 ( 99.954%)
       Frag:            0 (  0.000%)
       ICMP:           68 (  1.572%)
        UDP:            2 (  0.046%)
        TCP:         3629 ( 83.869%)
        IP6:            0 (  0.000%)
    IP6 Ext:            0 (  0.000%)
   IP6 Opts:            0 (  0.000%)
      Frag6:            0 (  0.000%)
```

Step 8: Log Packet in ASCII mode

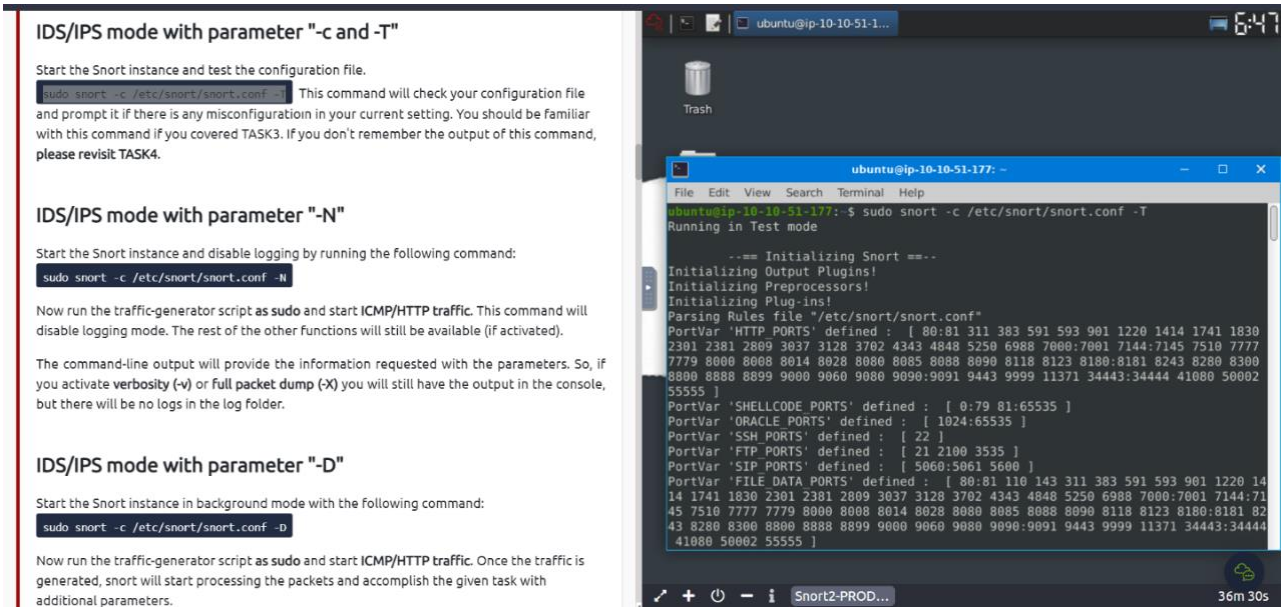


In Code View & Folder View

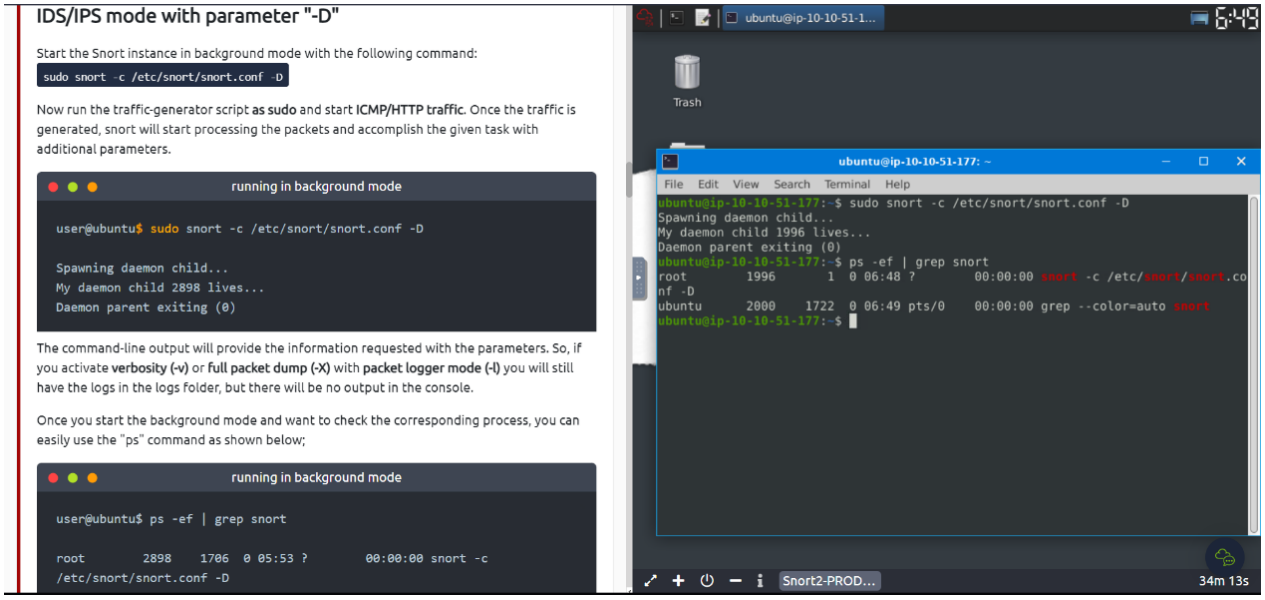


IDS & IPS Mode

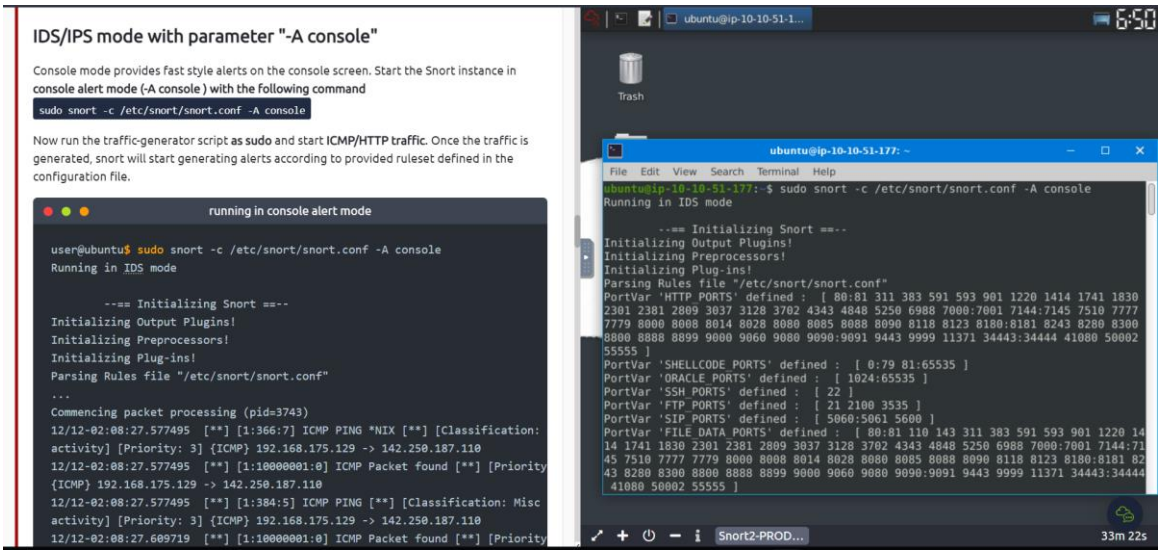
Step 1: Starting snort instance and testing conf file

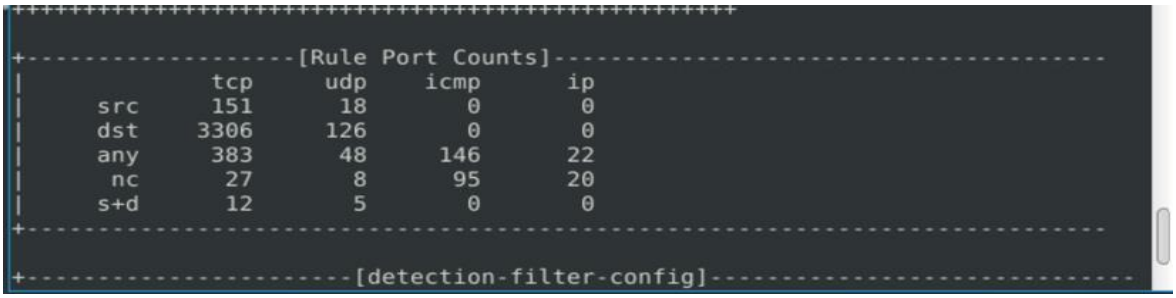


Step 2: Starting snort instance in background mode and checking process with ps command

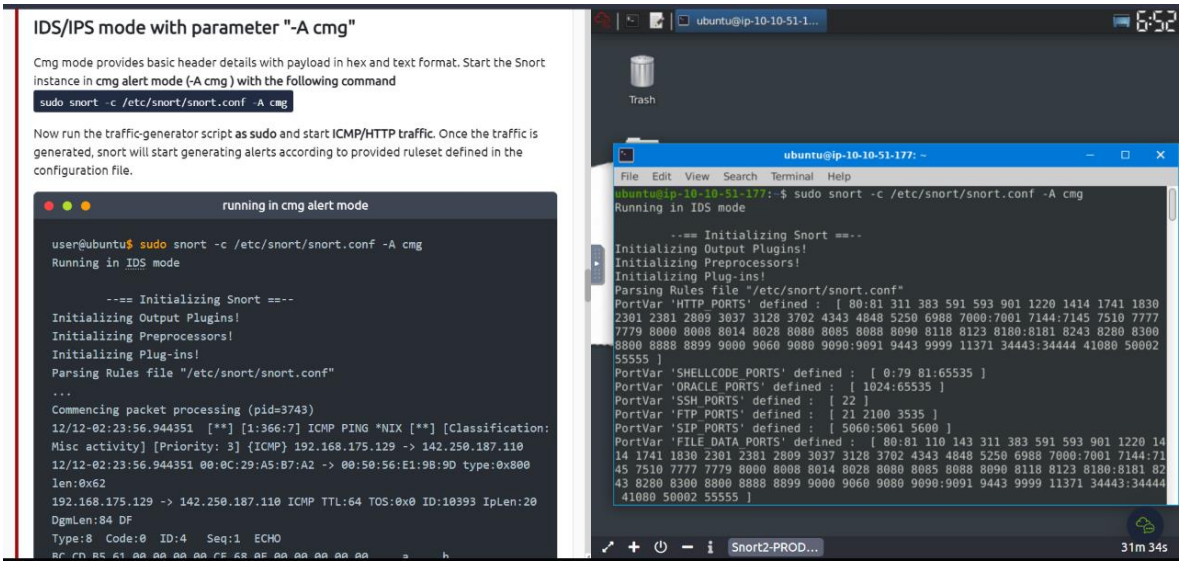


Step 3: Console alert mode

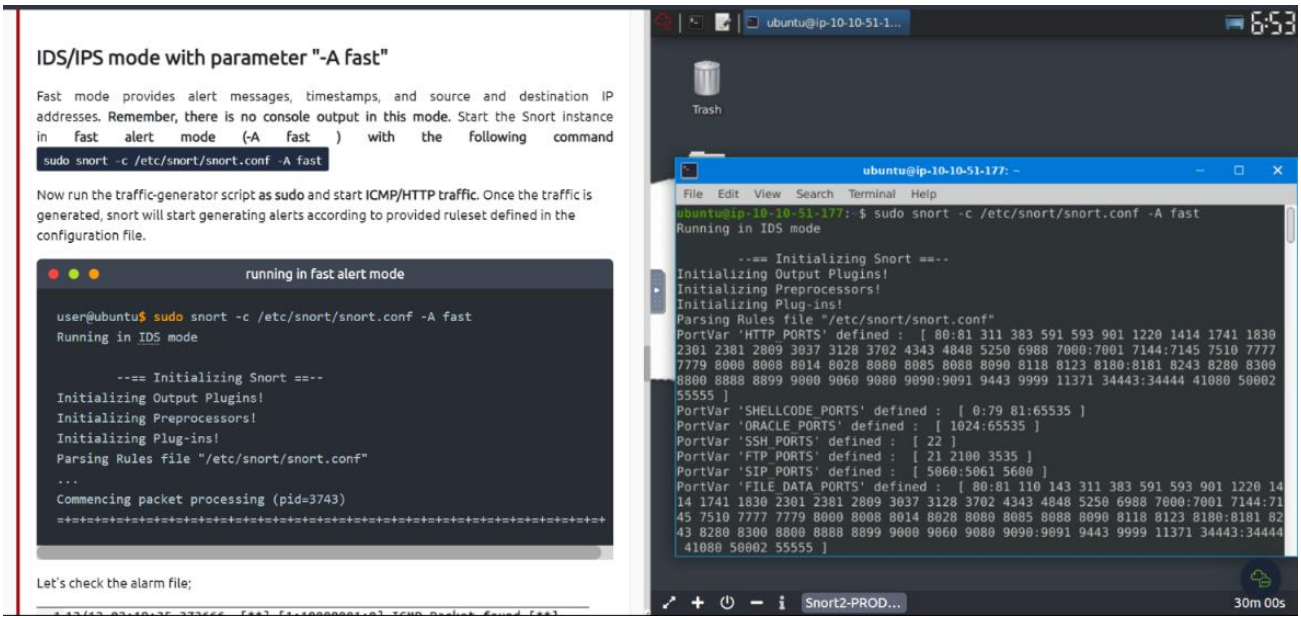




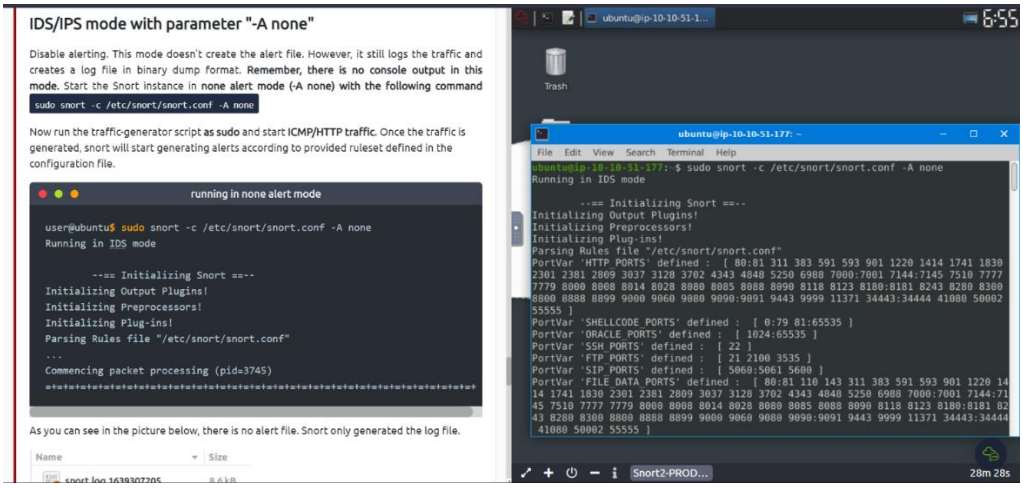
Step 4: cmg mode provides basic header details with payload in hex and text format



Step 5: Fast mode provides alert messages, timestamps, source, and destination IP address



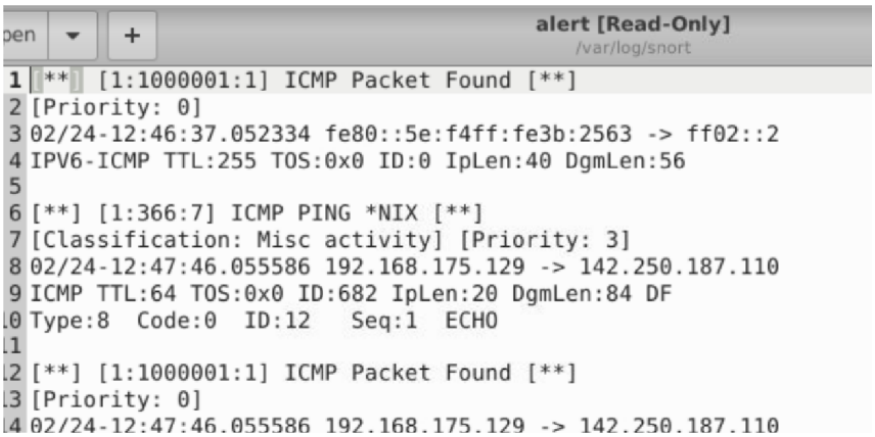
Step 6: -A none does not create the alert file but creates log file



Created Log Files

Name	Size	Type	Date Mod
alert	36.2 KiB	plain text document	Today
snort.log	198 bytes	application log	Today
snort.log.1677243122	0 bytes	plain text document	Today
snort.log.1677243619	0 bytes	plain text document	Today
snort.log.1677243859	0 bytes	plain text document	Today

Alert File



Conclusion/Summary:

Student Signature & Date	**Marks:**	**Evaluator Signature & Date**

Practical 6

Date: / /2023

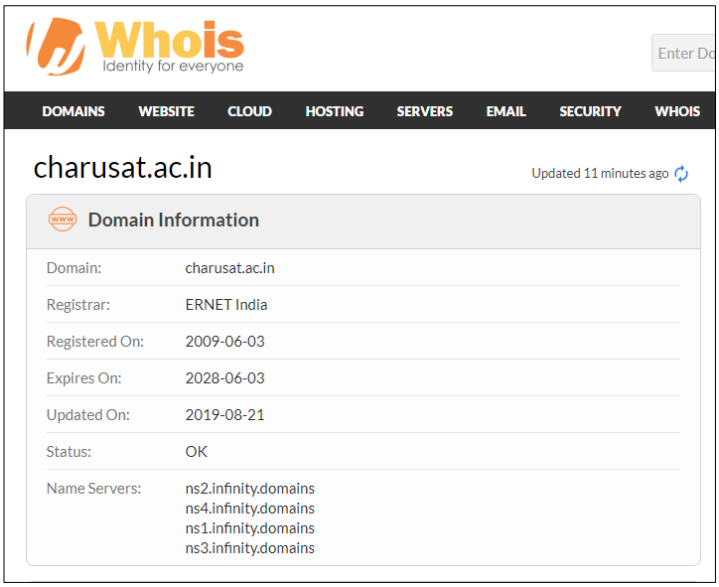
Aim: Implementation to gather information from any PC's connected to the LAN using whois, port scanners, network scanning, IP scanners etc.

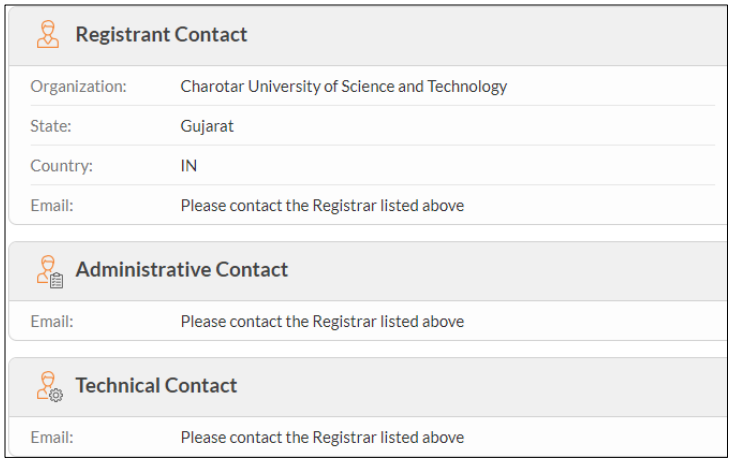
Solution:

1) IP scanner- whois:

'Whois' is a widely used Internet record listing that identifies who owns a domain and how to get in contact with them. Here for example we have searched for domain name charusat.ac.in https://www.charusat.ac.in/

It displays domain information like the register date, update date, registrar provider etc.





2) Port scanners

Port Scanner is an application that is used to determine the open ports on the network. Port scanning is performed to get information about open ports that are ready to receive information.

Port Scanning is a five-step process as described below.

Step 1: For port scanning, there is a need for active hosts. Active hosts can be discovered using the network scanning process.

Step 2: These active hosts are mapped to their IP addresses.

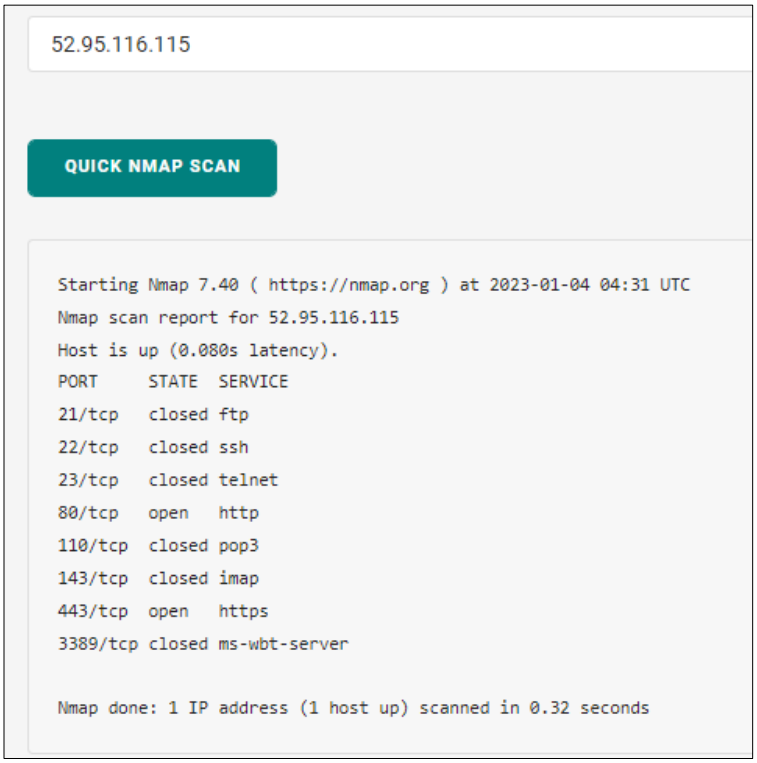
Step 3: Now we have active hosts and thus port scanning process is performed. In this process, packets are sent to specific ports on a host.

Step 4: Here responses will get analyzed.

Step 5: Through this analysis, information about running services will be learned and potential vulnerabilities will be identified.

2.1) Nmap Online port scanner:

Here we have searched for open ports for amazon.in using it's IP address.



3) Nikto:

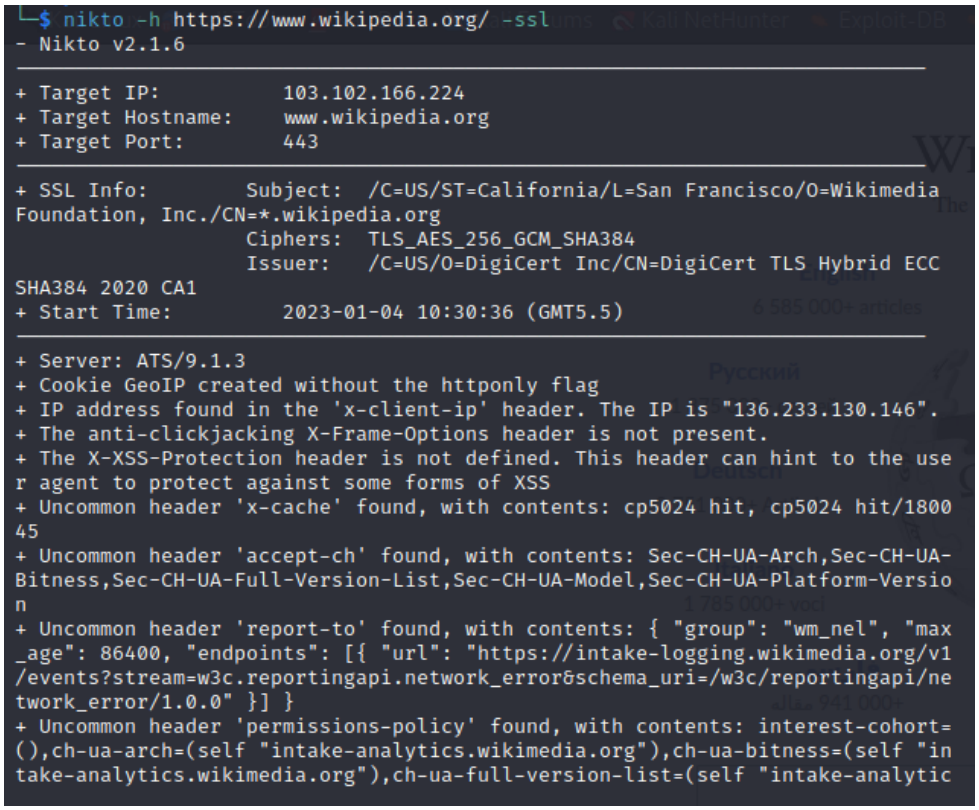
Nikto is an open source web server and web application scanner. Nikto can perform comprehensive tests against web servers for multiple security threats, including over 6700 potentially dangerous files/programs. Nikto can also perform checks for outdated web servers software, and version-specific problems.

To perform a simple domain scan, use the -h (host) flag:

nikto -h scanme.nmap.org

For domains with HTTPS enabled, you have to specify the -ssl flag to scan p:

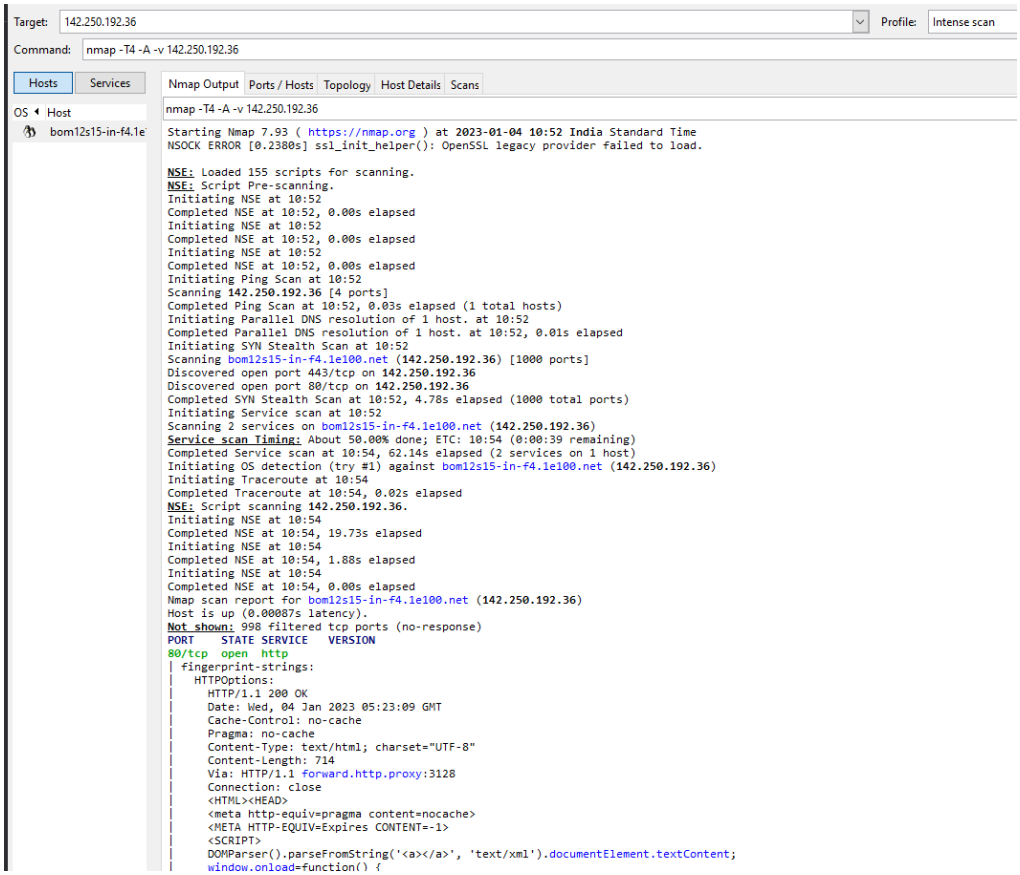
Here we have scanned Wikipedia.org



4) Zenmap:

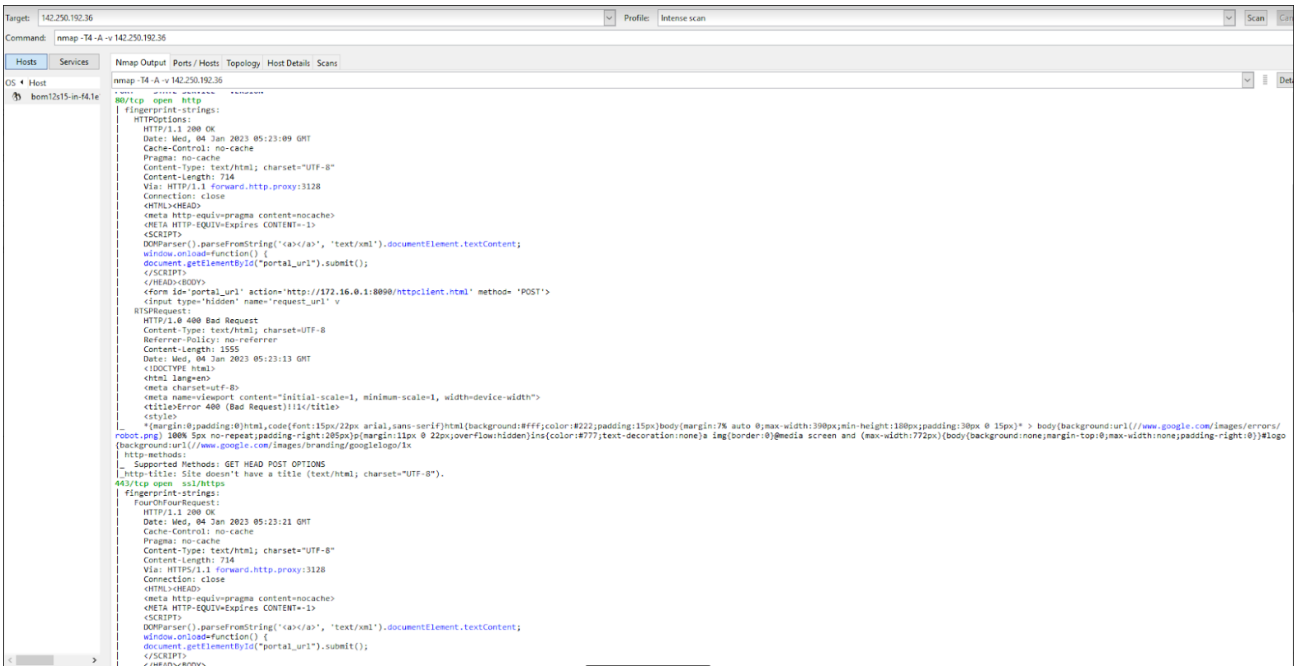
Zenmap is the official Nmap Security Scanner GUI. It is a multi-platform (Linux, Windows, Mac OS X, BSD, etc.) free and open source application which aims to make Nmap easy for beginners to use while providing advanced features for experienced Nmap users. Frequently used scans can be saved as profiles to make them easy to run repeatedly. A command creator allows interactive creation of Nmap command lines. Scan results can be saved and viewed later. Saved scan results can be compared with one another to see how they differ.

We have used it to scan for network and ip addresses

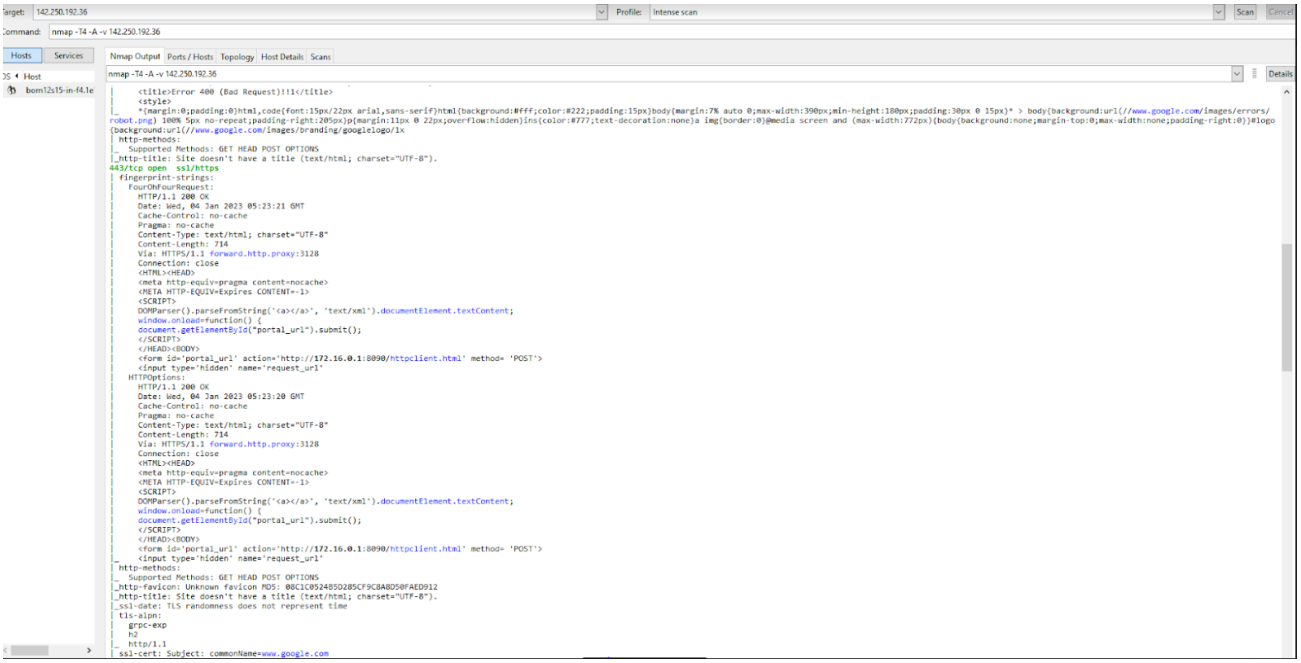


On Scanning we found two open ports and their information

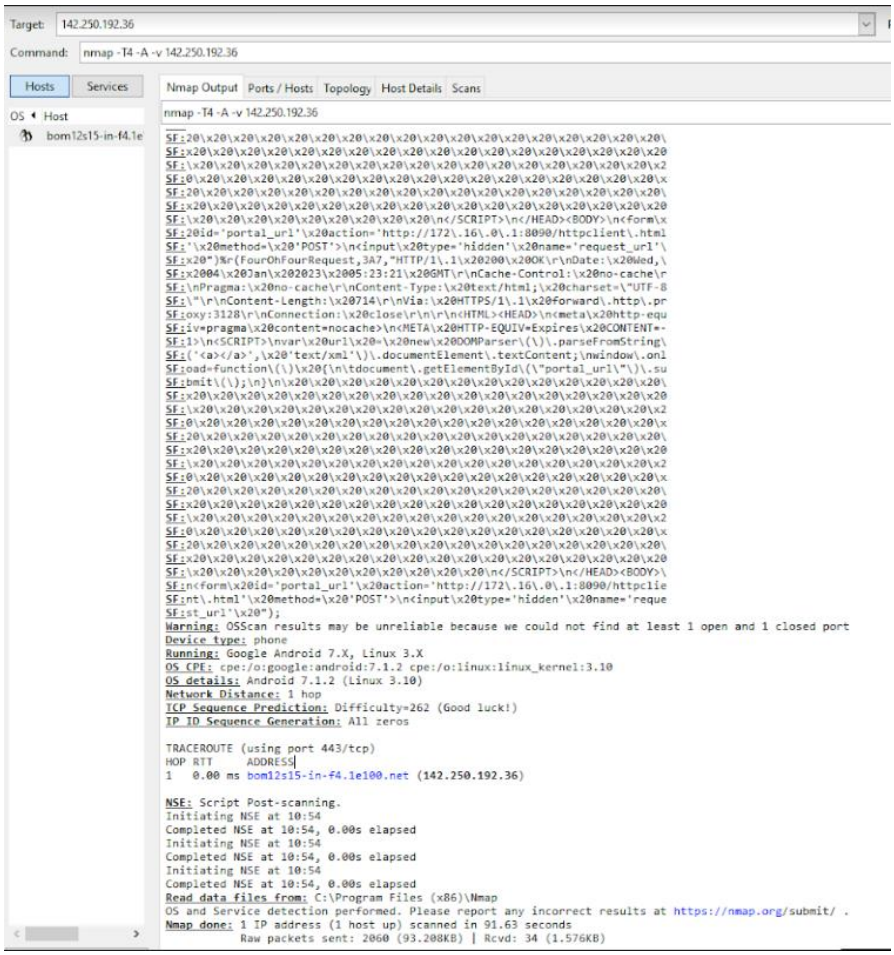
Port1



Port 2



Once the Scan is over



Conclusion/Summary:

Student Signature & Date	**Marks:**	**Evaluator Signature & Date**

Practical 7

Date: / /2023

Aim: Set up a Virtual lab environment with Windows XP (SP1), Metasploitable OS, and BRICKS/DVWA web server and an Attacker machine (KALI/BT) in virtual machines (network in NAT mode). Now carry out Vulnerability assessment in environment

a. Network VA/PT

i. Find the open ports in domain.

ii. Find out the hosts in domains.

iii. Find out the services running on domains and their versions.

iv. Banner Grabbing of server.

v. Find out default vulnerabilities in Services.

vi. Exploit the vulnerabilities.

vii. Deploy and maintain the backdoor.

b. Web VA/PT

i. Find the domain information.

ii. Find the details of server and its default vulnerabilities.

iii. Perform automated testing using BurpSuite or ZAP proxies.

Tools: nmap, netcat, netcraft, nslookup, whois, dig, ping, Nessus, Metasploit, FOCA.

Theory:

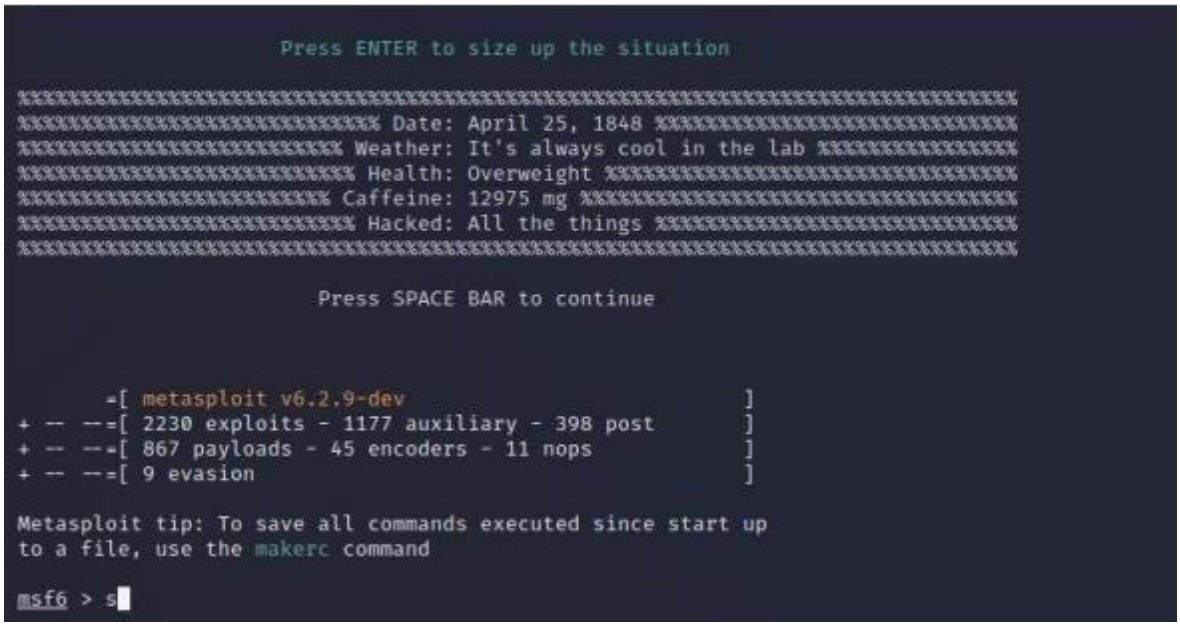
METASPLOIT:

- Metasploit is one of the best penetration testing frameworks that help a business find out and shore up vulnerabilities in their systems before exploitation by hackers. To put it simply, Metasploit allows hacking with permission.
- A Metasploit penetration test begins with the information gathering phase, wherein Matsploit integrates with various reconnaissance tools like Nmap, SNMP scanning, and Windows patch enumeration, and Nessus to find the vulnerable spot in your system.
- Once the weakness is identified, choose an exploit and payload to penetrate the chink in the armor.
- If the exploit is successful, the payload gets executed at the target, and the user gets a shell to interact with the payload. One of the most popular payloads to attack Windows systems is Meterpreter – an in-memory-only interactive shell.

Once on the target machine, Metasploit offers various exploitation tools for privilege escalation, packet sniffing, pass the hash, keyloggers, screen capture, plus pivoting tools. Users can also set up a persistent backdoor if the target machine gets rebooted.

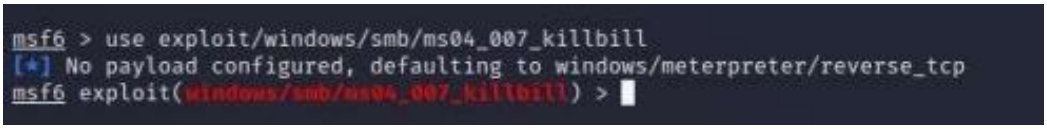
Implementation:

Step1: Start metasplot



Step 2: Find the vulnerability using nessus tool in windows xp.
Step 3: Search the vulnerability. Command: search ms04-007

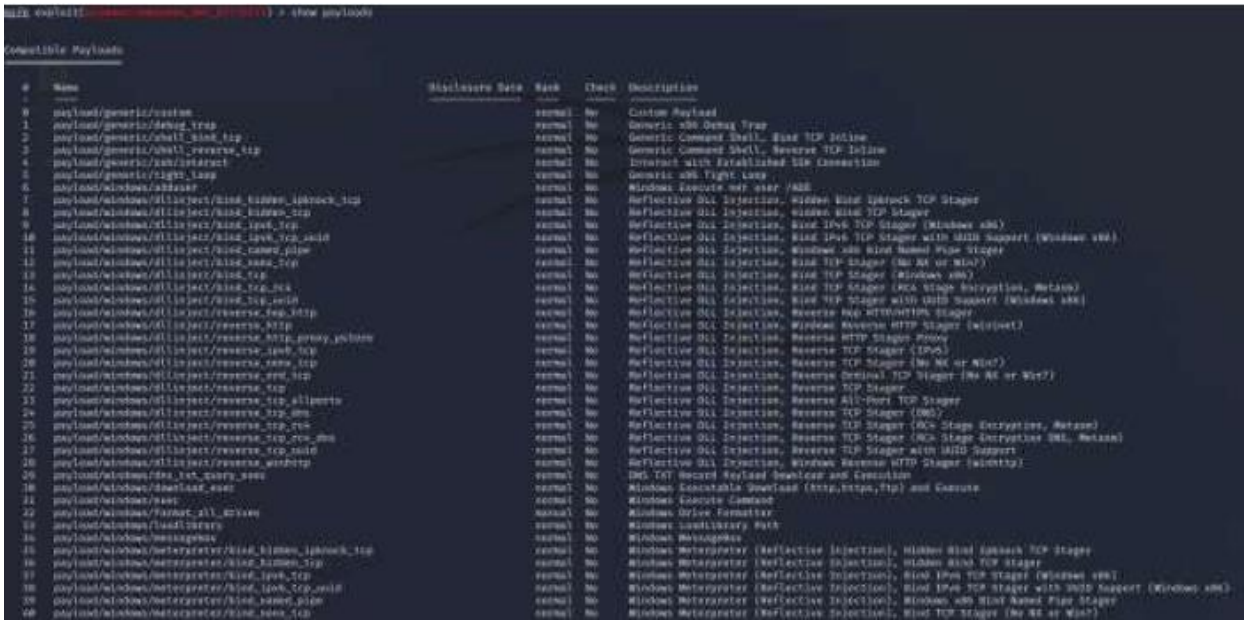
Step 4: Now, use the path of exploit.
Command: use exploit/windows/smd/ms04_007_killbill



Step 5: List out the option. Command: show options

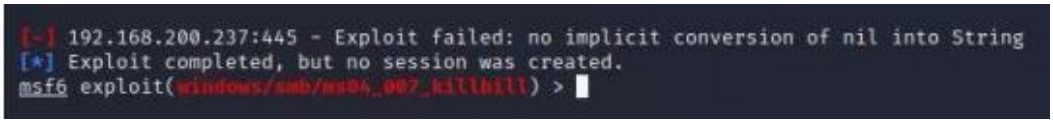
Step 6: Set the RHOSTS by using the IP of windows.
Command: set RHOSTS IP_address

Step 7: Now, to set payload, we have find the index of payload using the following command.
Command: show payloads



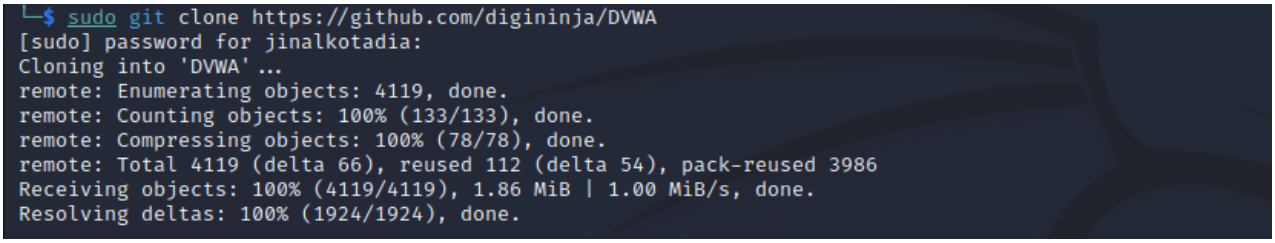
Step 8: Setting Payload
Command: set payload 106

Step 9: Final step is to perform exploit.
Command: exploit



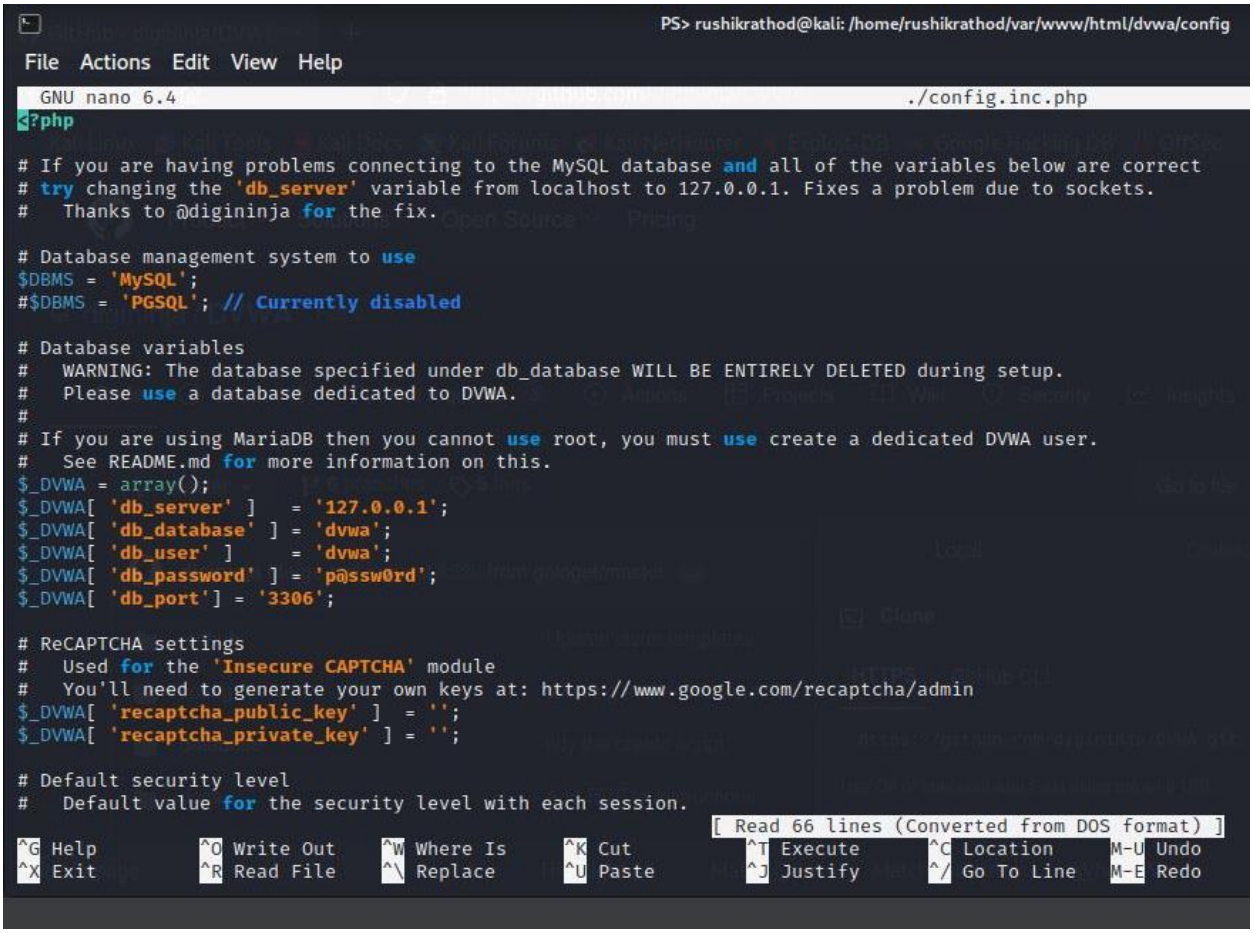
b. Web VA/PT

Step 1: Download Damn vulnerable web application (DVWA)

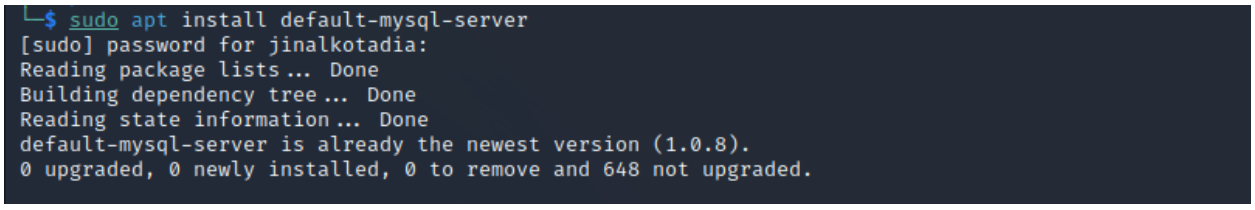


Step 2: Configure DVWA using flowing commands.

1. Sudo chmod –R 777 dvwa/
2. cd dvwa/config
3. sudo cp config.inc.php.dist config.inc.php
4. sudo nano config.inc.php



Step 3: Install Mysql on Kali in new cmd then start mysql.



Step 4: Configure MySql database

```
└─$ systemctl status mysql
● mariadb.service - MariaDB 10.6.10 database server
     Loaded: loaded (/lib/systemd/system/mariadb.service; disabled; preset: disabled)
     Active: active (running) since Wed 2023-02-15 02:16:09 EST; 7min ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
    Process: 52760 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/mysqld (code=>
    Process: 52762 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code>
    Process: 52764 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && VAR= ||   VAR=`cd >
    Process: 52810 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (cod>
    Process: 52812 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
   Main PID: 52793 (mariadbd)
     Status: "Taking your SQL requests now..."
      Tasks: 8 (limit: 4584)
     Memory: 99.7M
        CPU: 1.328s
     CGroup: /system.slice/mariadb.service
             └─52793 /usr/sbin/mariadbd

Feb 15 02:16:09 kali mariadbd[52793]: 2023-02-15  2:16:09 0 [Note] InnoDB: Loading buffer pool(s) >
Feb 15 02:16:09 kali mariadbd[52793]: 2023-02-15  2:16:09 0 [Warning] You need to use --log-bin to>
Feb 15 02:16:09 kali mariadbd[52793]: 2023-02-15  2:16:09 0 [Note] InnoDB: Buffer pool(s) load com>
Feb 15 02:16:09 kali mariadbd[52793]: 2023-02-15  2:16:09 0 [Note] Server socket created on IP: '1>
Feb 15 02:16:09 kali mariadbd[52793]: 2023-02-15  2:16:09 0 [Note] /usr/sbin/mariadbd: ready for c>
Feb 15 02:16:09 kali mariadbd[52793]: Version: '10.6.10-MariaDB-1+b1'  socket: '/run/mysqld/mysqld>
Feb 15 02:16:09 kali systemd[1]: Started MariaDB 10.6.10 database server.
Feb 15 02:16:09 kali /etc/mysql/debian-start[52814]: Upgrading MySQL tables if necessary.
Feb 15 02:16:09 kali /etc/mysql/debian-start[52826]: Checking for insecure root accounts.
Feb 15 02:16:09 kali /etc/mysql/debian-start[52830]: Triggering myisam-recover for all MyISAM tabl>
lines 1-28/28 (END)
```

```
└─$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.6.10-MariaDB-1+b1 Debian n/a

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create user 'user'@'127.0.0.1'identified by 'pass';
Query OK, 0 rows affected (0.009 sec)

MariaDB [(none)]> grant all privileges on dvwa.* to 'user'@'127.0.0.1' identified by 'pass';
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]> exit
Bye
```

Step 5: Install php and extensions

```
└─$ sudo apt -y install lsb-release apt-transport-https ca-certificates
[sudo] password for jinalkotadia:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
lsb-release is already the newest version (12.0-1).
lsb-release set to manually installed.
ca-certificates is already the newest version (20211016).
The following additional packages will be installed:
  apt apt-utils libapt-pkg6.0
Suggested packages:
  apt-doc aptitude | synaptic | wajig
The following NEW packages will be installed:
  apt-transport-https
The following packages will be upgraded:
  apt apt-utils libapt-pkg6.0
3 upgraded, 1 newly installed, 0 to remove and 1338 not upgraded.
Need to get 2,594 kB of archives.
After this operation, 28.7 kB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 libapt-pkg6.0 amd64 2.5.5 [903 kB]
```

```
└─$ sudo wget -O /etc/apt/trusted.gpg.d/php.gpg https://packages.sury.org/php/apt.gpg
--2023-02-15 02:50:01--  https://packages.sury.org/php/apt.gpg
Resolving packages.sury.org (packages.sury.org) ... 185.40.106.117
Connecting to packages.sury.org (packages.sury.org)|185.40.106.117|:443 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 1769 (1.7K) [application/octet-stream]
Saving to: '/etc/apt/trusted.gpg.d/php.gpg'

/etc/apt/trusted.gpg.d/p 100%[=========================================>]   1.73K  --.-KB/s    in 0s

2023-02-15 02:50:02 (19.5 MB/s) - '/etc/apt/trusted.gpg.d/php.gpg' saved [1769/1769]
```

```
└─$ sudo apt install php8.1 -y
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages will be upgraded:
  php8.1
1 upgraded, 0 newly installed, 0 to remove and 1346 not upgraded.
Need to get 31.2 kB of archives.
After this operation, 5,120 B of additional disk space will be used.
Get:1 https://packages.sury.org/php buster/main amd64 php8.1 all 8.1.16+repack-1+0~20230214.36+debia
n10~1.gbpb38498 [31.2 kB]
Fetched 31.2 kB in 2s (20.5 kB/s)
(Reading database ... 393836 files and directories currently installed.)
Preparing to unpack .../php8.1_8.1.16+repack-1+0~20230214.36+debian10~1.gbpb38498_all.deb ...
Unpacking php8.1 (8.1.16+repack-1+0~20230214.36+debian10~1.gbpb38498) over (8.1.12-1) ...
Setting up php8.1 (8.1.16+repack-1+0~20230214.36+debian10~1.gbpb38498) ...
```

```
┌──(kali㉿kali)-[~]
└─$ sudo apt install php7.4-{cli,json,imap,bcmath,bz2,intl,gd,mbstring,mysql,zi
p}
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
php7.4-cli is already the newest version (7.4.21-1+deb11u1).
```

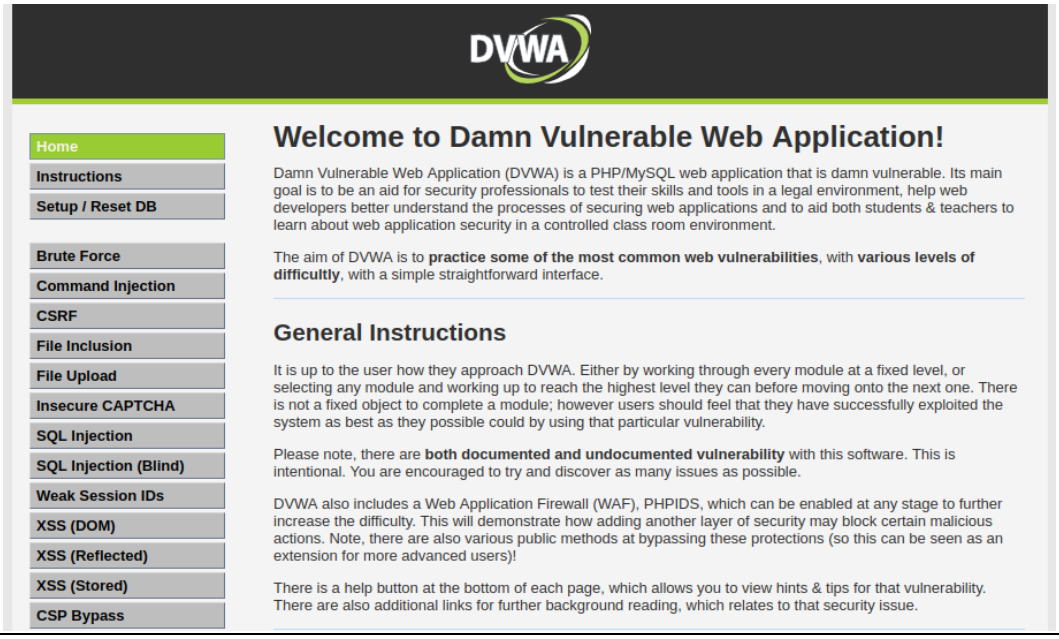
Step 6: Configure in apache server and check status

```
; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
; http://php.net/allow-url-fopen
allow_url_fopen = On

; Whether to allow include/require to open URLs (like http:// or ftp://) as fil
; http://php.net/allow-url-include
allow_url_include = On
```

```
└─$ systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; disabled; preset: disabled)
     Active: active (running) since Wed 2023-02-15 03:12:27 EST; 15s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 67914 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 67935 (apache2)
      Tasks: 6 (limit: 4584)
     Memory: 18.8M
        CPU: 172ms
     CGroup: /system.slice/apache2.service
             ├─67935 /usr/sbin/apache2 -k start
             ├─67937 /usr/sbin/apache2 -k start
             ├─67938 /usr/sbin/apache2 -k start
             ├─67939 /usr/sbin/apache2 -k start
             ├─67940 /usr/sbin/apache2 -k start
             └─67941 /usr/sbin/apache2 -k start

Feb 15 03:12:27 kali systemd[1]: Starting The Apache HTTP Server ...
Feb 15 03:12:27 kali apachectl[67934]: AH00558: apache2: Could not reliably determine the server's >
Feb 15 03:12:27 kali systemd[1]: Started The Apache HTTP Server.
lines 1-20/20 (END)
```

Step 7: Access DVWA on Your Browser

Open firefox and search: http://127.0.0.1/dvwa/setup.php which will open home page for dvwa.



Conclusion/Summary:		
Student Signature & Date	**Marks:**	**Evaluator Signature & Date**

Practical 8

Date: / /2023

Aim: Gather information of any domain/website/IP address using

following Information Gathering Tools.

1. **Samspade**
2. **Nslookup**
3. **Whois**
4. **Tracert**

Solution:

1) Samspade

Systems and security administrators have a number of useful tools at their disposal to obtain information about computers attached to other networks on the Internet, as well as information about the Internet itself. Ping, traceroute, whois and nslookup are among the essential utilities for even rudimentary maintenance and testing. But the native Windows environment includes only a few of these tools and they are, by and large, individual command line utilities and one has to go to third parties to obtain many of the missing utilities. Sam Spade is a nice piece of software that combines many of these common tools -- and several more uncommon ones -- into a single, integrated, Windows-compatible package.

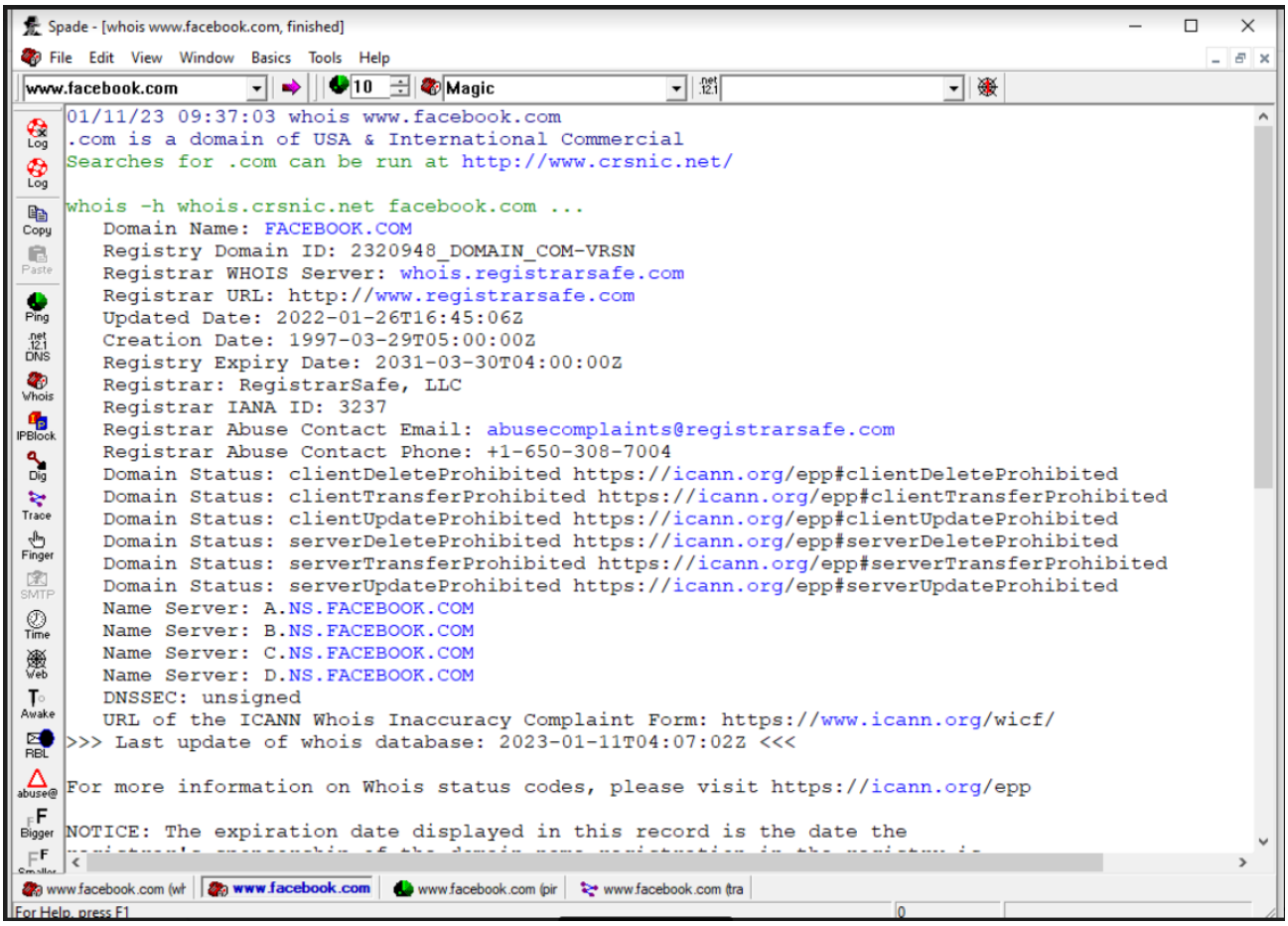
Sam Spade's utilities allow the user to look up information about a remote host or domain, generally for the purpose of initial reconnaissance or forensic analysis:

- *Ping* sends a series of packets to the indicated host to determine if that system is reachable via the network and provides an estimate of the round trip packet time.
- *Traceroute* traces the route that packets take from the user's system to the specified target host address, listing all intermediate routers and showing a graph of the hop-by-hop delay times. Fast and slow traceroute differ only in the number of attempts made to learn the route.
- *Nslookup* and *Decode URL* display the IP address and name of a specified host. This can help an investigator learn about the owner of a system from the domain name or obtain an IP address with which to further investigate the geographic location of a system.
- *Whois* provides ownership and contact information for the specified host's domain. This tool is increasingly convenient as the number of domain name registrars grows. When Network Solutions was the sole registrar for .com, for example, their whois database was the only one

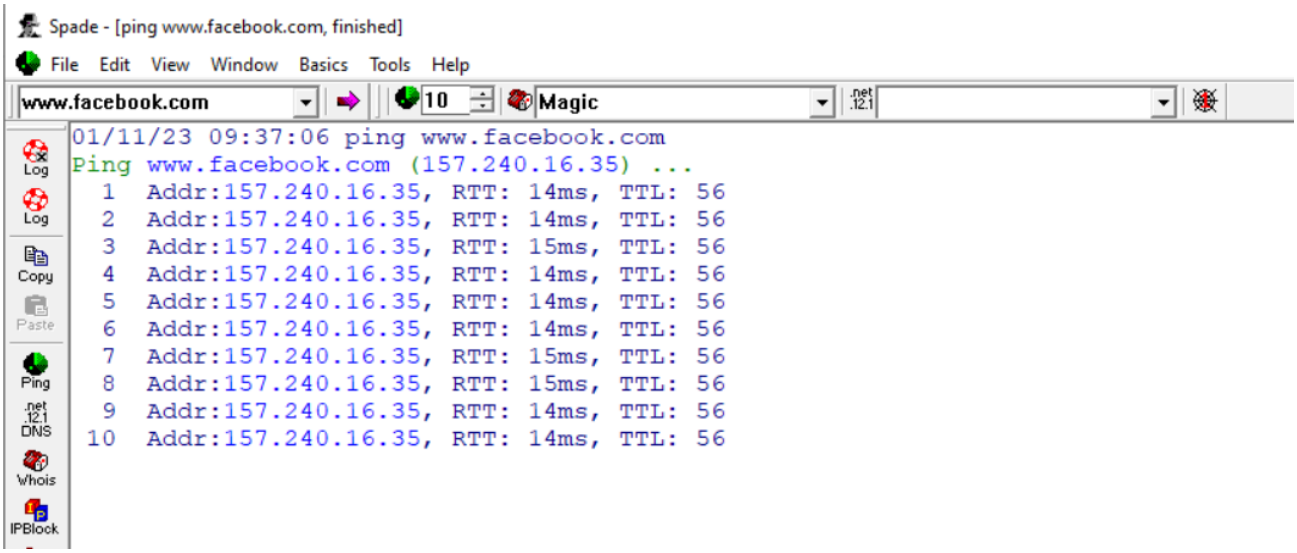
you needed to search. With about 100 accredited registrars today, you have to do a search just to find out which registrar to lookup. Sam Spade's whois function does this for you.

- *IP Block* indicates the owner of the IP address block to which the specified host belongs. By identifying the owner of an address block, you can start to narrow down where a host is geographically located and/or learn about the host's upstream Internet service provider (ISP).
- *DIG (Domain Internet Groper)*, like nslookup, looks up DNS information. Sam Spade's DIG function returns all DNS records associated with a specified host or domain, including the start of authority (SOA), mail exchange (MX) and name server (NS) records. This information allows the user to determine where to send e-mail to a host's domain and how to access the manager of the domain's name space.
- *Zone Transfer* is used to request that a DNS server send all of the information that it has about a given domain. Properly configured DNS servers will not comply with this request as a security precaution, but it will work surprisingly often. This is a great way to test your own name servers.
- *Finger* obtains host/user information from a system running the finger daemon (TCP port 79). Finger is generally (or should be) disabled at a host because it can give an attacker a lot of information about users and/or the host itself, but it isn't always turned off.

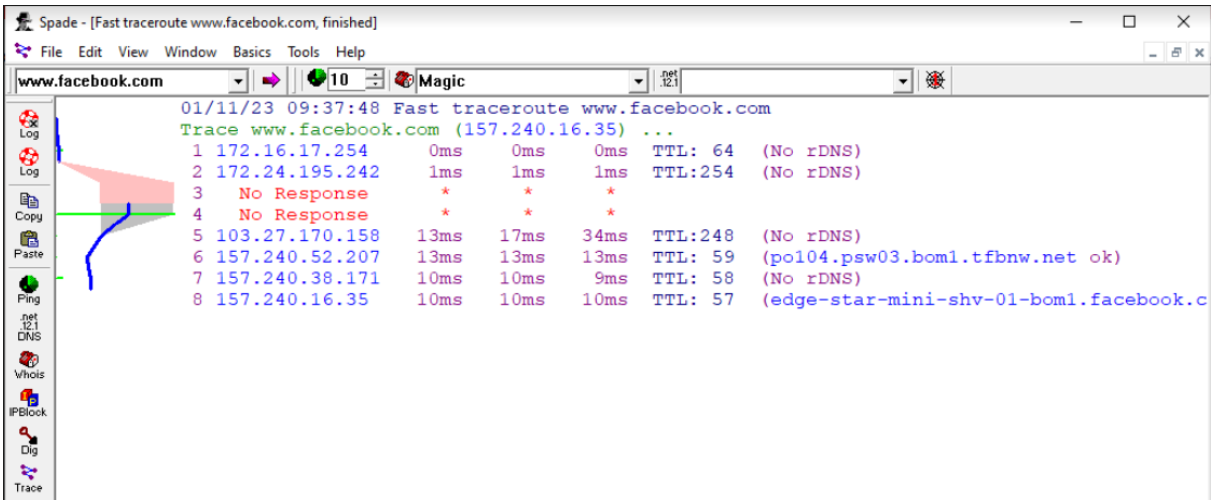
Samspade – who.is



Samspade – ping



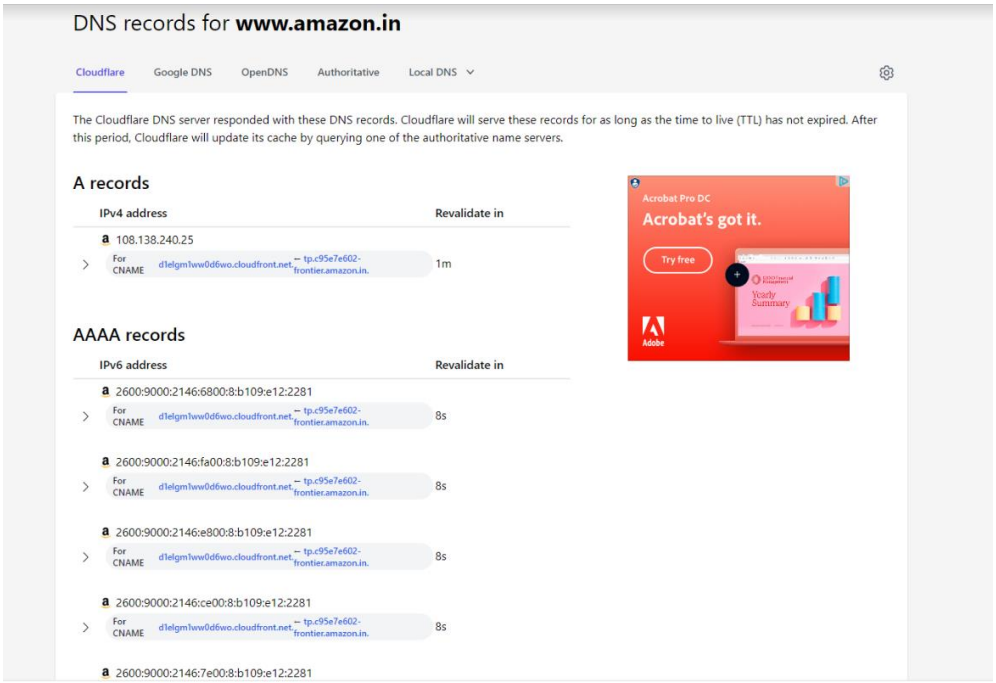
Samspade – tracert



Samespade – addresses

2) NsLookup

nslookup (from name server lookup) is a network administration command-line tool for querying the Domain Name System (DNS) to obtain the mapping between domain name and IP address, or other DNS records.



Whois IP 172.16.17.4

Updated 6 days ago

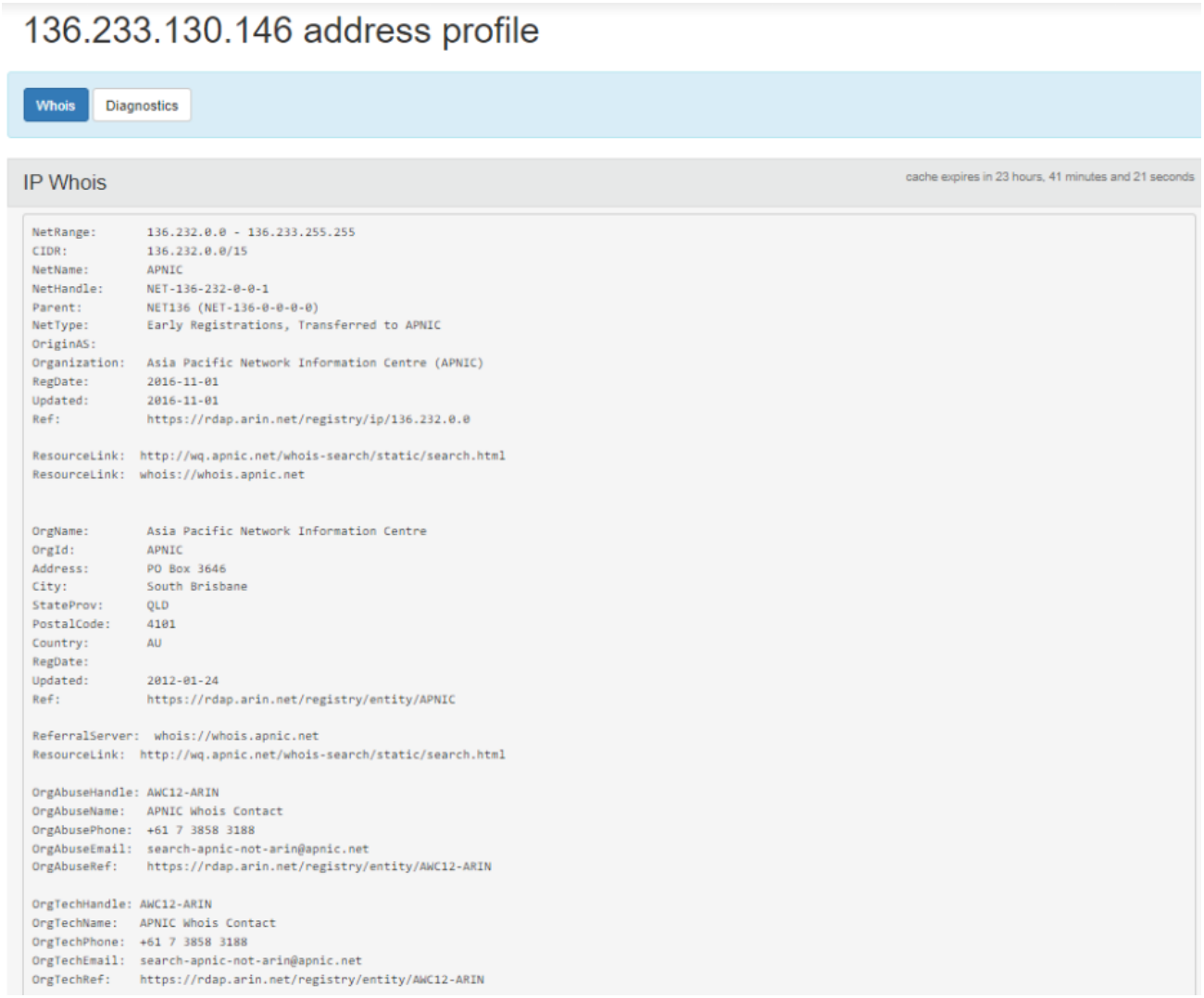
```
#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2023, American Registry for Internet Numbers, Ltd.
#


NetRange:       172.16.0.0 - 172.31.255.255
CIDR:           172.16.0.0/12
NetName:        PRIVATE-ADDRESS-BBLK-RFC1918-IANA-RESERVED
NetHandle:      NET-172-16-0-0-1
Parent:         NET172 (NET-172-0-0-0-0)
NetType:        IANA Special Use
OriginAS:
Organization:   Internet Assigned Numbers Authority (IANA)
RegDate:        1994-03-15
Updated:        2013-08-30
Comment:        These addresses are in use by many millions of independently
Comment:
Comment:        These addresses can be used by anyone without any need to coo
Comment:
Comment:        These addresses were assigned by the IETF, the organization t
Comment:        http://datatracker.ietf.org/doc/rfc1918
Ref:            https://rdap.arin.net/registry/ip/172.16.0.0


OrgName:        Internet Assigned Numbers Authority
OrgId:          IANA
Address:        12025 Waterfront Drive
Address:        Suite 300
City:           Los Angeles
StateProv:      CA
PostalCode:     90292
Country:        US
RegDate:
Updated:        2012-08-31
Ref:            https://rdap.arin.net/registry/entity/IANA


OrgAbuseHandle: IANA-IP-ARIN
OrgAbuseName:   ICANN
OrgAbusePhone:  +1-310-301-5820
OrgAbuseEmail:  abuse@iana.org
OrgAbuseRef:    https://rdap.arin.net/registry/entity/IANA-IP-ARIN

OrgTechHandle: IANA-IP-ARIN
OrgTechName:   ICANN
OrgTechPhone:  +1-310-301-5820
OrgTechEmail:  abuse@iana.org
OrgTechRef:    https://rdap.arin.net/registry/entity/IANA-IP-ARIN


#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2023, American Registry for Internet Numbers, Ltd.
#
```

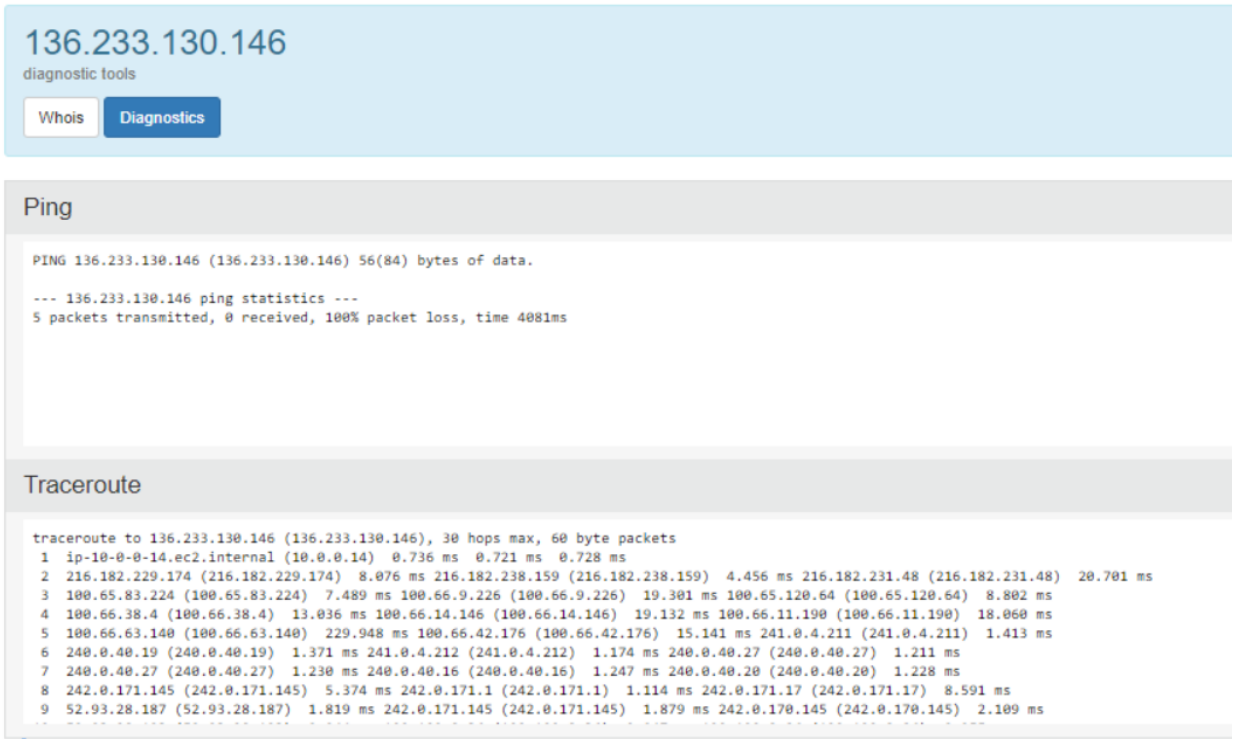
3) whois

Website Information - Search the whois database, look up domain and IP owner information, and check out dozens of other statistics.

On Demand Domain Data- Get all the data you need about a domain and everything associated with that domain anytime with a single search.

Domain names – Register your favorite domain name





4) Tracert

- The TRACERT diagnostic utility determines the route to a destination by sending Internet Control Message Protocol (ICMP) echo packets to the destination. In these packets, TRACERT uses varying IP Time-To-Live (TTL) values. Because each router along the path is required to decrement the packet's TTL by at least 1 before forwarding the packet, the TTL is effectively a hop counter. When the TTL on a packet reaches zero (0), the router sends an ICMP "Time Exceeded" message back to the source computer.

- TRACERT sends the first echo packet with a TTL of 1 and increments the TTL by 1 on each subsequent transmission, until the destination responds or until the maximum TTL is reached. The ICMP "Time Exceeded" messages that intermediate routers send back show the route. Note however that some routers silently drop packets that have expired TTLs, and these packets are invisible to TRACERT.

```
C:\Users\Administrator>tracert 172.16.17.4

Tracing route to host.docker.internal [172.16.17.4]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  host.docker.internal [172.16.17.4]

Trace complete.

C:\Users\Administrator>tracert www.amazon.in

Tracing route to d1elgm1ww0d6wo.cloudfront.net [18.66.54.214]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  172.16.17.254
  2     4 ms     3 ms     3 ms  136.232.113.77.static.jio.com [136.232.113.77]
  3     6 ms     6 ms     8 ms  10.67.49.2
  4     5 ms     4 ms     4 ms  172.18.32.130
  5    10 ms     9 ms    12 ms  10.63.94.141
  6     6 ms     6 ms     6 ms  10.81.155.111
  7     9 ms    10 ms    13 ms  10.81.155.108
  8     6 ms     6 ms     6 ms  10.67.59.163
  9    10 ms     6 ms     5 ms  10.83.244.197
 10     5 ms     5 ms     4 ms  172.29.203.69
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15    13 ms    13 ms    13 ms  server-18-66-54-214.bom78.r.cloudfront.net [18.66.54.214]
```

Conclusion/Summary:

Student Signature & Date	Marks:	Evaluator Signature & Date

Practical 9

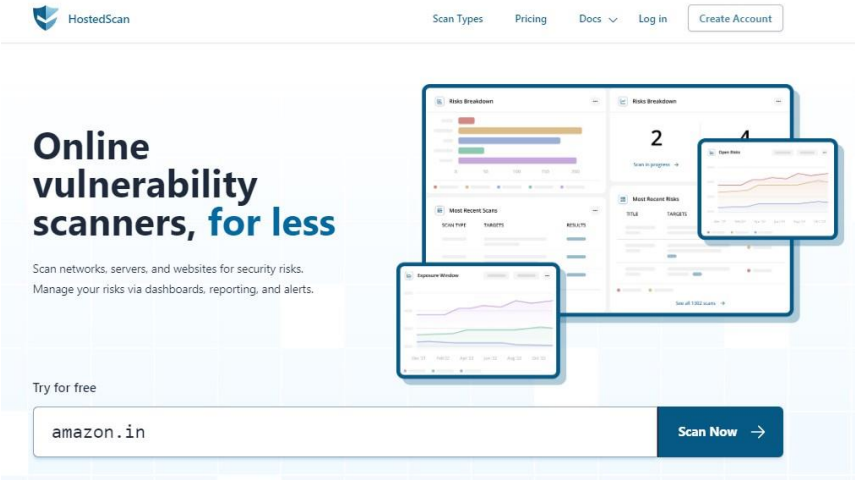
Date: / /2023

Aim: Identify any 5 online web portals for information gathering.Scan an IP address/URL for gathering information. Prepare a report.

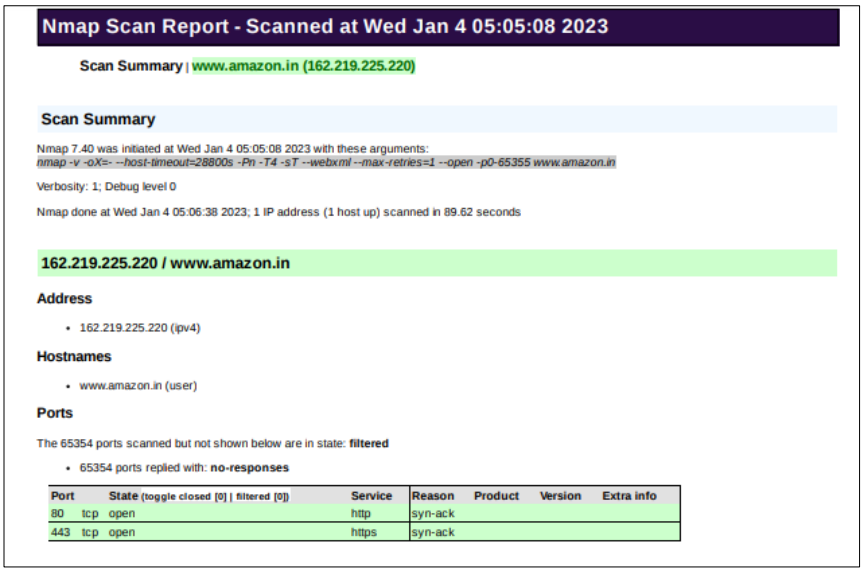
Solution:

1) HostedScan

Online tool for scanning networks, servers, and websites for security risks. To try for free type in the url and your email address, and a detailed report will be sent to your email address.



Nmap Scan for amazon.in



Nmap Scan Report - Scanned at Wed Jan 4 05:05:08 2023

Scan Summary | www.amazon.in (162.219.225.220)

Scan Summary

Nmap 7.40 was initiated at Wed Jan 4 05:05:08 2023 with these arguments:
nmap -v -oX=- --host-timeout=28800s -Pn -T4 -sT --webxml --max-retries=1 --open -p0-65355 www.amazon.in

Verbosity: 1; Debug level 0

Nmap done at Wed Jan 4 05:06:38 2023; 1 IP address (1 host up) scanned in 89.62 seconds

162.219.225.220 / www.amazon.in

Address

- 162.219.225.220 (ipv4)

Hostnames

- www.amazon.in (user)

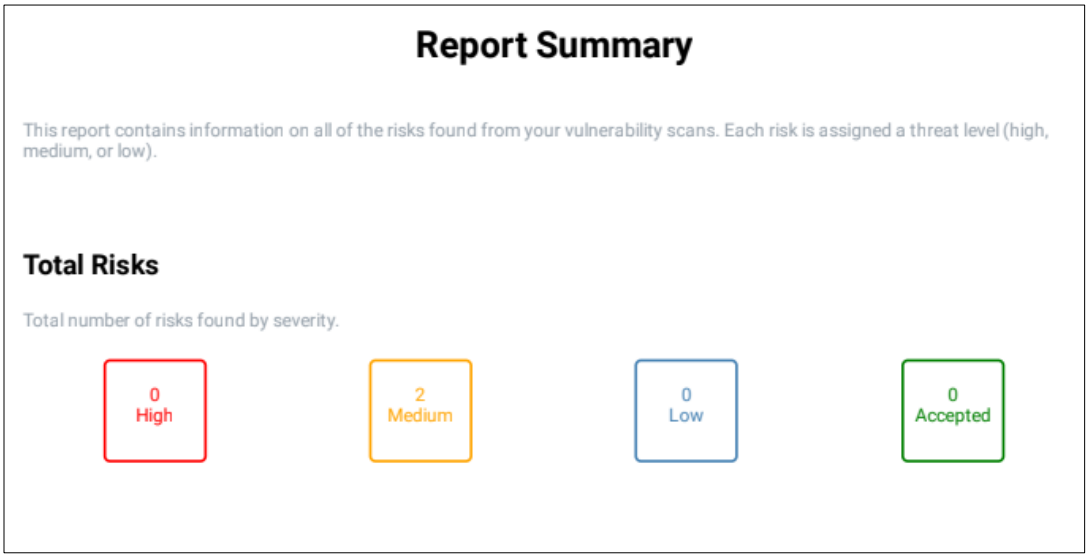
Ports

The 65354 ports scanned but not shown below are in state: **filtered**

- 65354 ports replied with: **no-responses**

Port	State (toggle closed [0] | filtered [0])	Service	Reason	Product	Version	Extra info
80 tcp	open	http	syn-ack			
443 tcp	open	https	syn-ack			

Report summary



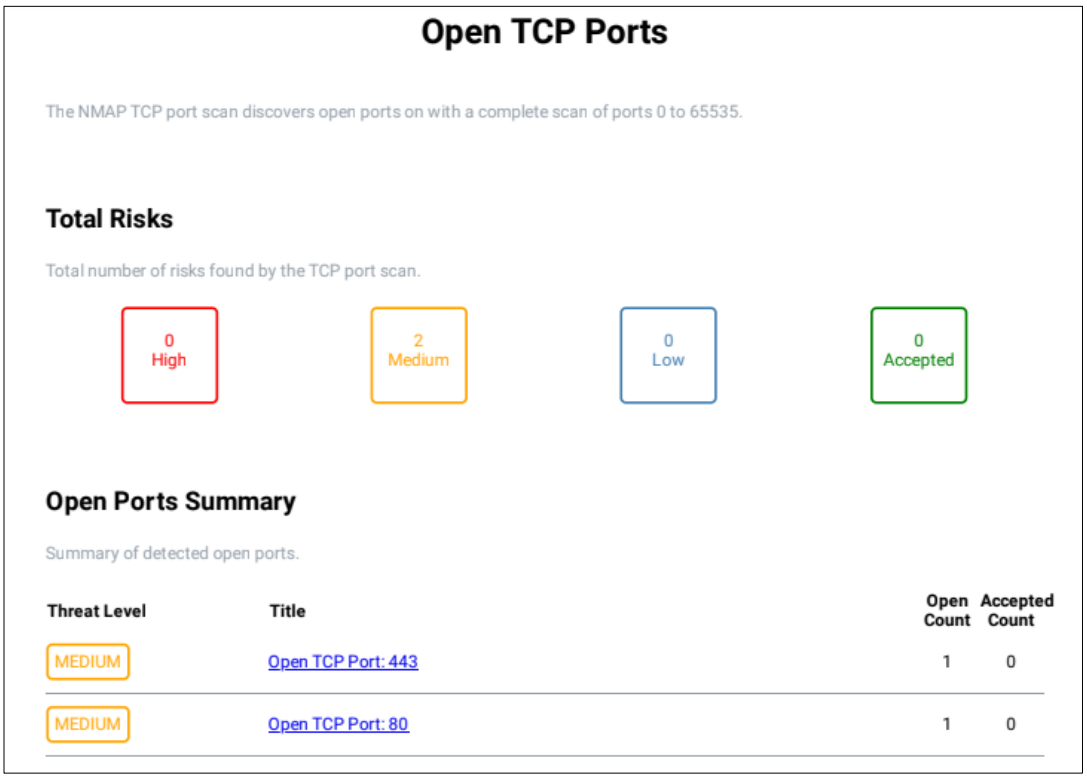
Report Summary

This report contains information on all of the risks found from your vulnerability scans. Each risk is assigned a threat level (high, medium, or low).

Total Risks

Total number of risks found by severity.

0 High	2 Medium	0 Low	0 Accepted



Open TCP Ports

The NMAP TCP port scan discovers open ports on with a complete scan of ports 0 to 65535.

Total Risks

Total number of risks found by the TCP port scan.

0 High	2 Medium	0 Low	0 Accepted

Open Ports Summary

Summary of detected open ports.

Threat Level	Title	Open Count	Accepted Count
MEDIUM	Open TCP Port: 443	1	0
MEDIUM	Open TCP Port: 80	1	0

2) Port scanners

Port Scanner is an application that is used to determine the open ports on the network. Port scanning is performed to get information about open ports that are ready to receive information.

Port Scanning is a five-step process as described below.

Step 1: For port scanning, there is a need for active hosts. Active hosts can be discovered using the network scanning process.

Step 2: These active hosts are mapped to their IP addresses.

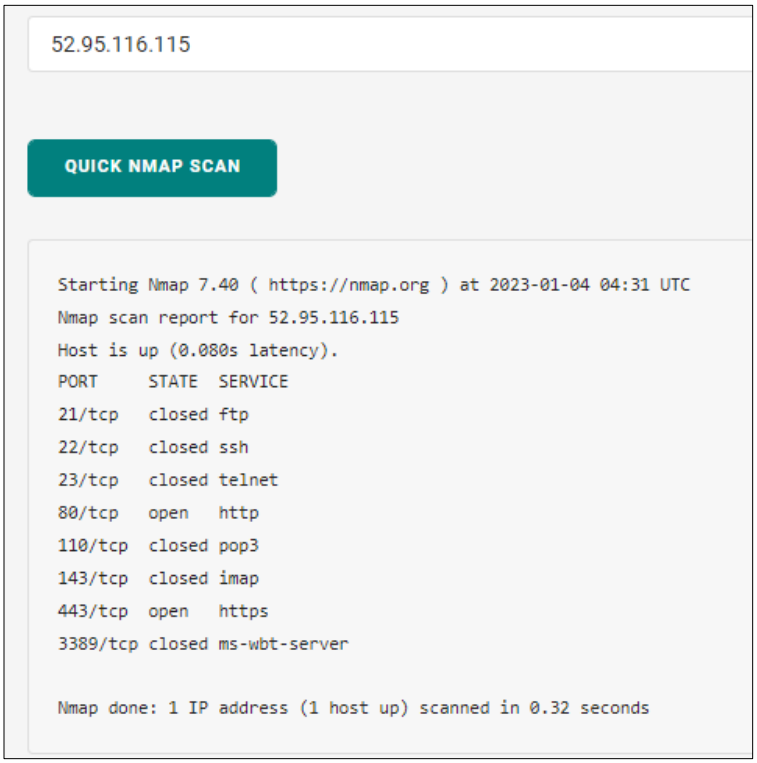
Step 3: Now we have active hosts and thus port scanning process is performed. In this process, packets are sent to specific ports on a host.

Step 4: Here responses will get analyzed.

Step 5: Through this analysis, information about running services will be learned and potential vulnerabilities will be identified.

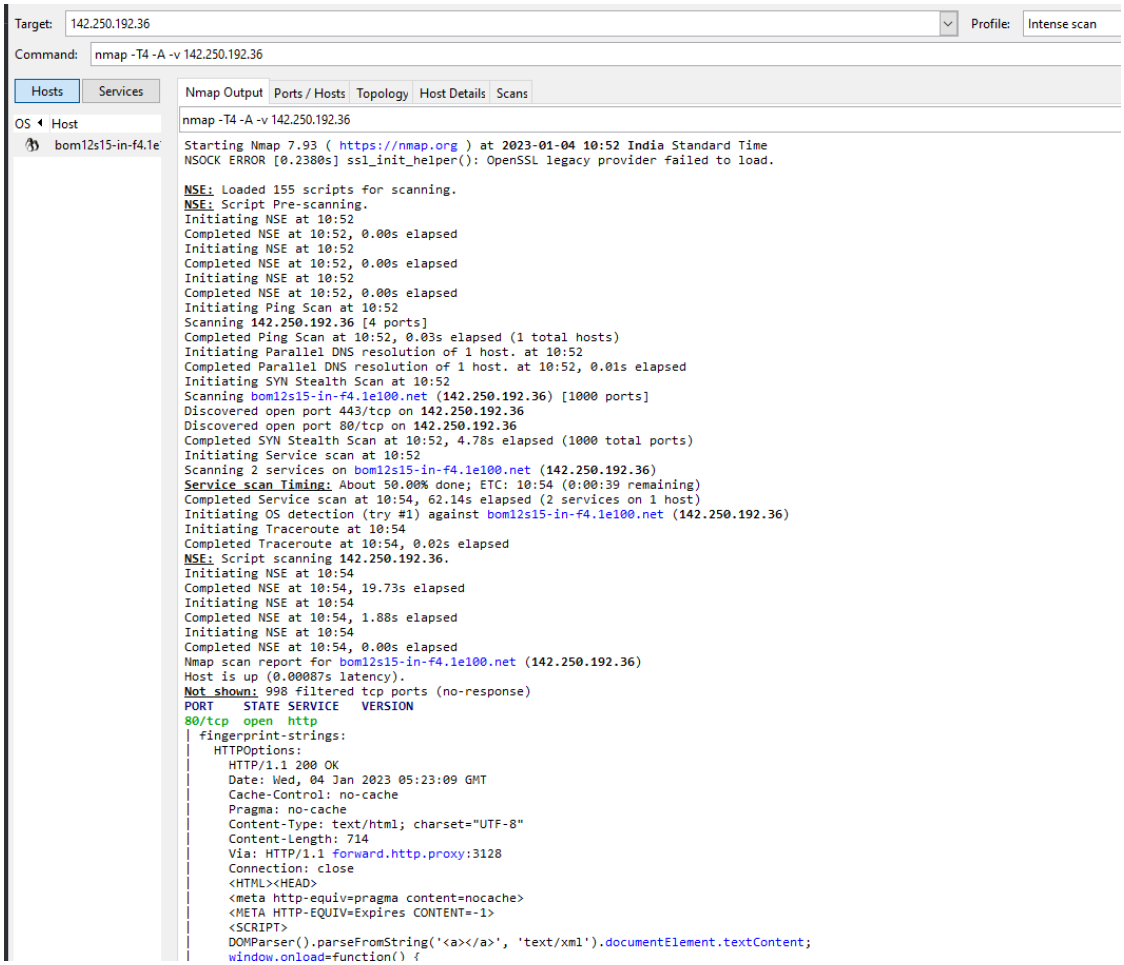
2.1) Nmap Online port scanner:

Here we have searched for open ports for amazon.in using it's IP address.



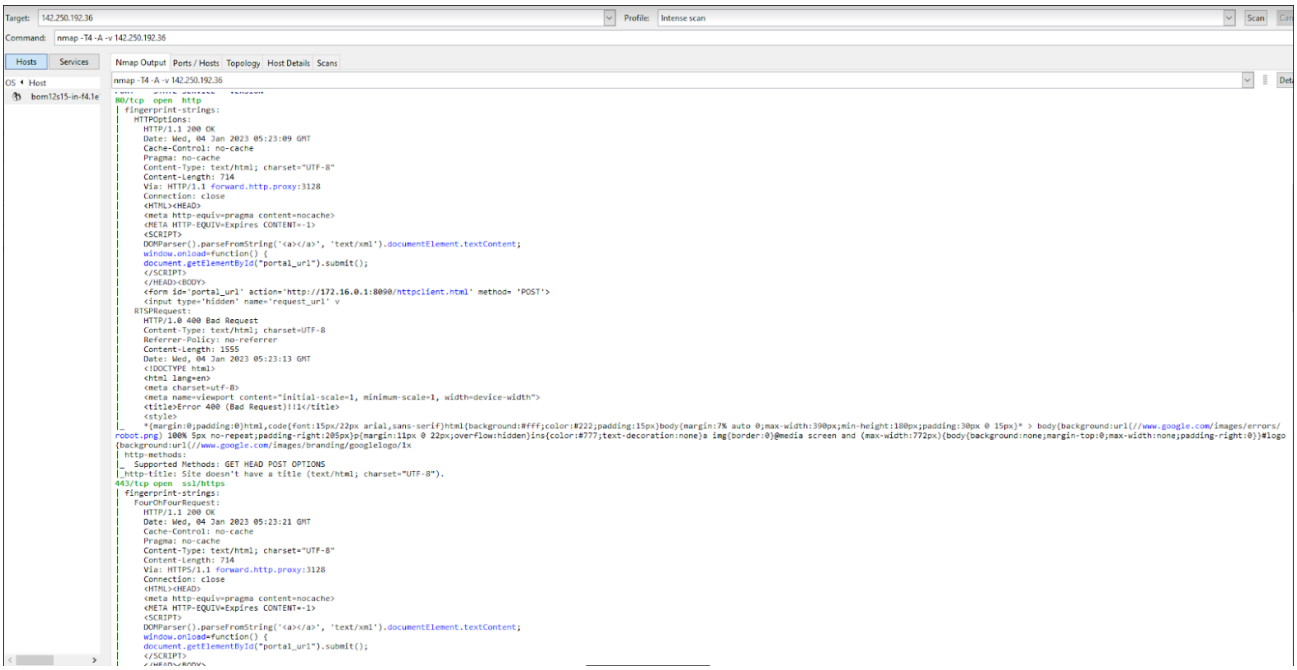
3) Zenmap:

- Zenmap is the official Nmap Security Scanner GUI. It is a multi-platform (Linux, Windows, Mac OS X, BSD, etc.) free and open source application which aims to make Nmap easy for beginners to use while providing advanced features for experienced Nmap users. Frequently used scans can be saved as profiles to make them easy to run repeatedly. A command creator allows interactive creation of Nmap command lines. Scan results can be saved and viewed later. Saved scan results can be compared with one another to see how they differ. The results of recent scans are stored in a searchable database.

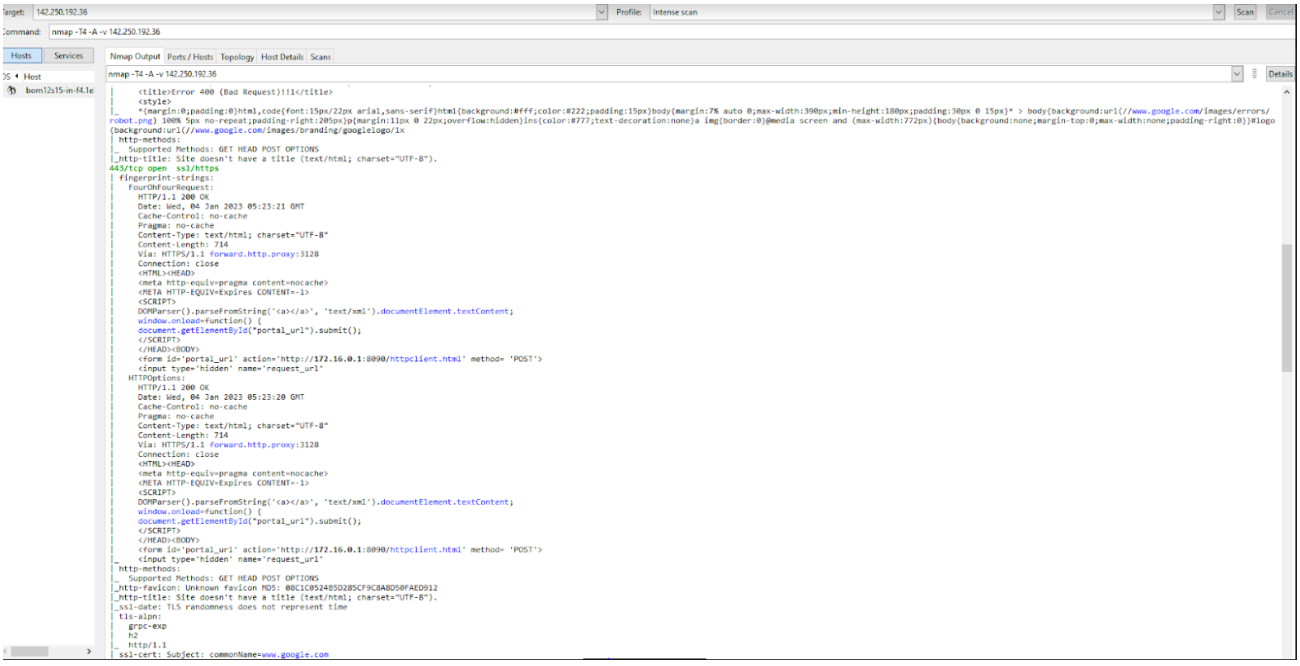


On Scanning we found two open ports and their information

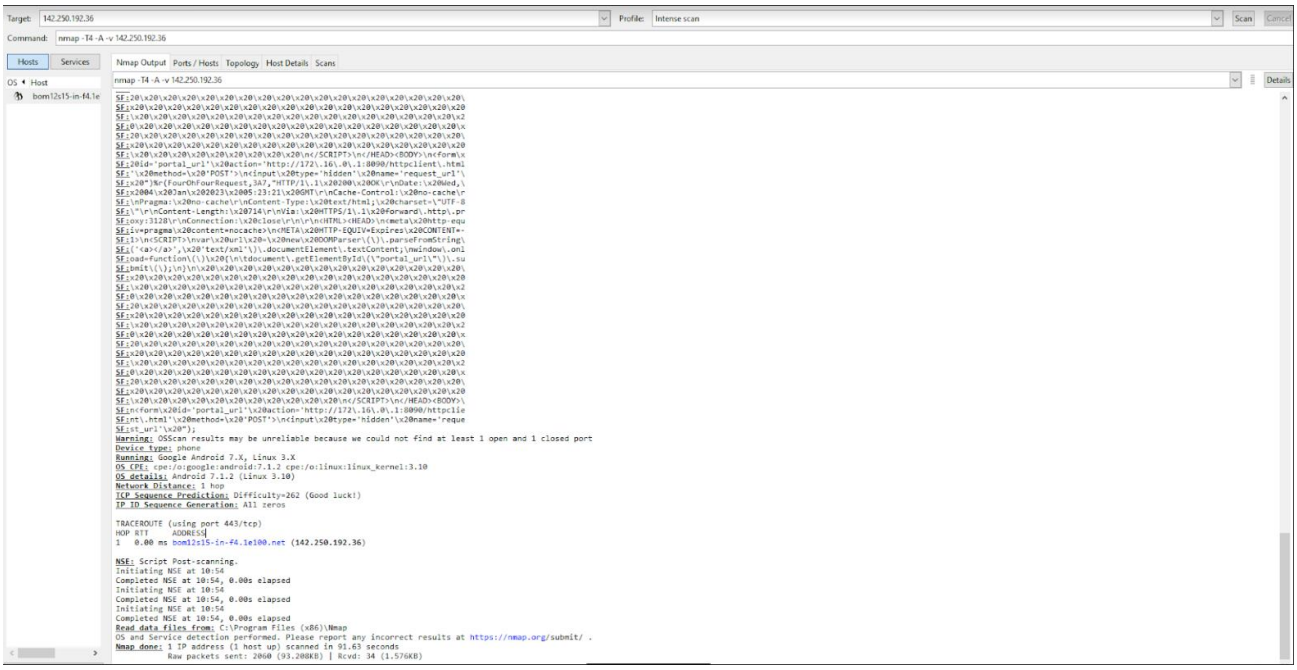
Port1



Port 2

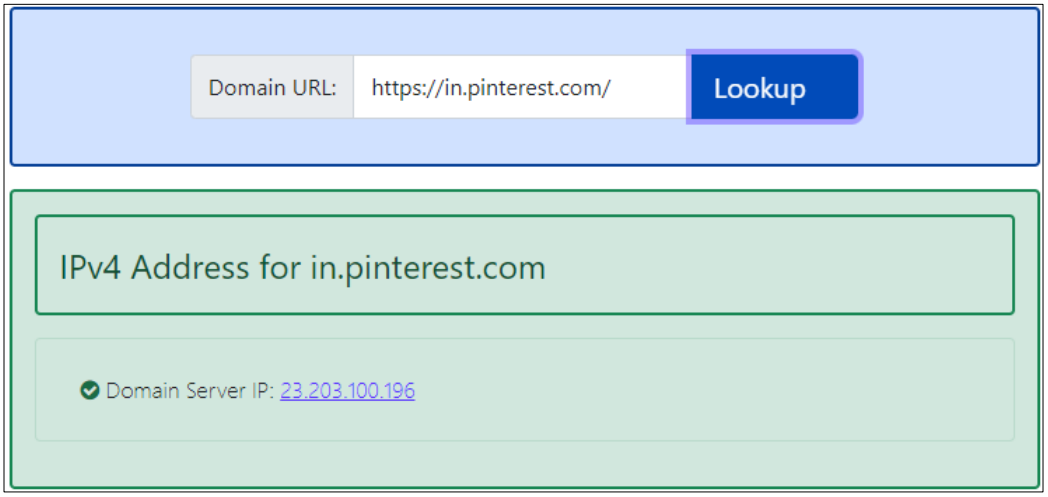


Once the Scan is over



4) whatismyip.com

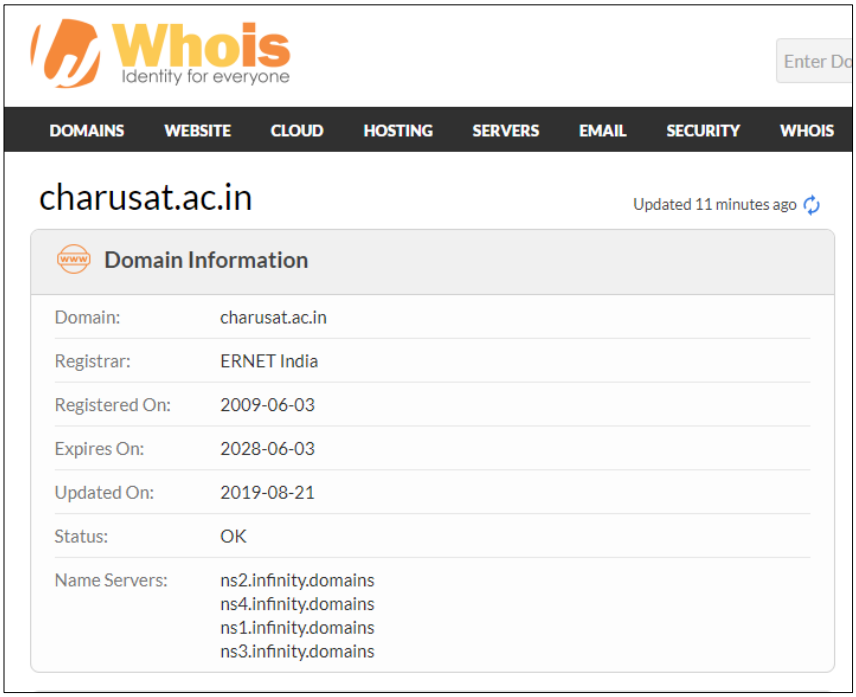
An online tool used to find Ip address of any URL or your own Ip address.

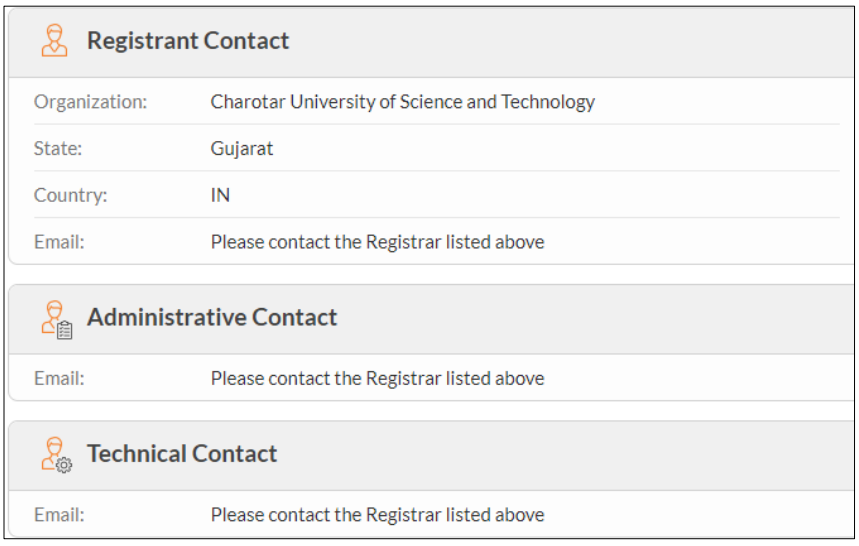


5) IP scanner- whois:

Whois is a widely used Internet record listing that identifies who owns a domain and how to get in contact with them. Here for example we have searched for domain name https://www.charusat.ac.in/

It displays domain information like the register date, update date, registrar provider etc..





Conclusion/Summary:		
Student Signature & Date	**Marks:**	**Evaluator Signature & Date**

Practical 10

Date: / /2023

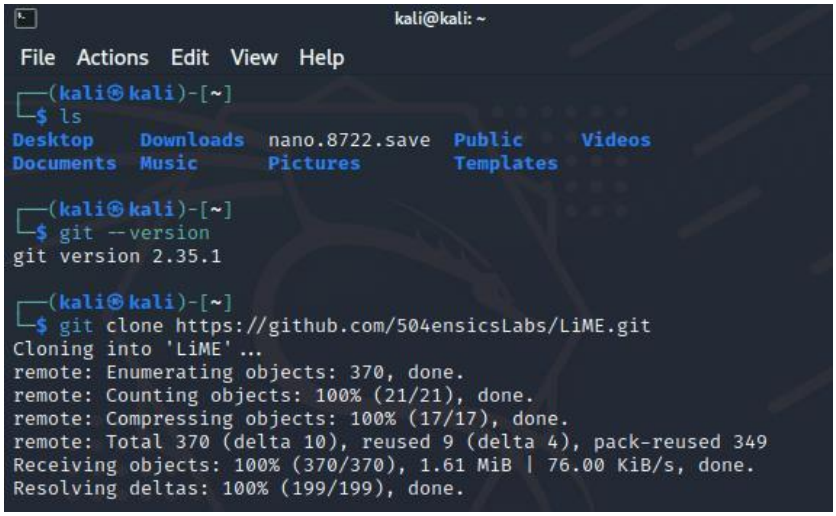
Aim: Perform Live / Memory Analysis on a Linux OS and prepare a detailed report.

Theory:

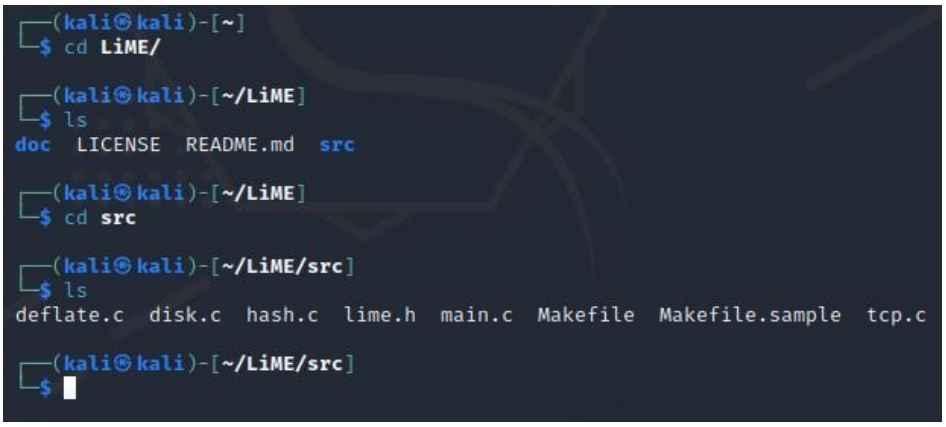
- Memory forensics is a way to find and extract this valuable information from memory. Volatility is an open-source tool that uses plugins to process this type of information. However, there's a problem: Before you can process this information, you must dump the physical memory into a file, and Volatility does not have this ability.

Implementation:

Step 1: Download from https://github.com/504ensicsLabs/LiME



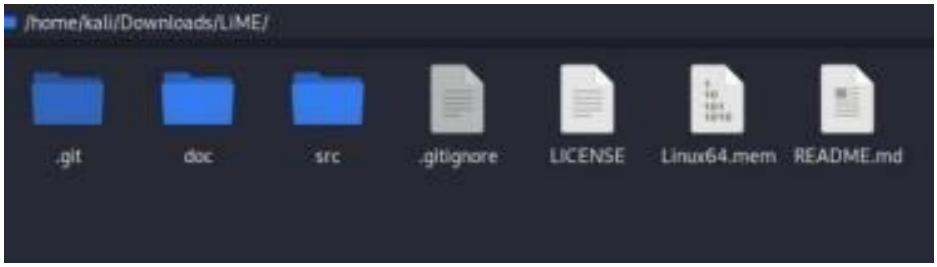
Step 2: Then, install openssh-server



Step 3: Start the ssh service and server and Check for the service status

```
┌──(kali㉿kali)-[~/LiME/src]
└─$ sudo insmod ./lime-5.5.0-kali2-amd64.ko "path=../Linux64.mem format=raw"
[sudo] password for kali:
```

Step 4: Navigate to ssh folder to check the details of the connection.



Conclusion/Summary:

Student Signature & Date	Marks:	Evaluator Signature & Date