



SIR M VISVESVARAYA INSTITUTE OF TECHNOLOGY
(Affiliated to VTU, Recognized by AICTE and Accredited by NBA, NAAC
and an ISO 9001-2008 Certified Institution)
Bengaluru – 562157



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(IoT & Cyber Security including Block Chain Technology)

CYBER SECURITY LAB

BICL404 - IV Semester B.E

(Academic Year 2023-2024)

Compiled and Prepared by:

**Mrs. Roopa H L
Assistant Professor
Dept. of IOT**

Under the Guidance of:

**Dr. Savita Choudary
Professor&Head
Dept. of IOT**

Department Vision and Mission

VISION

To build a center for imparting quality technical education and carrying out research activity to meet the current and future challenges in the domain of Computer Science and Engineering.

MISSION

- The Computer Science and Engineering department strives for excellence in teaching, applying, promoting and imparting knowledge through comprehensive academic curricula.
- Train students to effectively apply the knowledge to solve real-world problems, thus enhance their potential for life-long high-quality career and give them a competitive advantage in the ever-changing and fast paced computing.
- Prepare students to demonstrate a sense of societal and ethical responsibilities in

their professional endeavors.

- Creating amongst students and faculty a collaborative environment open to the free exchange of ideas, which leads to research activity and fuels innovation thinking.

PROGRAM OUTCOMES

PO's	PO Description
PO1	Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4	Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

PSOs	PSO Description
PSO1	An ability to design and analyze algorithms by applying theoretical concepts to build complex and computer- based systems in the domain of System Software, Computer Networks & Security, Web technologies, Data Science and Analytics.
PSO2	Be able to develop various software solutions by applying the techniques of Data Base Management, Complex Mathematical Models, Software Engineering practices and Machine Learning with Artificial Intelligence.

Cyber Security lab		Semester	IV
Course Code	BICL 404	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Credits	01	Exam Hours	100
Examination type (SEE)	Practical		

Course objectives:

- To get Practical exposure of Cyber security threats
- To get Practical exposure on Forensics Tools

SI.NO	Experiments
1	Install Kali Linux and explore basic Linux commands and tools.
2	Perform basic network scanning using the Nmap tool (Zenmap on Windows). Identify services, open ports, active hosts, operating systems, and vulnerabilities.
3	Phishing simulations (Google, LUCY and GoPhish).
4	Packet analysis using Wireshark.
5	Ransomware tabletop exercise on insider threat.
6	Perform SQL injection using BurpSuite
7	Installation of Wire shark, tcpdump, etc and observe data transferred in client server communication using UDP/TCP and identify the UDP/TCP datagram
8	Installation of rootkits and study about the variety of options
9	Perform an Experiment to Sniff Traffic using ARP Poisoning
10	Demonstrate intrusion detection system using snort

Course outcomes (Course Skill Set):

At the end of the course the student will be able to:

- Demonstrate the usage of tools to identify cyber threats/attacks
- Use Autopsy tools for digital forensic.
- Demonstrate Network analysis using Network miner tools.

@# 09042024

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment is to be evaluated for conduction with an observation sheet and recordwrite- up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to **30 marks** (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.
- In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability.
- The marks scored shall be scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course are 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the Head of the Institute.
- The examination schedule and names of examiners are informed to the university before the conduction of the examination. These practical examinations are to be conducted between the schedule mentioned in the academic calendar of the University.
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answerscript to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the

examiners jointly.

- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.

General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics **shall** be decided by the examiners)

Change of experiment is allowed only once and 15% of Marks allotted to the procedure partare to be made zero.

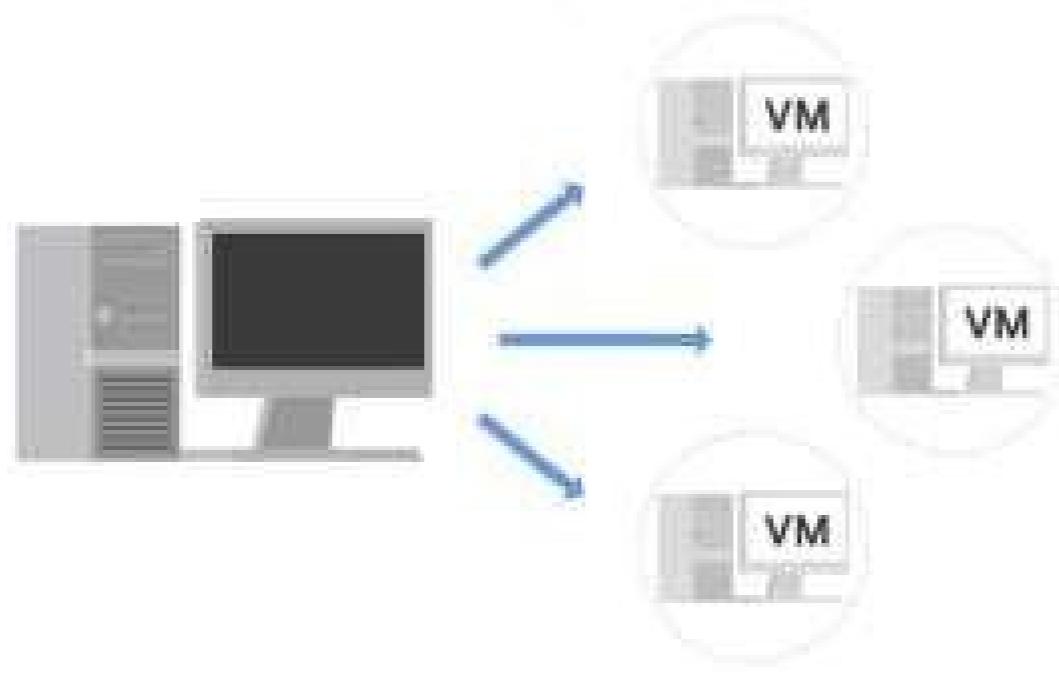
The minimum duration of SEE is 02 hours

Suggested Learning Resources:

- Real digital Forensics for Handheld Devices, E.P Dorothy, Auerback Publications, 2013
- The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics, J. Sammons, Syngress Publishing, 2012
- Handbook of Digital Forensics and Investigation, E. Casey , Academic Press, 2010
- Malware Forensics Field Guide for Windows Systems: Digital Forensics Field Guides, C.H Malin, E. Casey and J M Aquilina, Syngress, 2012
- The Best Damn Cybercrime and digital forensics Book Period, J Wiles and A Reyes, Syngress, 2007

What is a virtual machine?

A **virtual machine (VM)** is a virtual version of a physical computer. Virtual machines are one example of virtualization. Virtualization is the process of using software to create virtual representations of various physical machines. The term “virtual” refers to machines that don’t exist physically, but operate like they do because their software simulates physical hardware. Virtual systems don’t use dedicated physical hardware. Instead, they use software-defined versions of the physical hardware. This means that a single virtual machine has a virtual CPU, virtual storage, and other virtual hardware. Virtual systems are just code.



You can run multiple virtual machines using the physical hardware of a single computer. This involves dividing the resources of the host computer to be shared across all physical and virtual components. For example, **Random Access Memory (RAM)** is a hardware component used for short-term memory. If a computer has 16GB of RAM, it can host three virtual machines so that the physical computer and virtual machines each have 4GB of RAM. Also, each of these virtual machines would have their own operating system and function similarly to a typical computer.

Benefits of virtual machines

Security professionals commonly use virtualization and virtual machines. Virtualization can increase security for many tasks and can also increase efficiency.

Security

One benefit is that virtualization can provide an isolated environment, or a sandbox, on the physical host machine. When a computer has multiple virtual machines, these virtual machines are “guests” of the computer. Specifically, they are isolated from the host computer and other guest virtual machines. This provides a layer of security, because virtual machines can be kept separate from the other systems. For example, if an individual virtual machine becomes infected with malware, it can be dealt with more securely because it’s isolated from the other machines. A security professional could also intentionally place malware on a virtual machine to examine it in a more secure environment. **Efficiency**

Using virtual machines can also be an efficient and convenient way to perform security tasks. You can open multiple virtual machines at once and switch easily between them. This allows you to streamline security tasks, such as testing and exploring various applications.

Experiment No 1. Install Kali Linux and explore basic Linux commands and tools.

Kali Linux Tutorial



What is Kali Linux?

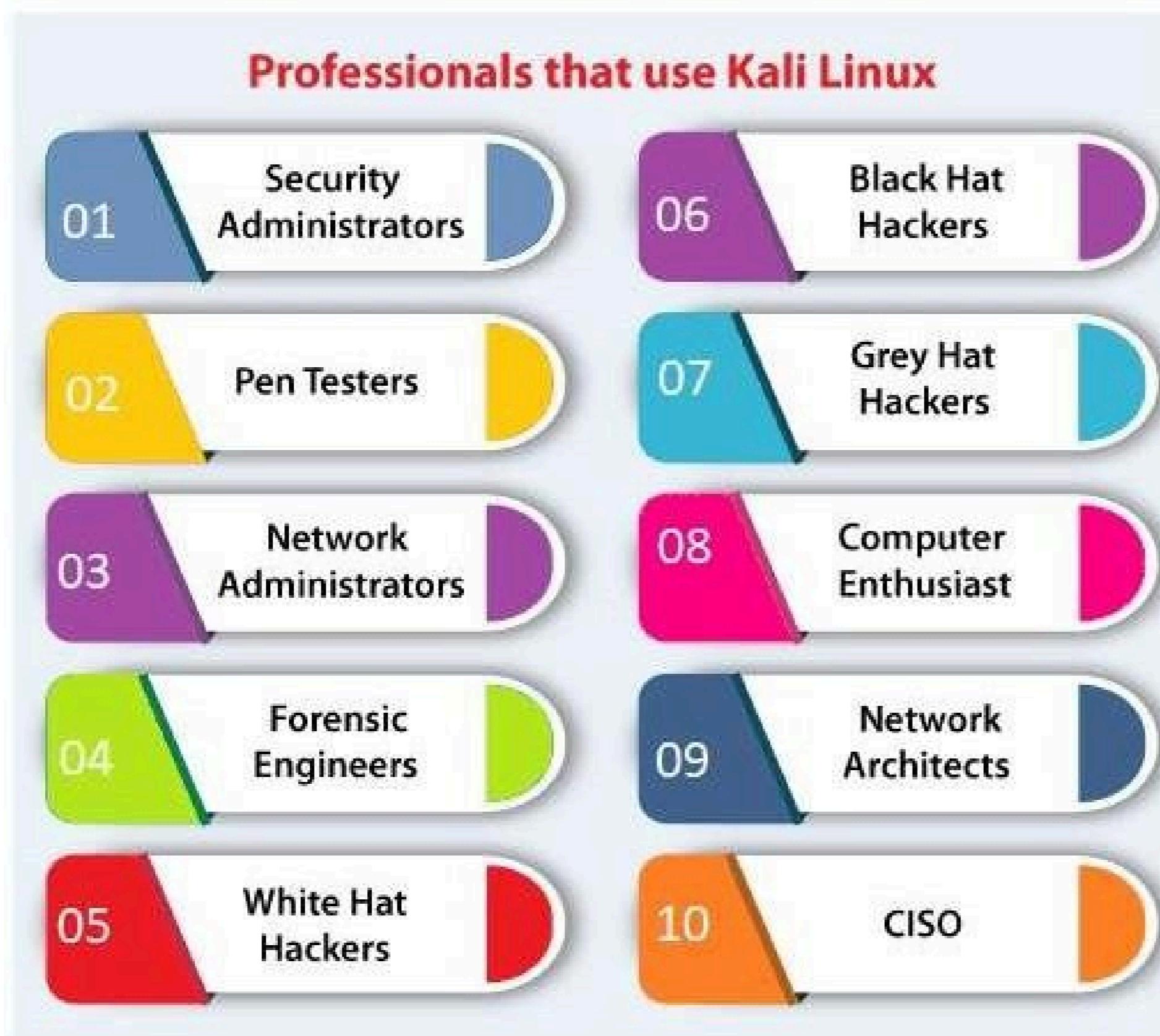
Kali Linux is a **Debian-based Linux distribution** that is designed for **digital forensics** and **penetration testing**. It is funded and maintained by **Offensive Security**, an information training company. Kali Linux was developed through the rewrite of **BackTrack** by **Mati Aharoni** and **Devon Kearns** of **Offensive Security**. Kali Linux comes with a large number of tools that are well suited to a variety of information security tasks, including **penetration testing**, **computer forensics**, **security research**, and **reverse engineering**.

Approximately, Kali Linux has 600 penetration testing programs, such as OWASP ZAP web application security scanners and Burp Suite, Airxrack-ng (software suite for wireless penetration-testing LANs), sqlmap (database takeover tool and automatic SQL injection), John the Ripper (password cracker), Metasploit (framework for penetration testing), Wireshark (packet analyzer), Nmap (port scanner), Armitage (a tool for graphical cyber-attack management), etc.

Who Uses Kali Linux and Why?

Kali Linux is a one-of-a-kind operating system since it is one of the few platforms that are freely utilized by both good and bad guys. This operating system is widely used by both **Security Administrators** and **Black Hat Hackers**. One is responsible for detecting and preventing security breaches, while the other is responsible for identifying and perhaps exploiting security breaches. The number of tools configured and preinstalled on the operating system makes Kali Linux a Swiss Army Knife in any security professional's toolbox.

Professionals that Use Kali Linux



1. Security Administrators

Security Administrators are responsible for protecting their institution's information and data. They use Kali Linux to review their environments(s) and ensure there are no easily discoverable vulnerabilities.

2. Pen Testers

Pen Testers use Kali Linux to audit environments and perform reconnaissance on corporate environments they've been recruited to examine.

3. Network Administrators

Network Administrators are responsible for keeping the network running smoothly and securely. They audit their network with Kali Linux. **For example**, Kali Linux has the capacity to detect illegitimate access points.

4. Forensic Engineers

Kali Linux has a '**Forensic Mode**', which permits a forensic engineer to perform data search and recovery in some cases.

5. White Hat Hackers

White Hat Hackers, like **Pen Testers**, utilize Kali Linux to audit and uncover potential vulnerabilities in an environment.

6. Black Hat Hackers

Black Hat Hackers use Kali Linux in order to find and exploit vulnerabilities. It contains a number of social engineer applications that a Black Hat Hacker can use to compromise an organization or individual.

7. Grey Hat Hackers

Grey Hat Hackers are in the middle of the spectrum between **White Hat** and **Black Hat Hackers**. They will use Kali Linux in the same as the two listed above.

8. Computer Enthusiast

Computer Enthusiast is a very general term, but anybody interested in learning more about networking or computers can use Kali Linux to better understand **IT, networking, and common vulnerabilities**.

9. Network Architects

Network architects are responsible for designing secure network environments. They use Kali Linux to check their initial designs and make sure nothing was missed or configured incorrectly.

10. CISO

CISO (Chief Information Security Officers) utilizes Kali Linux to audit their environment internally and find out if any new applications or rogue configurations have been installed.

Running Kali Linux Pre-Built VM on VirtualBox

A quick way to run a Kali Linux VM is by using a pre-built VirtualBox image. The section below explains how to obtain and start a pre-built Kali Linux image on VirtualBox.

1. Visit the [Pre-built VMs](#) page on the official Kali Linux website.

2. Select the desired architecture and click the download button in the bottom left corner of the **VirtualBox** card.

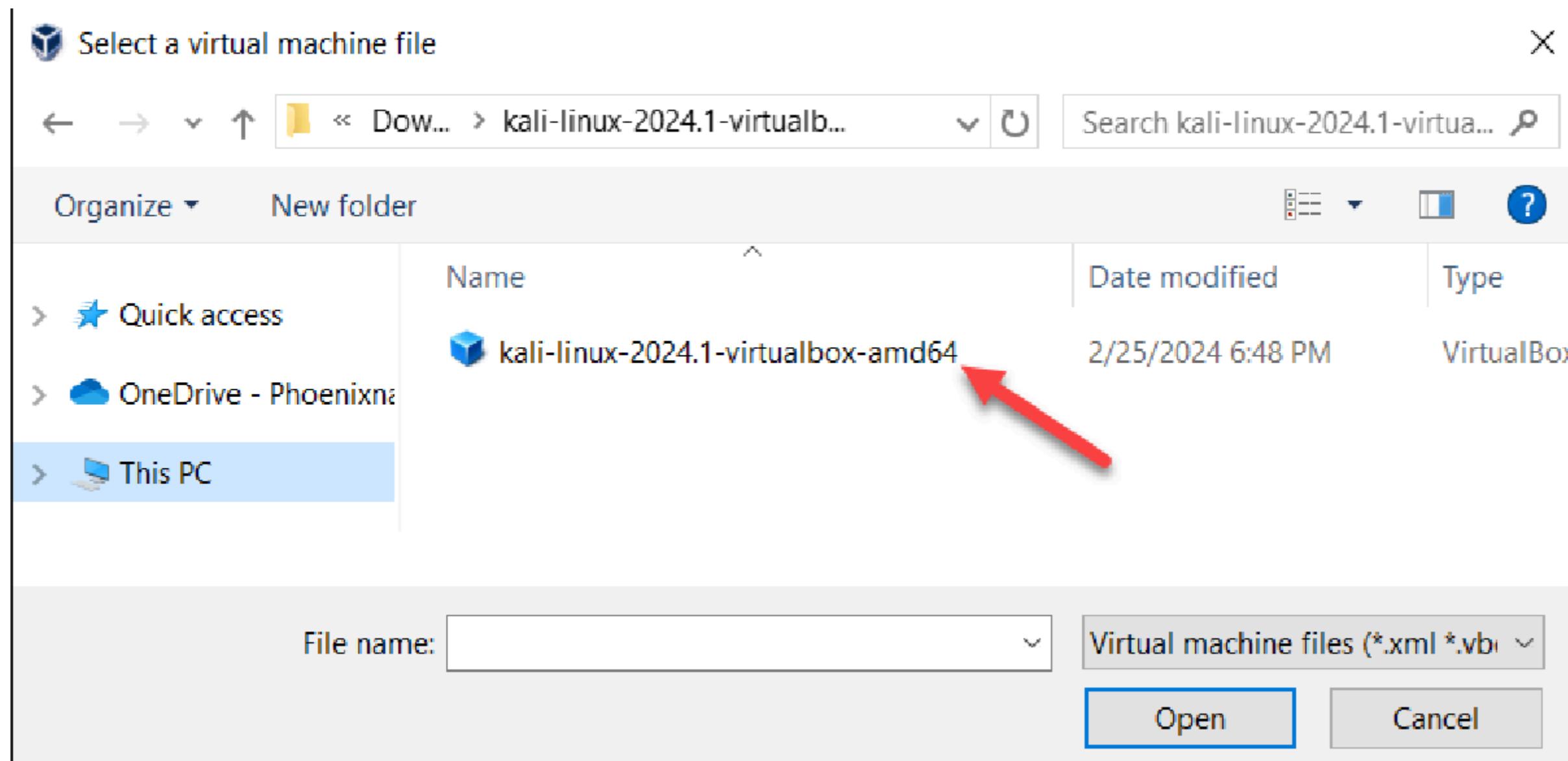


3. Wait for the **7z** file to download, then unpack it to a directory of your choice.

4. Open **VirtualBox Manager** and select the **Add** button in the top menu.

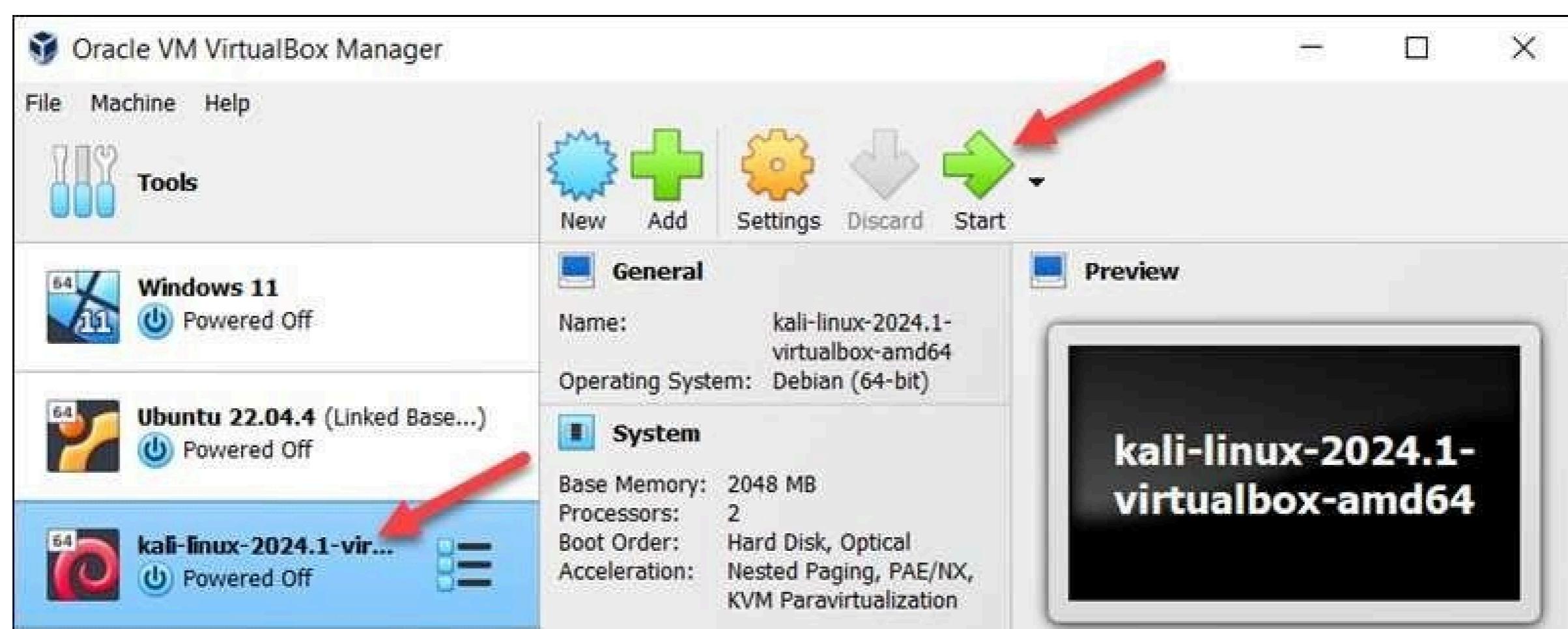


5. Locate the virtual machine file you downloaded and unpacked. Double-click the file to open it.



A Kali *Linux VM instance appears in the menu on the left side of the screen.

6. Select the instance and click the **Start** button in the top menu.



Wait for the system to boot up.

7. On the login screen, use username **kali** and password **kali** to log in.

Program 1. Install KALI LINUX and explore Linux commands and Tools.

A. ->Steps to install kali Linux

1. Open Kali Linux website in any browser Download to kali Linux file from official website for Virtual box (2.9gb)

URL: www.Kali.org

(Note: download 64 bit)

2. Open new tab and search for WIN RAR application to execute the above downloaded file. Search for win Rar download and install the latest win Rar version.
3. Now search and download the “Virtual Box 6.1” and install for windows hosts.
4. Now open files and create a folder named ‘KALI’ in any location of your choice.
5. Now go to Download Section and Open the Kali Linux file that was downloaded earlier with the help of WIN RAR application.
6. Now the win rar window will be opened, now click on “Extract To” and select the location of kali folder which was created earlier and extract all files her.
7. Now open the installed “Virtual Box” and click on “add” option.
8. Now select the “Kali” folder and “Kali file” inside it and open it.
9. Now click on settings, go to network option, click on “nat” command and convert it to Bridge Adaptor.

10. Then click on star and login to kali using username and password as “kali” .

->Basic Commands

1. date

output: Tue May 15 02:07:58 AM 2024

```
(kali㉿kali)-[~]
$ date
Fri 08 Oct 2021 08:41:25 AM EDT
```

- 2.Kali

output: /Kali / kali

3. wh

oami

output:

Kali

```
(kali㉿kali)-[~]
$ whoami
kali

(kali㉿kali)-[~]
$ who
kali      tty7          2021-10-08 08:39 (:0)
```

CYBER SECURITY LAB

BICL404

output: /home/ Kali

```
(kali㉿kali)-[~]
└─$ pwd
/home/kali
(javatpoint) [~]
└─$ cd Desktop
(kali㉿kali)-[~/Desktop]
└─$ pwd
/home/kali/Desktop
(kali㉿kali)-[~/Desktop]
└─$ █
```

5. Unam

e Output:

Linux

```
(kali㉿kali)-[~]
└─$ uname
Linux
(kali㉿kali)-[~]
└─$ uname -a
Linux kali 5.10.0-kali7-686-pae #1 SMP Debian 5.10.28-1kali1 (2021-04-12) i686 GNU/Linux
(kali㉿kali)-[~]
└─$ users
kali
```

6. Hist

ory: Output:

date Kali

Whoami

Pwd

Uname

```
(kali㉿kali)-[~]
└─$ history
 1
 2 airmon-ng
 3 air
 4 airmon-ng start [root]
 5 sudo airmon-ng
 6 sudo ip link set IFACE down
 7 ifconfig
 8 sudo apt-get install kali-linux-wireless
 9 iwconfig
10 air
11 ifconfig
12 sudo iw dev
13 lsb_release -a
14 clear
15 cat /etc/os-release
16 clear
17 hostnamectl
18 clear
19 hostnamectl
20 hostnamectl
21 clear
22 hostnamectl
23 iwconfig
24 sudo iw dev
25 sudo update
26 timedatectl
27 timedatectl list-timezones
28 timedatectl
```

7. user

s Output:

Kali

```
Home
└──(kali㉿kali)-[~]
    └──$ users
        kali
```

8. ls

Output: Desktop Documents Music Pictures

```
Home
└──(kali㉿kali)-[~]
    └──$ ls -al
        total 148
        drwxr-xr-x 15 kali kali 4096 Oct  8 08:43 .
        drwxr-xr-x  3 root root 4096 May 30 18:01 ..
        -rw-r--r--  1 kali kali   1 Jun  1 01:59 .bash_history
        -rw-r--r--  1 kali kali 220 May 30 18:01 .bash_logout
        -rw-r--r--  1 kali kali 5349 May 30 18:01 .bashrc
        -rw-r--r--  1 kali kali 3526 May 30 18:01 .bashrc.original
        drwxr-xr-x 11 kali kali 4096 Oct  8 08:40 .cache
        drwx----- 11 kali kali 4096 Sep 17 12:51 .config
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Desktop
        -rw-r--r--  1 kali kali   5 May 31 17:33 .dmrc
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Documents
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Downloads
        -rw-r--r--  1 kali kali 11759 May 30 18:01 .face
        lrwxrwxrwx  1 kali kali   5 May 30 18:01 .face.icon → .face
        drwx-----  3 kali kali 4096 May 31 03:35 .gnupg
        -rw-----  1 kali kali   0 May 31 03:35 .ICEauthority
        drwxr-xr-x  3 kali kali 4096 May 31 03:35 .local
        drwx-----  5 kali kali 4096 Aug  8 06:02 .mozilla
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Music
        drwxr-xr-x  2 kali kali 4096 Oct  8 08:41 Pictures
        -rw-r--r--  1 kali kali   807 May 30 18:01 .profile
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Public
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Templates
        -rw-----  1 kali kali   4 Oct  8 08:39 .vboxclient-draganddrop.pid
        -rw-----  1 kali kali   4 Oct  8 08:39 .vboxclient-seamless.pid
        drwxr-xr-x  2 kali kali 4096 May 31 03:35 Videos
        -rw-----  1 kali kali   49 Oct  8 08:39 .Xauthority
        -rw-----  1 kali kali 6947 Oct  8 08:43 .xsession-errors
```

9. U

ptime

Output:

```
Home
└──(kali㉿kali)-[~]
    └──$ uptime
        09:34:53 up 57 min,  1 user,  load average: 0.29, 0.18, 0.16
```

10. C

al

Output

:

```
Home
└──(kali㉿kali)-[~]
    └──$ cal
        October 2021
        Su Mo Tu We Th Fr Sa
                           1  2
        3  4  5  6  7  8  9
        10 11 12 13 14 15 16
        17 18 19 20 21 22 23
        24 25 26 27 28 29 30
        31
```

Program 2: Perform basic network scanning using the nmap tool (Zenmap on windows). Identify services, open ports, active hosts, operating systems, and vulnerabilities.

Nmap is defined as a tool that can detect or diagnose services that are running on an **Internet-connected system** by a network administrator in their networked system used to identify potential security flaws. It is used to automate redundant tasks, such as monitoring the service. Nmap is convenient during penetration testing of networked systems. Nmap provides the network details, and also helps to determine the security flaws present in the system. Nmap is **platform-independent** and runs on popular **operating systems** such as **Linux, Windows and Mac**.

Nmap is a useful tool for network scanning and auditing purposes.

- o It can search for hosts connected to the Network.
- o It can search for free ports on the target host.
- o It detects all services running on the host with the help of **operating system**.
- o It also detects any **flaws or potential vulnerabilities** in networked systems.

- Open any browser and search “nmap download” and open the first website([URL: " nmap.org"](http://nmap.org)).
- Click on windows option and download latest stable release version of nmap for windows “nmap-7.95-setup.exe” .
- Its about 32MB file, after downloading it install in your system

Identify Ports

Port Scanning Basics: The six port states recognized by Nmap

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.

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.

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..

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.

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.

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.

-p <port number>

-p <port ranges> nmap -p80 127.0.0.1 nmap -p23,25,80,8080

nmap -p U:53,111,137,T:21-25,80,139,8080 127.0.0.1

To perform basic operation

- Open cmd(command prompt) and type “ipconfig” to get the IP addresses of your system.
- Now copy any IP address from there and paste that IP address in the “target section” inside the nmap program or application.
- And click on “scan” button, eg: address 192.168.56.1

Output:

Starting nmap 7.95(<https://nmap.org>) at 2024-05-21 12:12 India standard

time NSE: loaded 157 scripts for scanning

NSE: script pre-scanning

NSE: script scanning 192.168.27.29

Not shown:997 closed tcp

ports(reset)

PORT	STATE	SERVICE	VERSION
135/tcp	open	msrpc	microsoft windows
pc 139/tcp	open	netbios-ssn	microsoft windows
			netbios-ssn

445/tcp open

 microsoft-ds? Device

type: general purpose Running:

 microsoft windows 10/11

Oscpe:

cpc:/0:microsoft:windows_10cpc:/0:microsoft:

 windows_11

 os details: microsoft windows 10 1607_11 23hz

 uptime guess:0.045 days(since tue may 21 11:06:53

 2024) network distance:0 hops

 tcp sequence prediction:

 difficulty=254 Ip Id sequence

 generation: incremental

 service info:05:windows:cpe:cpc:/0:microsoft:windows

Host script results:

1smbz-security-mode:

1 3:1:1:

1_message signing enabled but not

required 1 date:2024-05-21 706:41:58

1_start_date:n/a

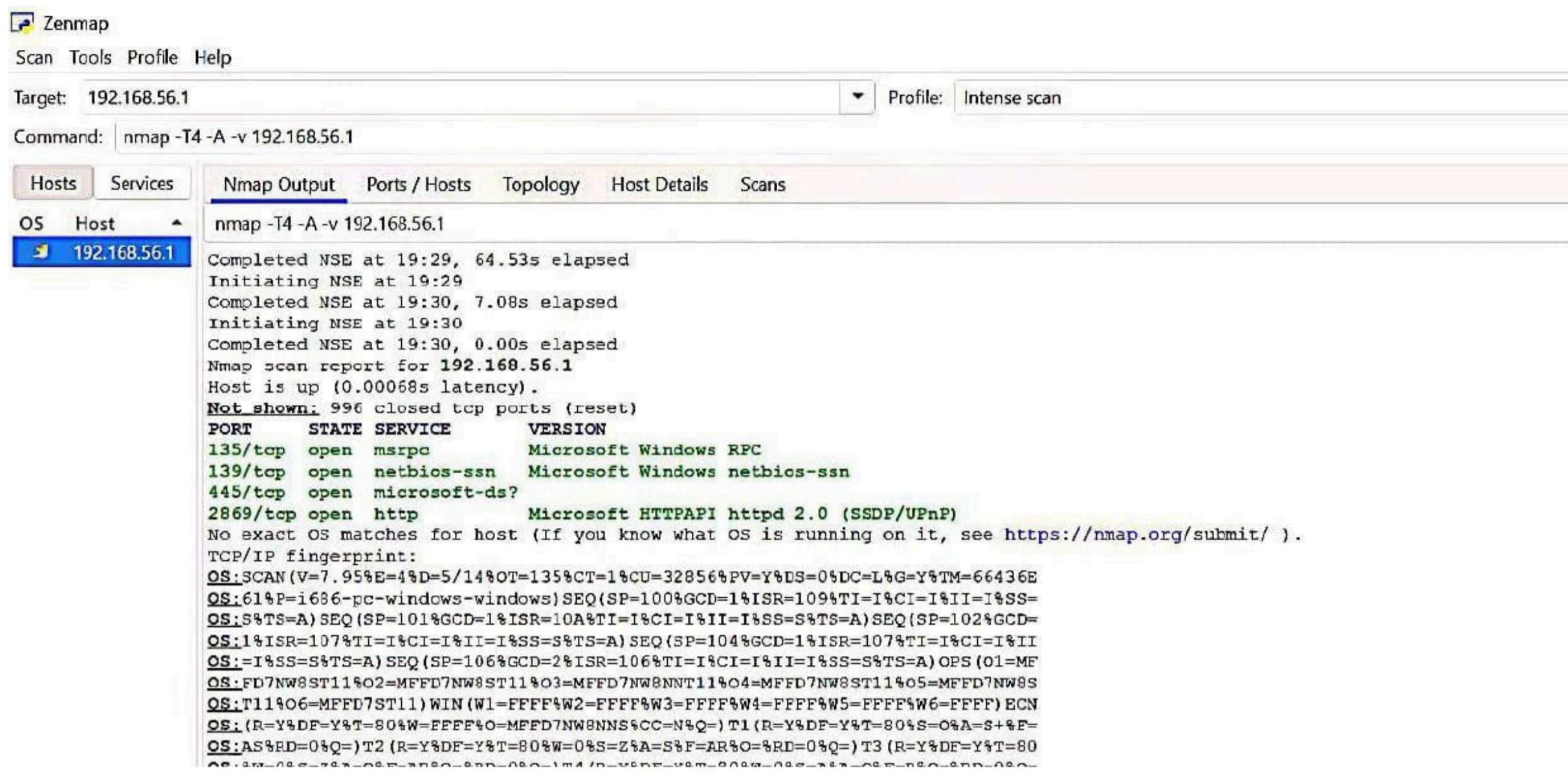
NSE: script port_scanning

Initiating NSE at 12:12

Completed NSE at 12:12,0.00s

elapsed Initiating NSE at 12:12

Completed NSE at 12:12,0.00s elapsed



```
Completed NSE at 19:29, 64.53s elapsed
Initiating NSE at 19:29
Completed NSE at 19:30, 7.08s elapsed
Initiating NSE at 19:30
Completed NSE at 19:30, 0.00s elapsed
Nmap scan report for 192.168.56.1
Host is up (0.00068s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
135/tcp    open  msrpc        Microsoft Windows RPC
139/tcp    open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds?
2869/tcp   open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).

TCP/IP fingerprint:
OS:SCAN(V=7.95%E=4%D=5/14%OT=135%CT=1%CU=32856%PV=Y%DS=0%DC=L%G=Y%TM=66436E
OS:61%P=i686-pc-windows-windows)SEQ(SP=100%GCD=1%ISR=109%TI=I%CI=I%II=I%SS=
OS:S%TS=A)SEQ(SP=101%GCD=1%ISR=10A%TI=I%CI=I%II=I%SS=S%TS=A)SEQ(SP=102%GCD=
OS:1%ISR=107%TI=I%CI=I%II=I%SS=S%TS=A)SEQ(SP=104%GCD=1%ISR=107%TI=I%CI=I%II
OS:=I%SS=S%TS=A)SEQ(SP=106%GCD=2%ISR=I%CI=I%II=I%SS=S%TS=A)OPS(O1=MF
OS:FD7NW8ST11%O2=MFFD7NW8ST11%O3=MFFD7NW8NNT11%O4=MFFD7NW8ST11%O5=MFFD7NW8S
OS:T11%O6=MFFD7ST11)WIN(W1=FFFF%W2=FFFF%W3=FFFF%W4=FFFF%W5=FFFF%W6=FFFF)ECN
OS:(R=Y%DF=Y%T=80%W=FFFF%O=MFFD7NW8NNNS%CC=NQ=)T1(R=Y%DF=Y%T=80%S=O%A=S+%F=
OS:AS%HD=0%Q=)T2(R=Y%DF=Y%T=80%W=0%S=Z%A=S%F=AR%O=%RD=0%Q=)T3(R=Y%DF=Y%T=80
OS:AS%HD=0%Q=)T4(R=Y%DF=Y%T=80%W=0%S=Z%A=S%F=AR%O=%RD=0%Q=)
```

To find open ports and active ports

- Open nmap and give IP address in target column and give the nmap commands in command section as follows.

->Target:192.168.27.29

->Command: nmap -p80 192.168.27.29

Output:

PORT	STATE	SERVICE
80/tcp	closed	http



```
Starting Nmap 7.95 ( https://nmap.org ) at 2024-05-14 20:12 India Standard Time
Nmap scan report for 192.168.56.1
Host is up (0.00s latency).

PORT      STATE SERVICE
80/tcp    closed http

Nmap done: 1 IP address (1 host up) scanned in 0.41 seconds
```

->Target:192.168.27.29

->Command: nmap -p 1-200

192.168.27.29 Output:

Not shown:197 closed tcp
ports(reset) PORT STATE
SERVICE

PORT	STATE	SERVICE
135/tcp	open	msrpc
137/tcp	filtered	netbios-ns
139/tcp	open	netbios-ssn

```

Starting Nmap 7.95 ( https://nmap.org ) at 2024-05-14 20:12 India Standard Time
Nmap scan report for 192.168.56.1
Host is up (0.00071s latency).
Not shown: 197 closed tcp ports (reset)
PORT      STATE    SERVICE
135/tcp   open     msrpc
137/tcp   filtered netbios-ns
139/tcp   open     netbios-ssn
445/tcp   open     microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 1.67 seconds
  
```

->Target:192.168.27.29

->Command: nmap -F

192.168.27.29 Output:

ports(reset)	PORT	STATE	SERVICE
135/tcp	open	msrpc	
139/tcp	open	netbios-ssn	
445/tcp	open		microsoft-ds

```

Starting Nmap 7.95 ( https://nmap.org ) at 2024-05-14 20:13 India Standard Time
Nmap scan report for 192.168.56.1
Host is up (0.00081s latency).
Not shown: 97 closed tcp ports (reset)
PORT      STATE    SERVICE
135/tcp   open     msrpc
139/tcp   open     netbios-ssn
445/tcp   open     microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 0.41 seconds
  
```

->Target:192.168.56.1

->Command: nmap -p -

192.168.56.1 Output:

ports(reset)	PORT	STATE	SERVICE
5040/tcp	open	unknown	
7680/tcp	open		pando-pub
49664/tcp	open	unknown	

```

Zenmap
Scan Tools Profile Help
Target: 192.168.56.1
Command: nmap -p - 192.168.56.1
Hosts Services
OS Host
192.168.56.1
Nmap Output Ports / Hosts Topology Host Details Scans
nmap -p - 192.168.56.1
Starting Nmap 7.95 ( https://nmap.org ) at 2024-05-14 20:13 India Standard Time
Nmap scan report for 192.168.56.1
Host is up (0.00046s latency).
Not shown: 65522 closed tcp ports (reset)
PORT      STATE    SERVICE
135/tcp    open     msrpc
137/tcp    filtered netbios-ns
139/tcp    open     netbios-ssn
445/tcp    open     microsoft-ds
2869/tcp   open     icslap
5040/tcp   open     unknown
7680/tcp   open     pando-pub
49664/tcp  open     unknown
49665/tcp  open     unknown
49666/tcp  open     unknown
49667/tcp  open     unknown
49668/tcp  open     unknown
49671/tcp  open     unknown

Nmap done: 1 IP address (1 host up) scanned in 5.83 seconds
  
```

->Target:192.168.56.1

->Command: nmap -sT

192.168.56.1 Output:

Not shown:997 filtered tcp

ports(reset) PORT STATE
SERVICE

PORT	STATE	SERVICE
135/tcp	open	msrpc
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds

```

Zenmap
Scan Tools Profile Help
Target: 192.168.56.1
Command: nmap -sT 192.168.56.1
Hosts Services
OS Host
192.168.56.1
Nmap Output Ports / Hosts Topology Host Details Scans
nmap -sT 192.168.56.1
Starting Nmap 7.95 ( https://nmap.org ) at 2024-05-14 20:14 India Standard Time
Nmap scan report for 192.168.56.1
Host is up (0.0026s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE    SERVICE
135/tcp    open     msrpc
139/tcp    open     netbios-ssn
445/tcp    open     microsoft-ds
2869/tcp   open     icslap

Nmap done: 1 IP address (1 host up) scanned in 6.14 seconds
  
```

Program 3: Phishing simulations(Google,LUCY and GoPhish).

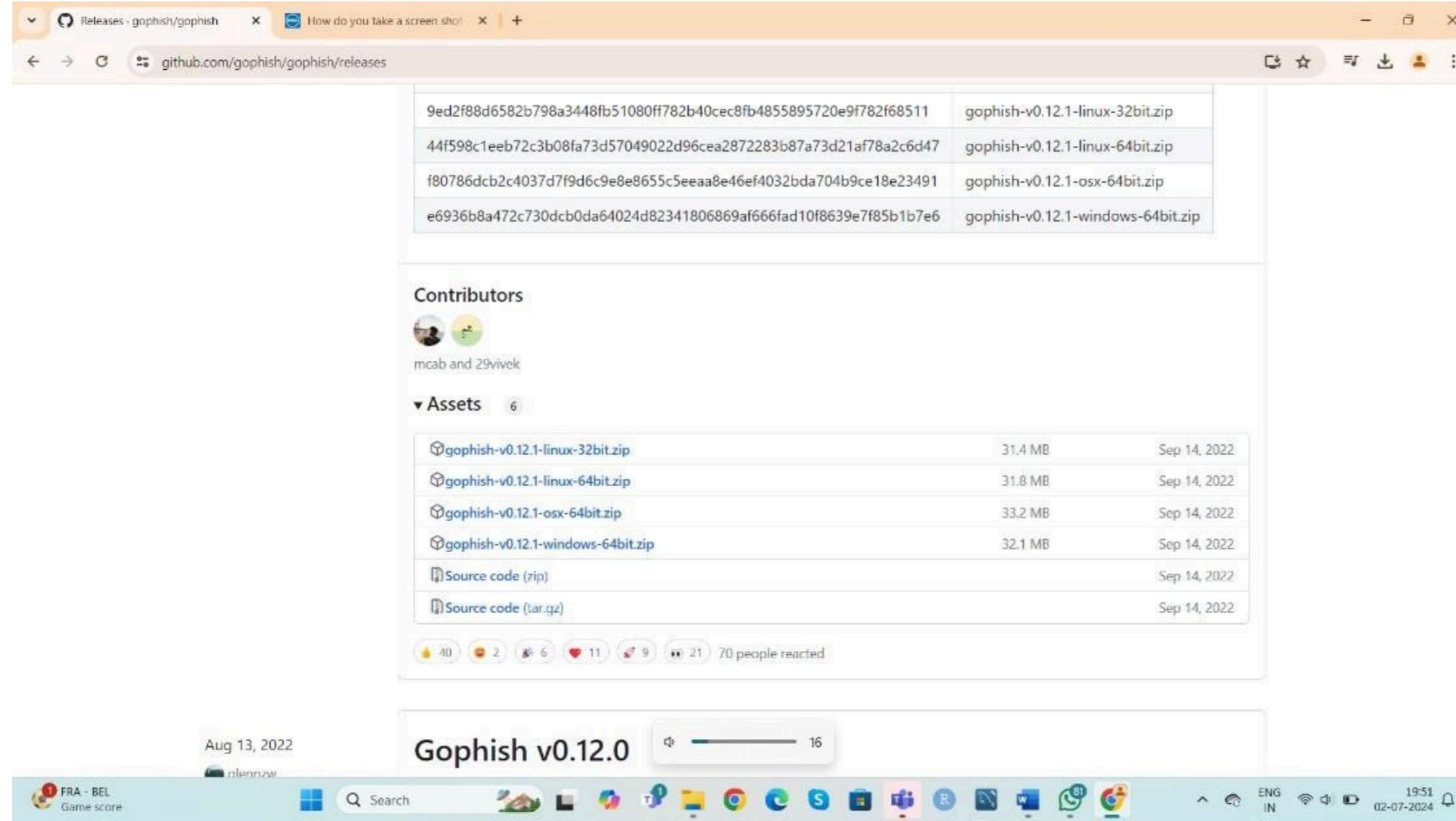
Step1:search gophish in the browser

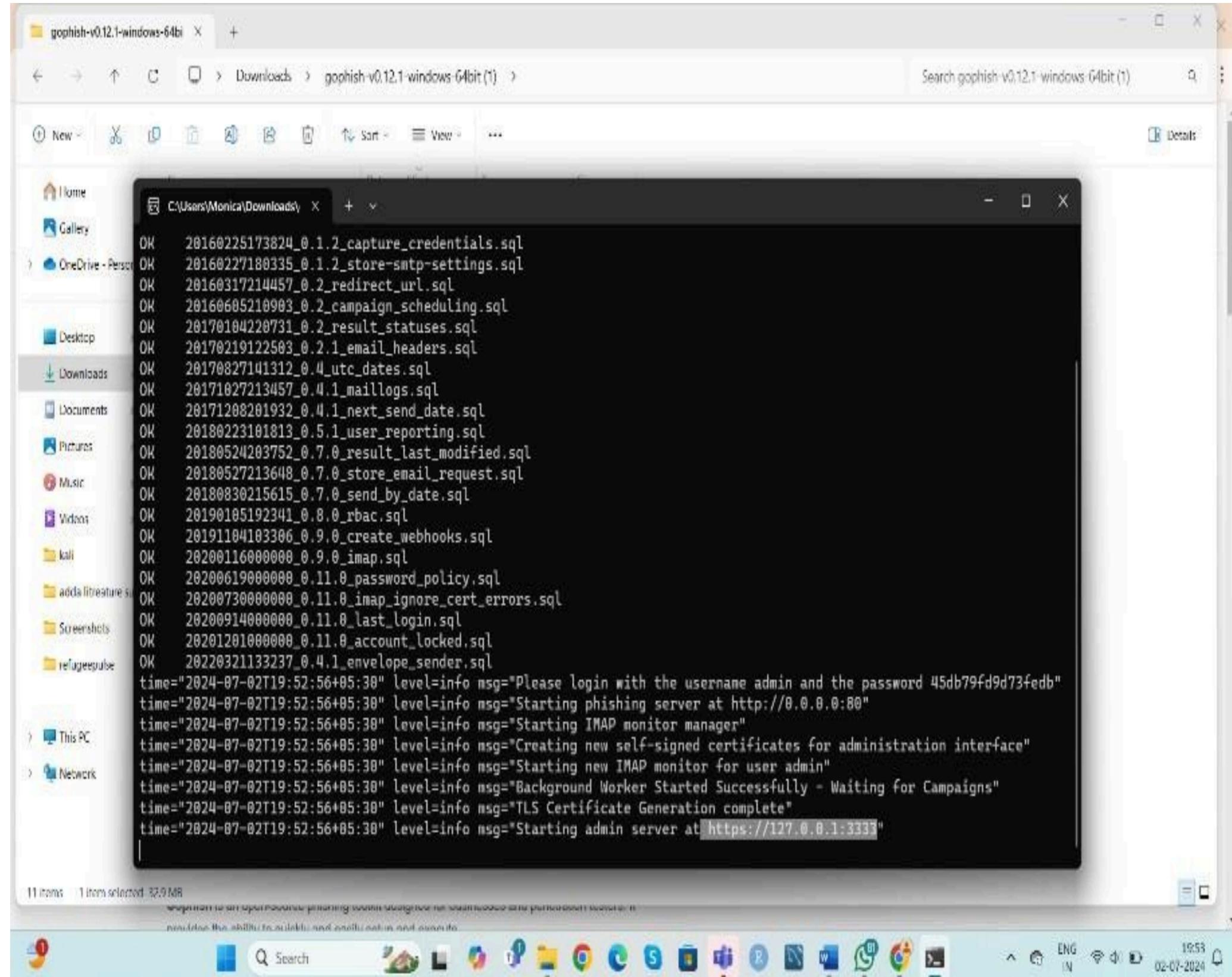
Step 2:extract gophish 12.1 windows 64bit to system Step 3:copy the http link from the prompt

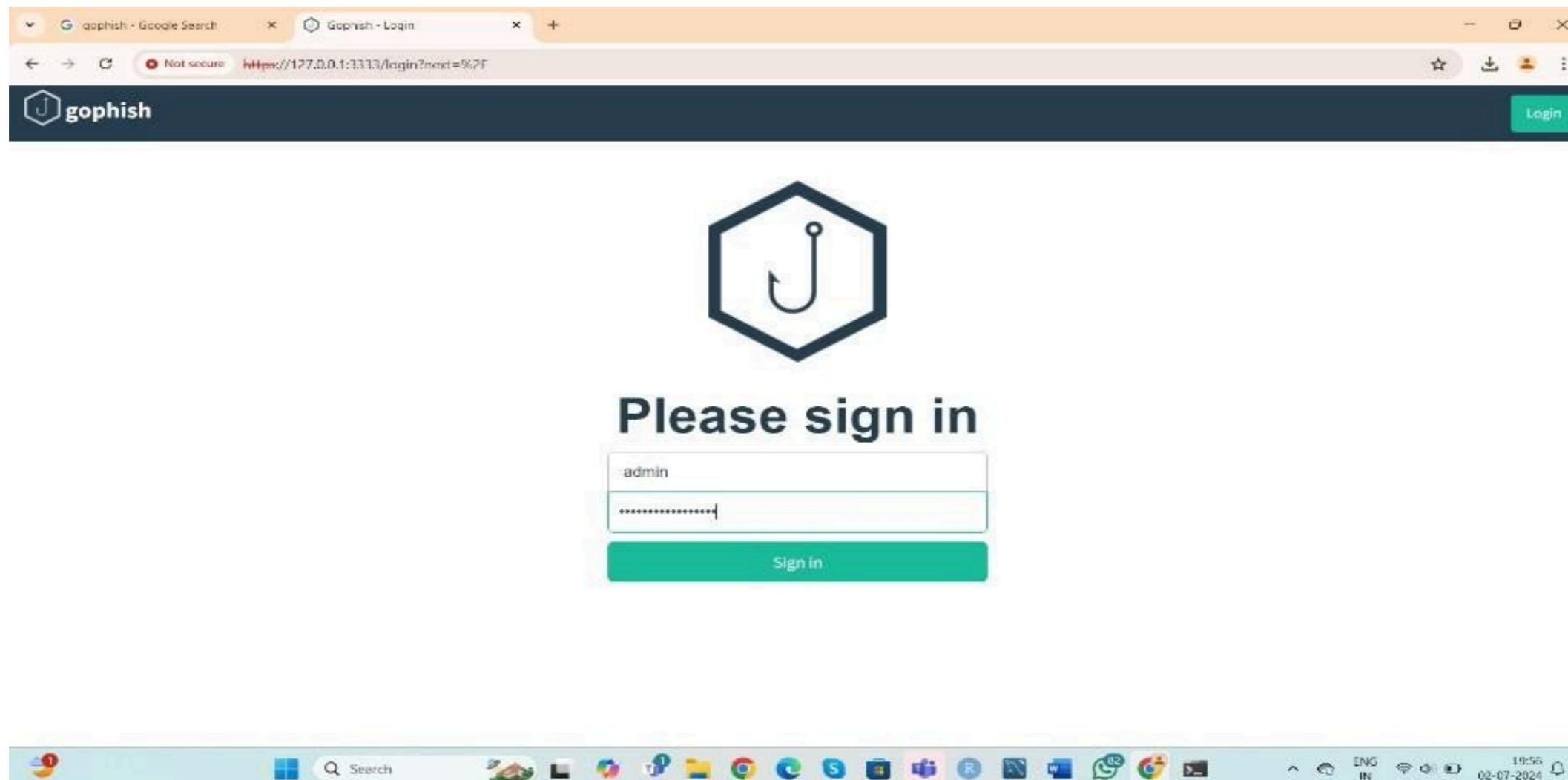
Step 4:and search in the browser

Step 5:enter username and password from prompt Step 6:reset the password

Step 7:save password and dashboard will open

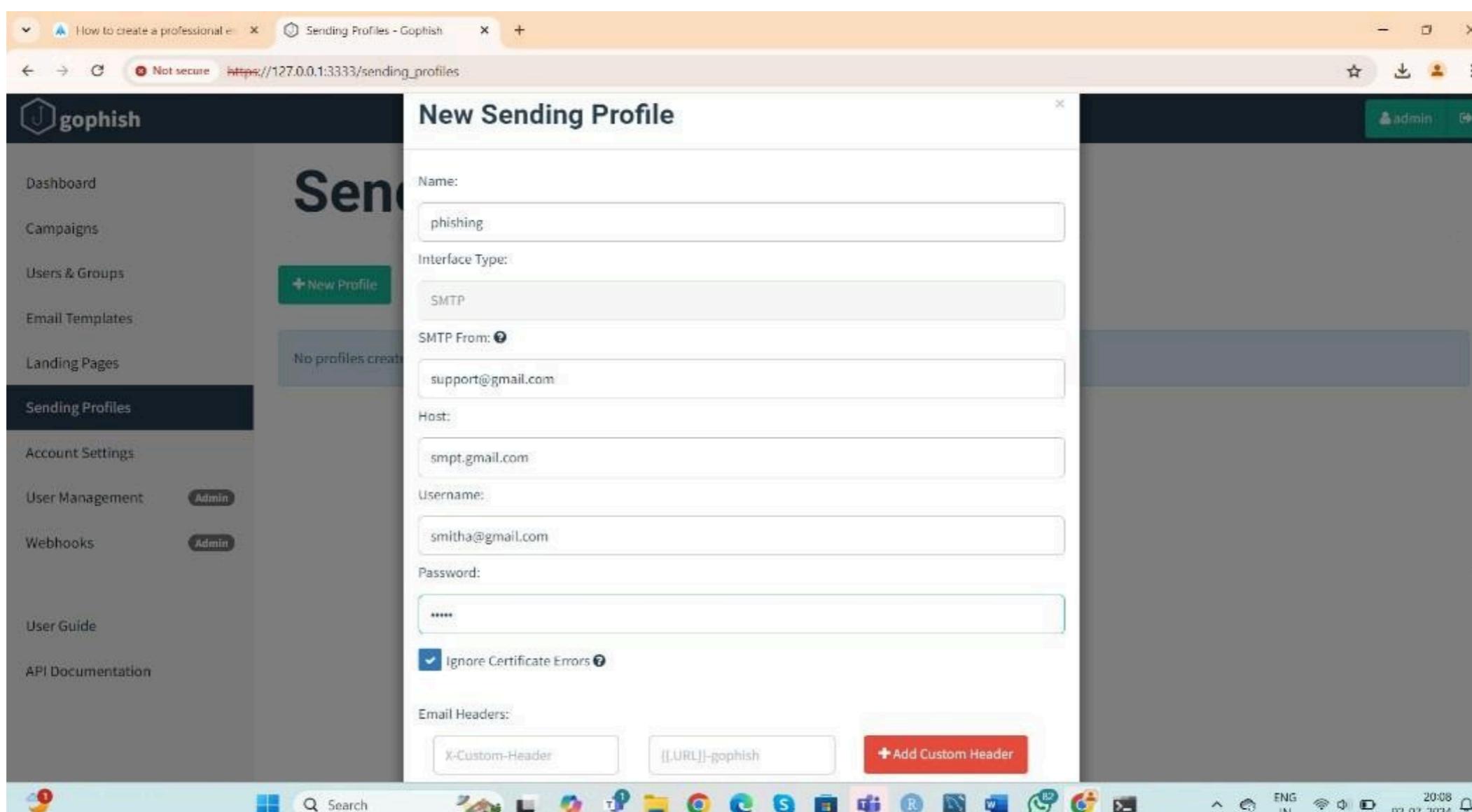






create sending profile

step 1:go to sending profile step 2:select new profile
step 3:fill the details such as name: phishing
smtp from: support@gmail.com
host:smpt.gmail.com username: own username
password: email id password step 4:save the profile

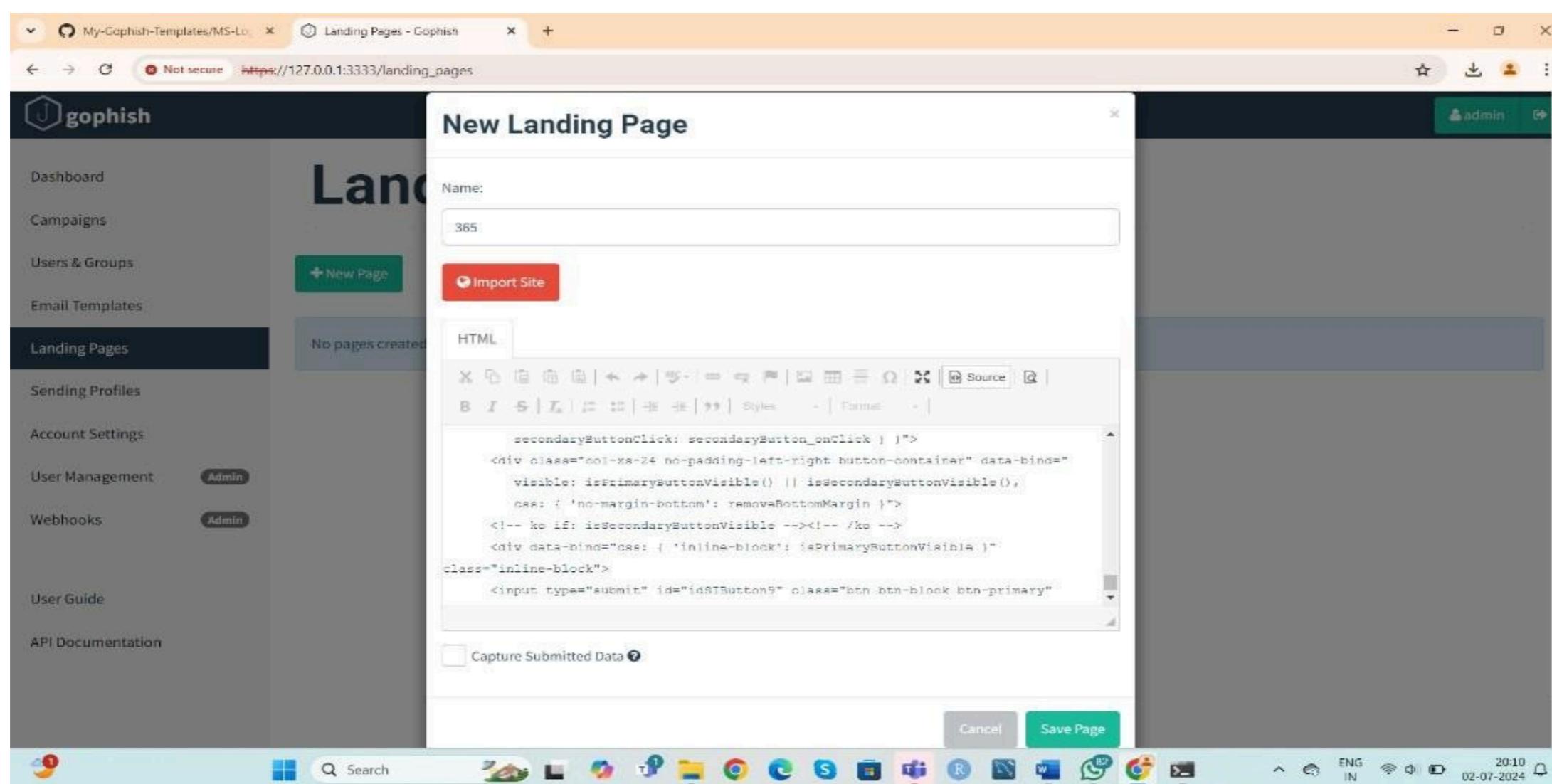


create landing page

step 1:go to landing page step 2:select new page step 3:fill the details, name:365

step 4:to fill http browse gophising 365 templates step 5:select the github link copy ms code

step 6:paste the code in the landing page html and go to search then ms login page appears step 7:go back to landing page and save the page.



The screenshot shows a GitHub repository interface. The repository name is "My-Gophish-Templates/MS-Login". The current file being viewed is "MS-Login.html". The code editor displays the HTML and CSS for a Microsoft login page template. The code includes a title, a link to a Microsoft logo icon, and a style section with font-family, text-size-adjust, and background-image properties. The GitHub interface includes a sidebar with files like "README.md" and "linkedin.html", and a top navigation bar with tabs like "Code", "Blame", and "Raw". The bottom of the screen shows a Windows taskbar with various icons and system status.

```
1<!DOCTYPE html>
2<html>
3  <head>
4    <title>Microsoft Login</title>
5    <link rel="shortcut icon" href="https://aadcdn.msftauth.net/ests/2.1/content/images/favicon_a_eupayfghqisai7k9so161g2.ico" />
6    <style>
7      font-family:sans-serif;
8      -ms-text-size-adjust:100%;
9      webkit text size adjust:100%
10     }
11     body {
12       margin:0;
13       background-image:url('https://i.stack.imgur.com/vzbuQ.jpg');
14       background-repeat: no-repeat;
15       background-attachment: fixed;
16       background-size: 100% 100%;
17     }
18     article,
19     aside,
20     details,
21     figcaption,
22     figure,
23     footer,
24     header,
25     main
```

create email template

step 1:go to new template

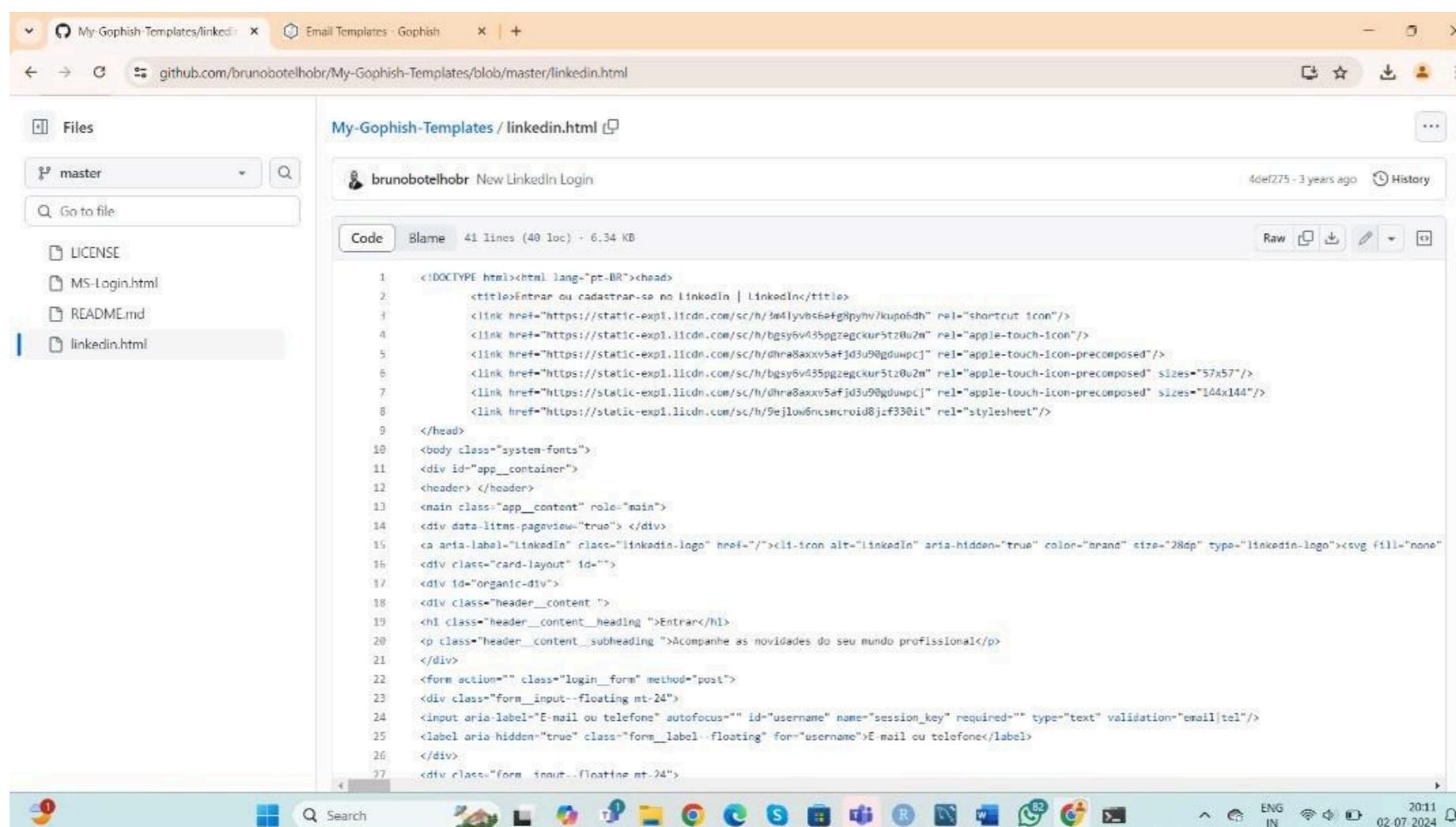
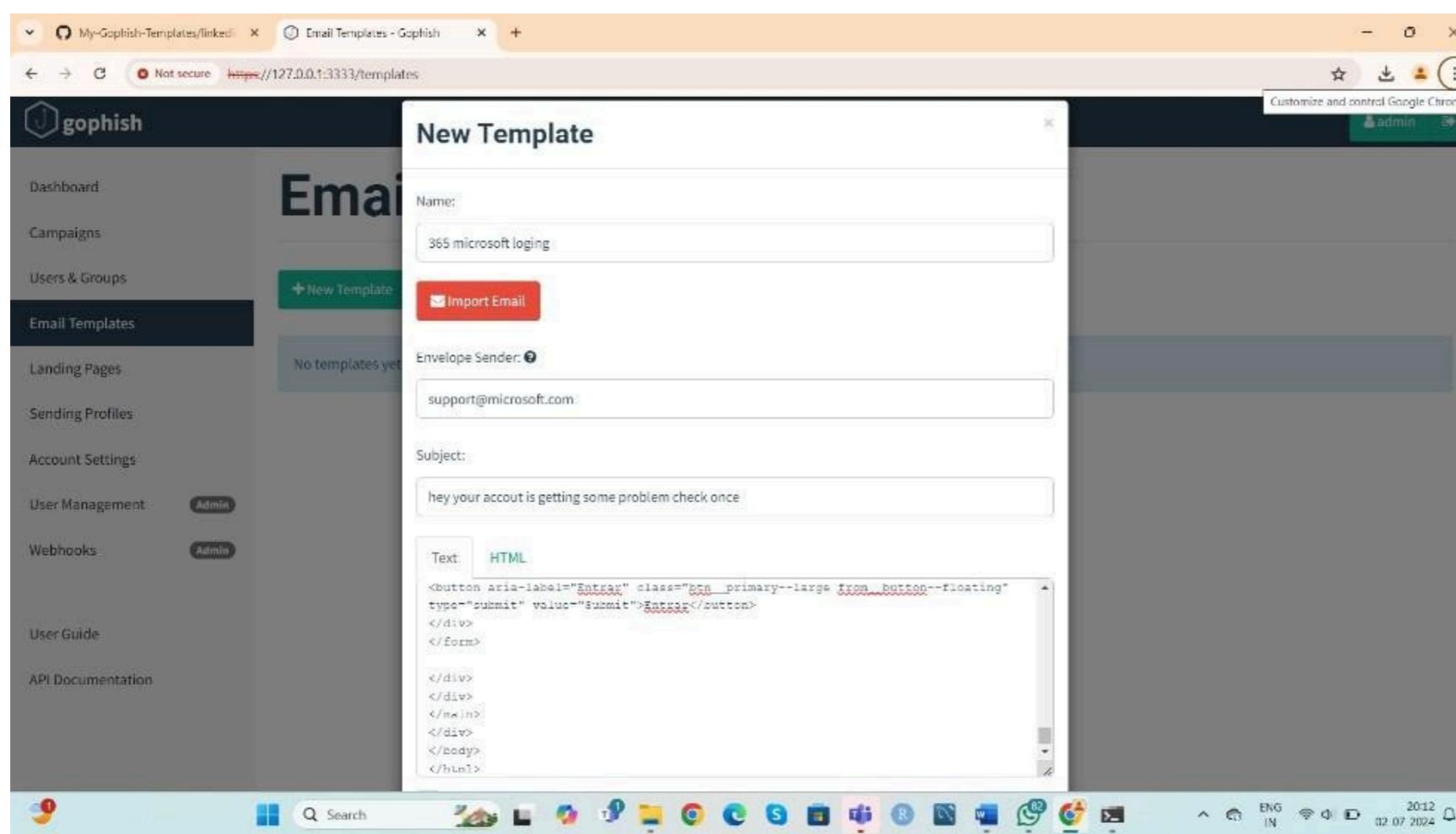
step 2:enter the necessary details such as, name:365 microsoft

loging envelope [sender:support@microsoft.com](mailto:support@microsoft.com)

subject: hey your account is getting some problem

check once text html: to fill this go to github gophising template copy the linkdin code and paste it copy the linkdin code and paste it

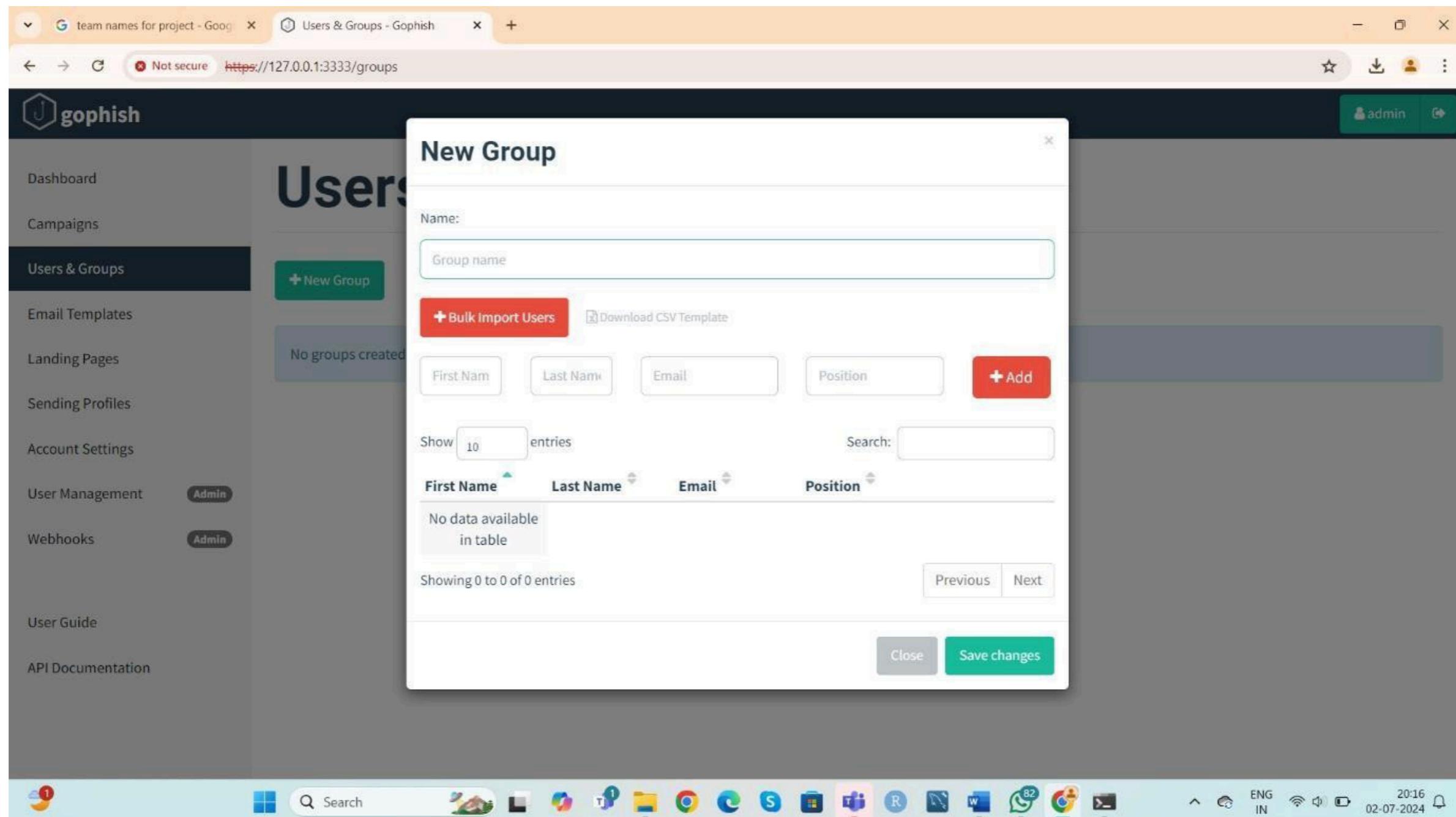
step 3:save the template



create users and groups

step 1:new group

step 2:enter necessary details such as, name(group name) step 3:add the members with name and email and position step 4:save changes

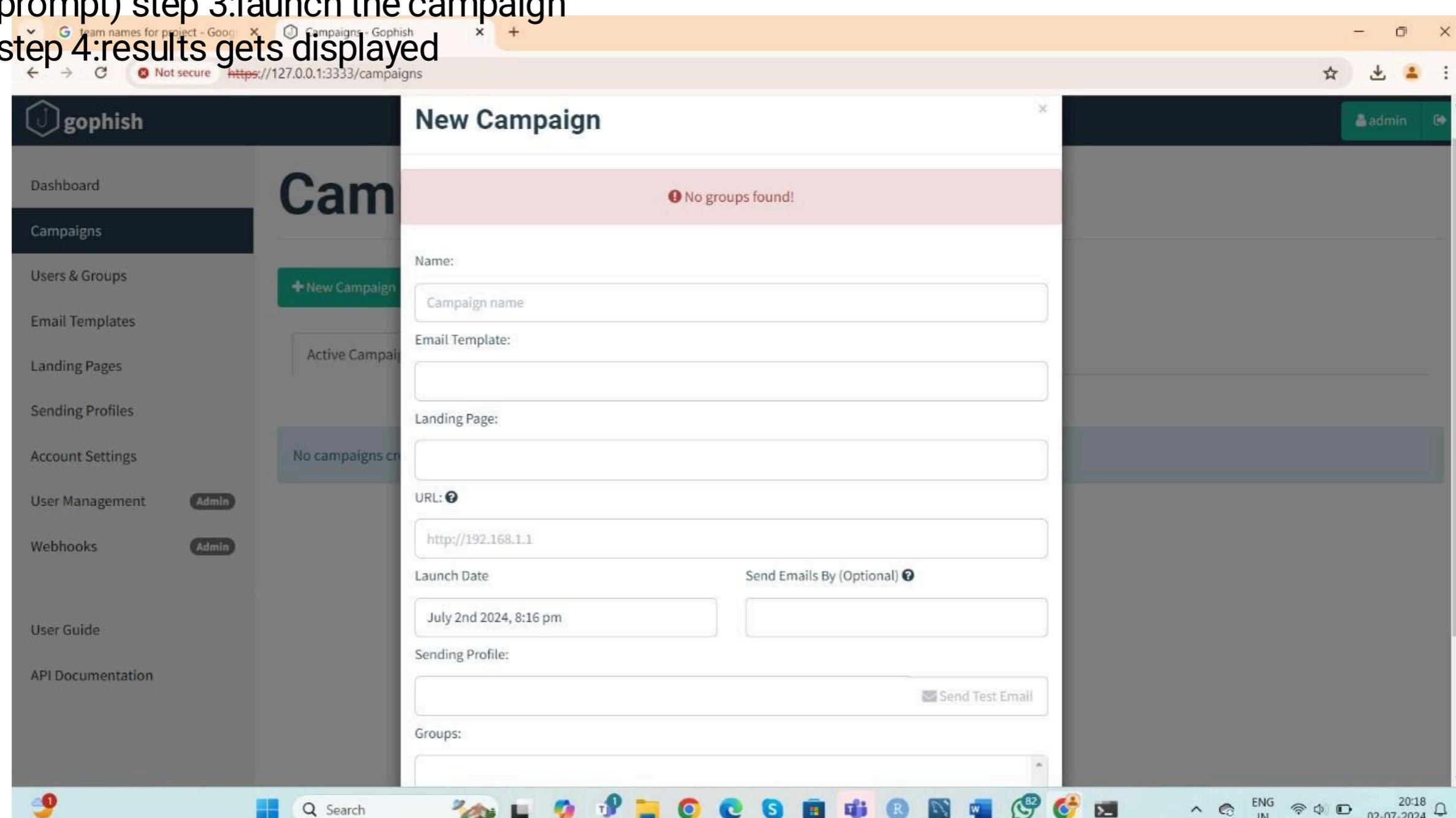


create campaigns

step 1:new campaign

step 2: enter the details eg: name: phishing and http link (from prompt) step 3:launch the campaign

step 4:results gets displayed



Program 4. Packet analysis using Wireshark

Wireshark is an open-source network protocol analysis software program, widely considered the industry standard. A global organization of network specialists and software developers supports Wireshark and continues to make updates for new network technologies and encryption methods. Government agencies, corporations, non-profits, and educational institutions use Wireshark for troubleshooting and teaching purposes. There truly isn't a better way to learn low-level networking than to look at traffic under the Wireshark microscope.

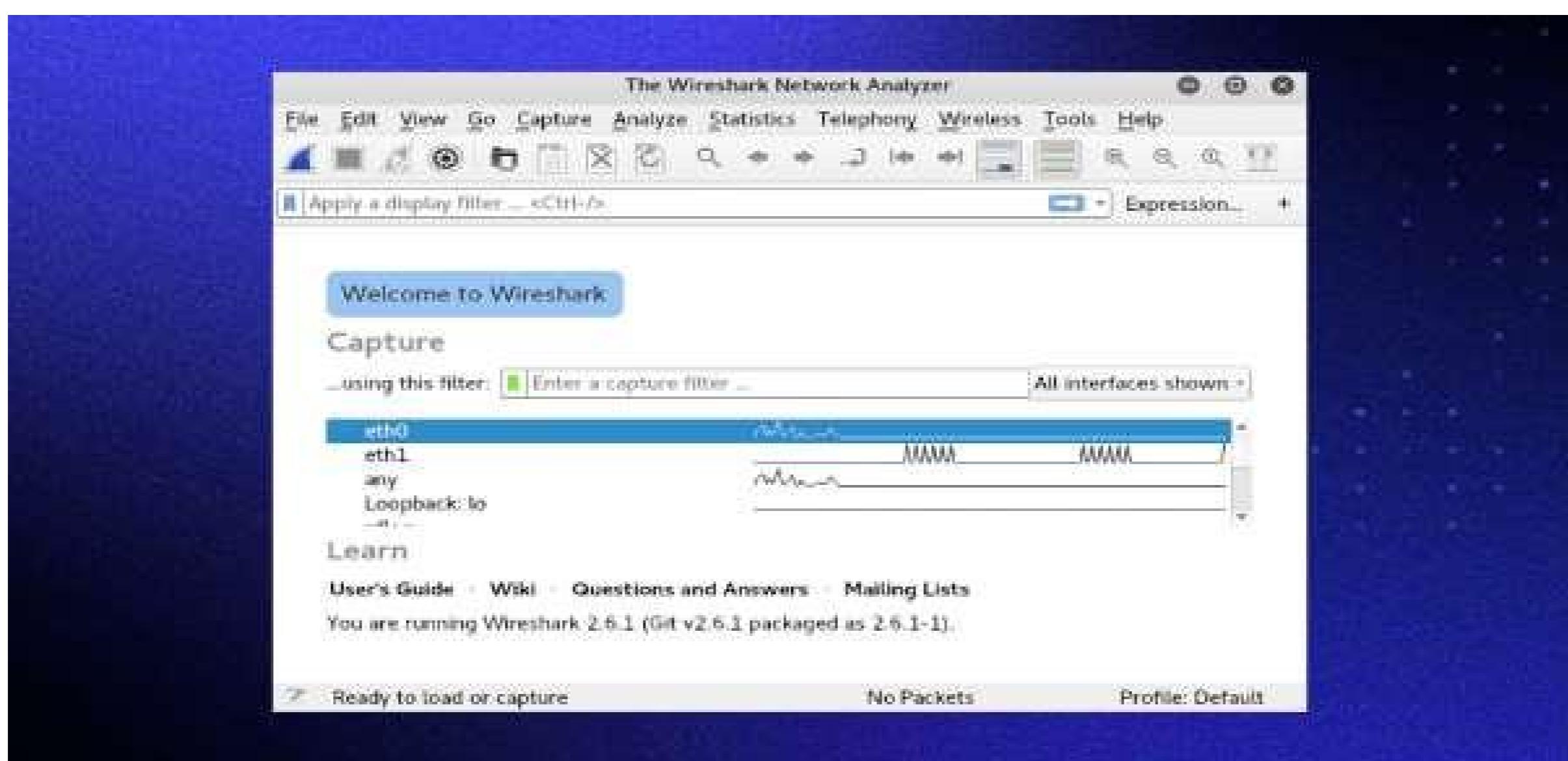
Wireshark is a packet sniffer and analysis tool. It captures network traffic from ethernet, Bluetooth, wireless (IEEE.802.11), token ring, and frame relay connections, among others, and stores that data for offline analysis. A “packet” is a single message from any network protocol

Wireshark helps:

1. Network administrators troubleshoot problems across a network
2. Security engineers examine security issues across a network
3. QA engineers verify applications
4. Developers debug protocol implementations
5. Network users learn about a specific protocol

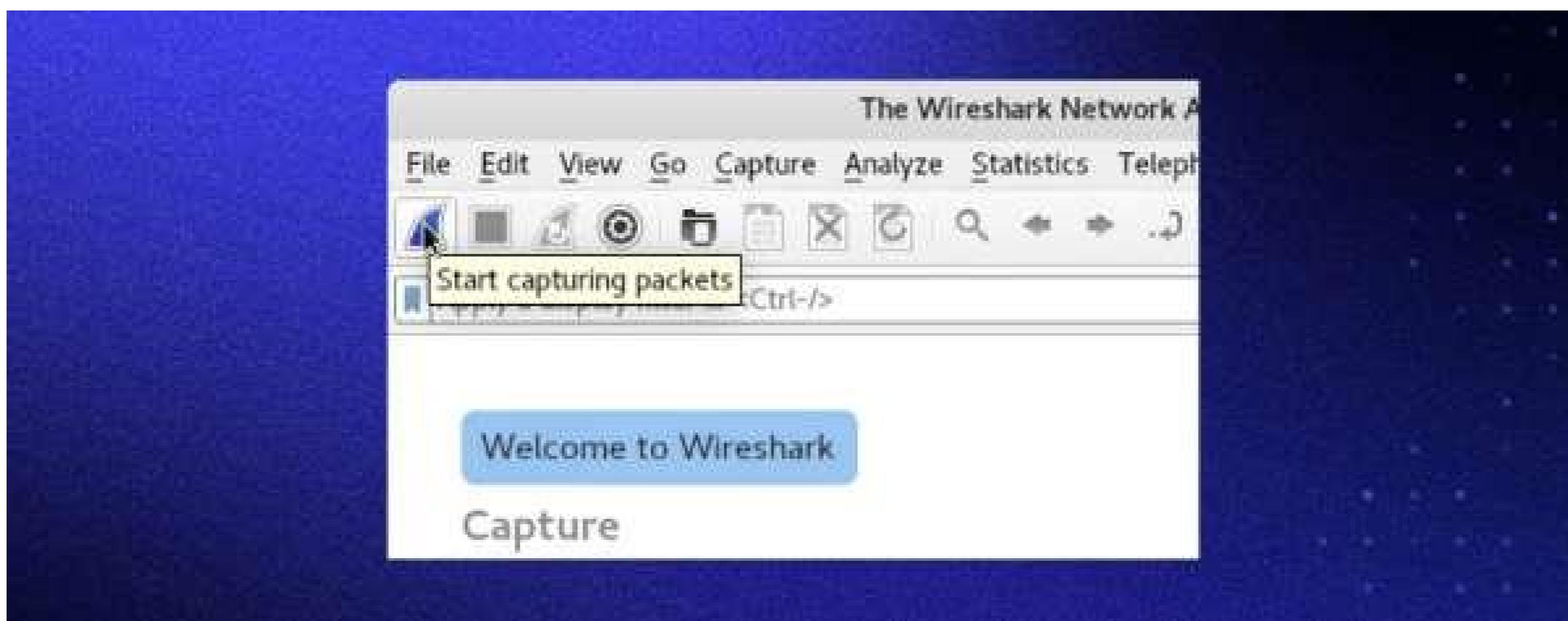
Capturing data packets on Wireshark

When you open Wireshark, you see a screen showing you a list of all the network connections you can monitor. You also have a capture filter field to only capture the network traffic you want to see.



You can select one or more of the network interfaces using shift+left-click. Once select the network interface, you can start the capture, and there are several ways to do that.

Click the first button on the toolbar, titled “Start capturing packets.”

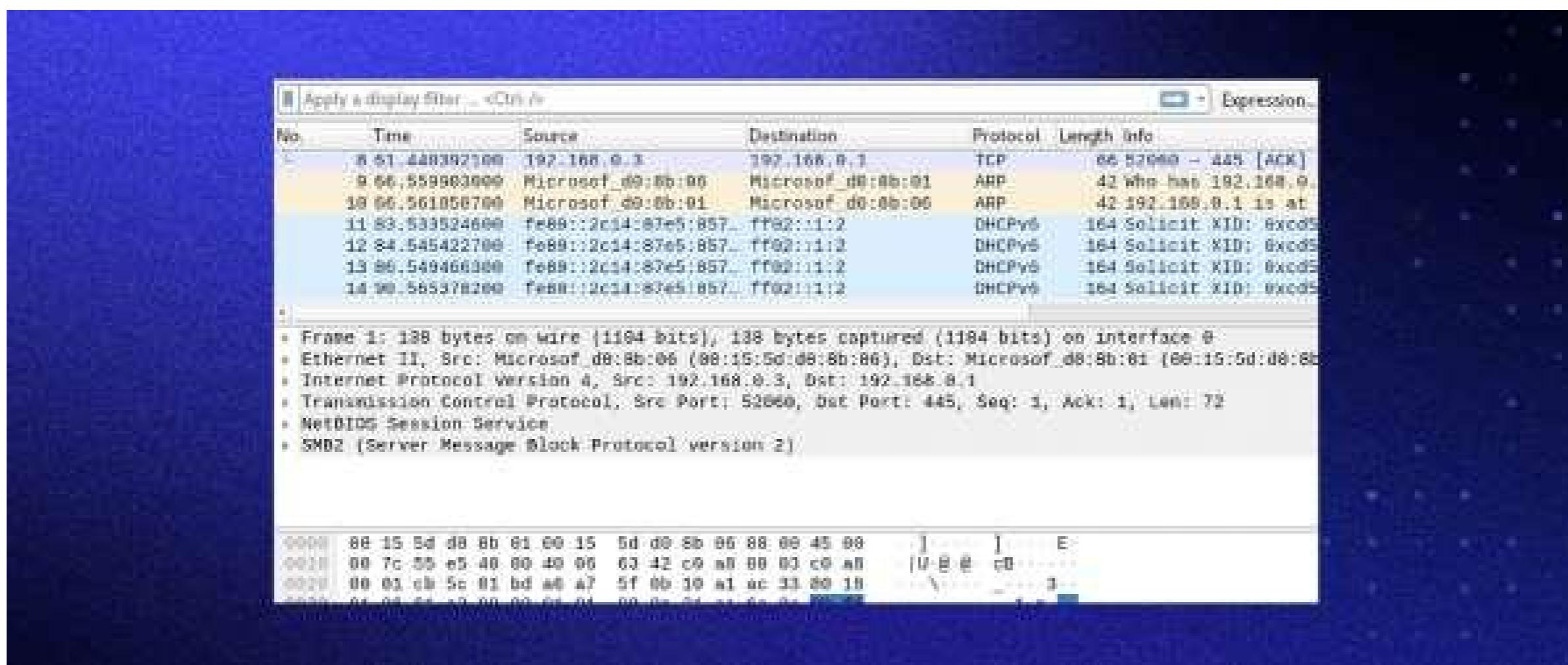


You can select the menu item Capture -> Start.



Or you could use the keystroke Control+E.

During the capture, Wireshark will show you the packets captured in real-time.



Once you have captured all the packets needed, use the same buttons or menu options to stop the capture as you did to begin.

Best practice dictates stopping Wireshark's packet capture before analysis.

Analyzing data packets on Wireshark

Wireshark shows you three different panes for inspecting packet data. The Packet List, the top pane, lists all the packets in the capture. When you click on a packet, the other two panes change to show you the details about the selected packet. You can also tell if the packet is part of a conversation. Here are details about each column in the top pane:

- No.:** This is the number order of the packet captured. The bracket indicates that this packet is part of a conversation.
- Time:** This column shows how long after you started the capture this particular packet was captured. You can change this value in the Settings menu to display a different option.
- Source:** This is the address of the system that sent the packet.
- Destination:** This is the address of the packet destination.
- Protocol:** This is the type of packet. For example: TCP, DNS, DHCPv6, or ARP.
- Length:** This column shows you the packet's length, measured in bytes.
- Info:** This column shows you more information about the packet contents, which will vary depending on the type of packet.

Wireshark filters

Some of the best features of Wireshark are the capture filters and display filters. Filters allow you to view the capture the way you need to see it to troubleshoot the issues at hand. Below are several filters to get you started. Wireshark filters

Some of the best features of Wireshark are the capture filters and display filters. Filters allow you to view the capture the way you need to see it to troubleshoot the issues at hand. Below are several filters to get

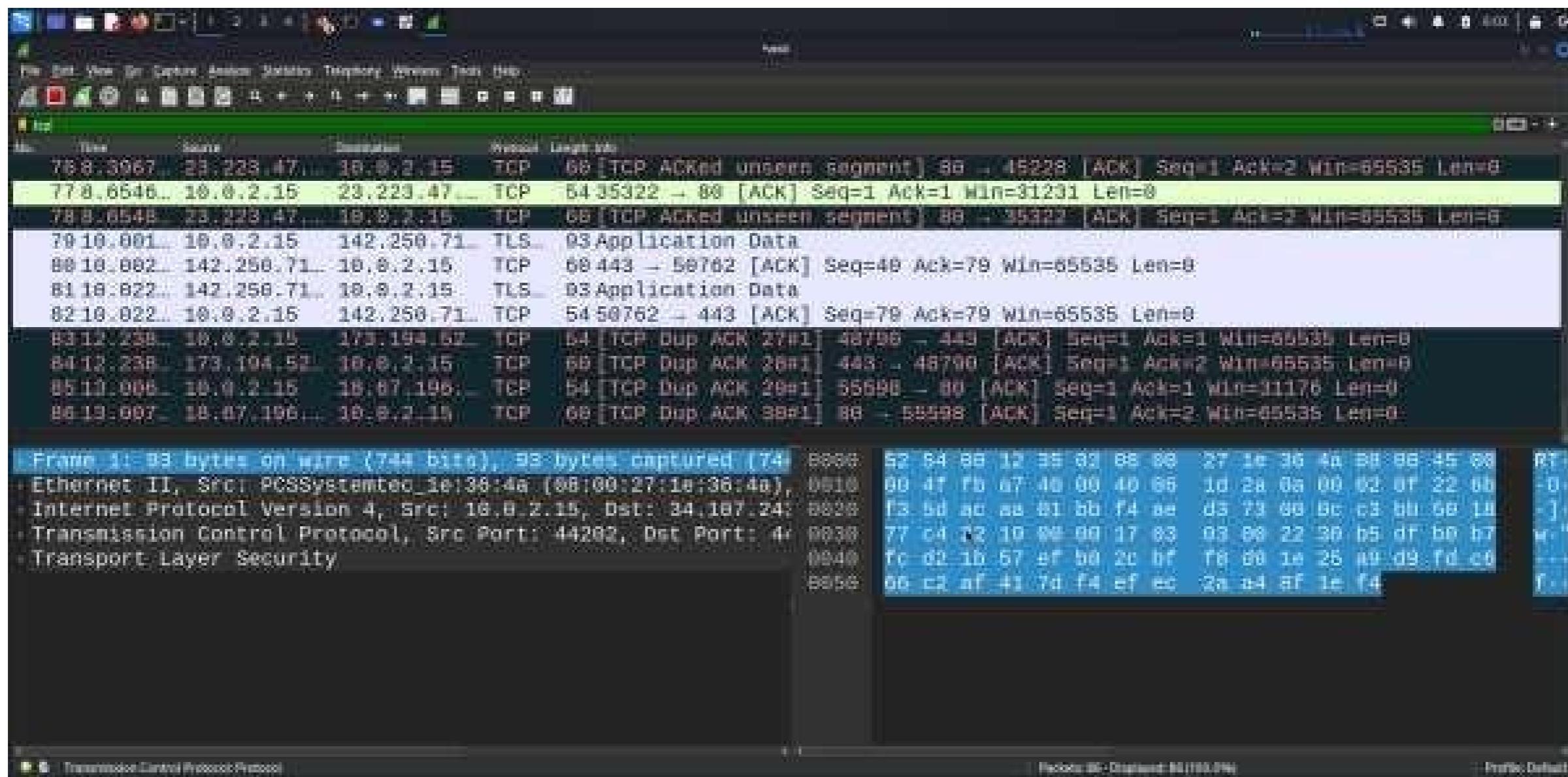
CYBER SECURITY LAB
you started.

BICL404

Wireshark capture filters

Capture filters limit the captured packets by the chosen filter. If the packets don't match the filter, Wireshark won't save them. Examples of capture filters include:

- host IP-address: This filter limits the captured traffic to and from the IP address
- net 192.168.0.0/24: This filter captures all traffic on the subnet
- dst host IP-address: Capture packets sent to the specified host
- port 53: Capture traffic on port 53 only
- port not 53 and not arp: Capture all traffic except DNS and ARP traffic



CYBER SECURITY LAB

BICL404

Program 5. Ransomware tabletop exercise on insider threat.

A ransomware tabletop exercise for an insider threat program involves simulating a ransomware attack originating from within the organization. The goal is to assess the readiness of the organization's response to such an incident and identify areas for improvement. Here are the steps to conduct such an exercise:

1. Planning and Preparation

Define Objectives: Establish what you aim to achieve with the exercise, such as testing response protocols, communication channels, and decision-making processes.

Form an Exercise Planning Team: Include representatives from IT, cybersecurity, HR, legal, and senior management. **Develop a Scenario:** Create a realistic and detailed ransomware attack scenario involving an insider threat. Include specifics like the type of ransomware, how it was deployed, and its impact on systems and data.

Prepare Materials: Create briefing documents, injects (pre-planned updates that move the scenario forward), and any other materials participants will need.

2. Participant Selection

Identify Participants: Select participants from relevant departments such as IT, cybersecurity, HR, legal, communications, and senior leadership.

Assign Roles: Define clear roles and responsibilities for each participant, ensuring they understand their part in the exercise.

3. Exercise Execution

Kick-off Meeting: Start with a briefing to explain the exercise objectives, scenario, and rules of engagement.

Scenario Introduction: Present the initial scenario to the participants. This could involve an insider (e.g., a disgruntled employee) deploying ransomware on the company's network.

Injects and Updates: Throughout the exercise, introduce new developments or complications (injects) to simulate how the situation might evolve. For example, injects could include ransomware spreading to critical systems, demands for ransom, or discovery of the insider's identity.

Facilitated Discussion: Encourage participants to discuss their actions, decisions, and thought processes. This helps uncover gaps in knowledge, communication breakdowns, or procedural issues.

4. Debrief and Evaluation

Hot Wash: Conduct an immediate debrief right after the exercise to gather initial impressions and feedback from participants while the experience is fresh.

After-Action Review (AAR): Schedule a more detailed review session to analyze the exercise in depth. Discuss what went well, what didn't, and why.

Lessons Learned: Identify key lessons learned and areas for improvement. Focus on enhancing the insider threat program, refining response protocols, and improving communication strategies.

5. Report and Follow-up

Document Findings: Compile a comprehensive report detailing the exercise scenario, participant actions, key findings, and recommendations.

Action Plan: Develop an action plan to address identified gaps and improve the organization's readiness.

Assign responsibilities and timelines for implementing changes.

~~CYBER SECURITY LAB~~ Follow-up: Monitor the progress of the action plan and schedule follow-up exercises to ensure continuous improvement. ~~RIGL404~~

By following these steps, you can conduct an effective ransomware tabletop exercise that enhances your organization's ability to respond to insider threats and ransomware incidents.

Program 6. Perform SQL Injection using Burpsuite

Open Terminal of your Kali Linux.

Give the command “**sudo apt-get install juice-shop**” .

Install juice shop from the above command, if you get any error then use the command “**sudo apt update**” , and retry till there’ s no error and complete the execution.

Once juice shop gets installed in your system, give the command “**juice-shop -h**” , you’ ll see that the firefox browser will open and Juice-shop site will be opened.

Now create your new account in that website.

Now click on Extensions and add “foxyproxy” extension to your browser.

Now again open terminal and give the command “**burpsuite**” and wait till the burpsuite window will open and click on OK and open burpsuite completely.

Go to proxy section in burpsuite and turn ON the interceptor.

Now come back to firefox and click on foxyproxy extension and click on options section.

Then click on proxies and click on add, you’ ll get a new window there enter Title and give your username and password which you had given at the time of login to juice-shop website.

Now give the hostname as “**127.0.0.1**” and host as “**8080**” .

Save the proxy and comeback to juice-shop website and log out of your account.

After logging out, click on extension -> proxy -> and enable the proxy that we had created.

After enabling it, again login to the juice-shop website with your credentials.

After logging IN, you came to burpsuite window and see changes in proxy section where we had turned on the interceptor.

Now right click in the same window and click on send to intruder, the intruder window will blink , click on it.

Now open firefox, come to new window and search “**sql.txt files for injection**” , open the first github website and download the code from there and save in download section.

Now come back to burpsuite window and come to intruder section where we have clicked the add option.

Now click on payload and click on load option, there select the downloaded sql.txt file from download section, then click on START ATTACK.

Wait for some time, the attack starts and note down the readings as OUTPUT.

OUTPUT

2. Intruder attack of http://127.0.0.1:42000					
Results	Positions	Payloads	Resource pool	Settings	
Filter: Showing all items					
Requ...	Payload	Status code	Error	Timeout	Length
0		200	<input type="checkbox"/>	<input type="checkbox"/>	736
1	'		<input type="checkbox"/>	<input type="checkbox"/>	
2	"		<input type="checkbox"/>	<input type="checkbox"/>	
3	#		<input type="checkbox"/>	<input type="checkbox"/>	
4	-		<input type="checkbox"/>	<input type="checkbox"/>	
5	--		<input type="checkbox"/>	<input type="checkbox"/>	
6	'%20--		<input type="checkbox"/>	<input type="checkbox"/>	
7	-';		<input type="checkbox"/>	<input type="checkbox"/>	
8	'%20;		<input type="checkbox"/>	<input type="checkbox"/>	
9	=%20'		<input type="checkbox"/>	<input type="checkbox"/>	
10	=%20;		<input type="checkbox"/>	<input type="checkbox"/>	
11	=%20--		<input type="checkbox"/>	<input type="checkbox"/>	
12	\v23		<input type="checkbox"/>	<input type="checkbox"/>	

Program 7. Installation of Wire shark, tcpdump, etc and observe data transferred in client server communication using UDP/TCP and identify the UDP/TCP datagram.

Tcpdump is a command line utility that allows you to capture and analyze network traffic going through your system. It is often used to help troubleshoot network issues, as well as a security tool.

A powerful and versatile tool that includes many options and filters, tcpdump can be used in a variety of cases. Since it's a command line tool, it is ideal to run in remote servers or devices for which a GUI is not available, to collect data that can be analyzed later.

It can also be launched in the background or as a scheduled job using tools like cron. Topics:

1. Installation on Linux

2. Capturing packets with tcpdump

```
sudo tcpdump -interface any
```

```
$ sudo tcpdump -i any -c 5
```

```
$ sudo tcpdump -i any -c5 -nn
```

3. Understanding the output format

4. Filtering packets

4.1 Protocol - \$ sudo tcpdump -i any -c5 icmp

4.2 Host \$ sudo tcpdump -i any -c5 -nn host 54.204.39.132

4.3 Port \$ sudo tcpdump -i any -c5 -nn port 80

4.4 Source IP/hostname \$ sudo tcpdump -i any -c5 -nn src 192.168.122.98

4.5 Complex expressions \$ sudo tcpdump -i any -c5 -nn src 192.168.122.98 and port80

5. Checking packet content - \$ sudo tcpdump -i any -c10 -nn -A port 80

6. Saving (Writing) & Opening(Reading) packets

```
$ sudo tcpdump -i any -c10 -nn -w webserver.pcap port 80
```

```
$ tcpdump -nn -r webserver.pcap
```

Steps to follow

Search Wireshark in the browser.

Install Wireshark windows x64 installer in the system.

Then go to Wireshark application and open it.

Wireshark network analyser gets displayed.

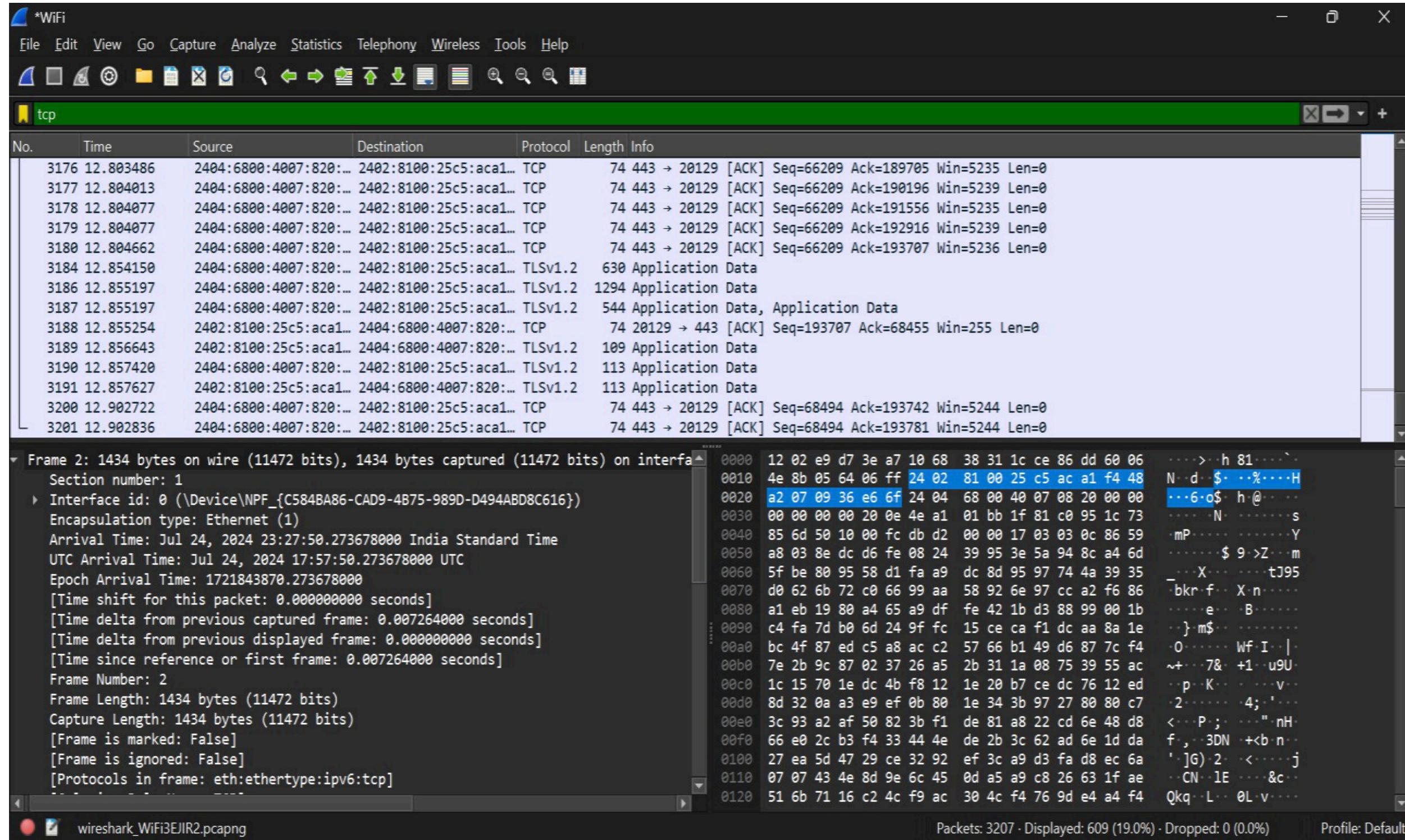
Select any files such as ethernet, adapter for loopback traffic capture etc.

Various packets in the selected file in them, select any one of the packet and capture them.

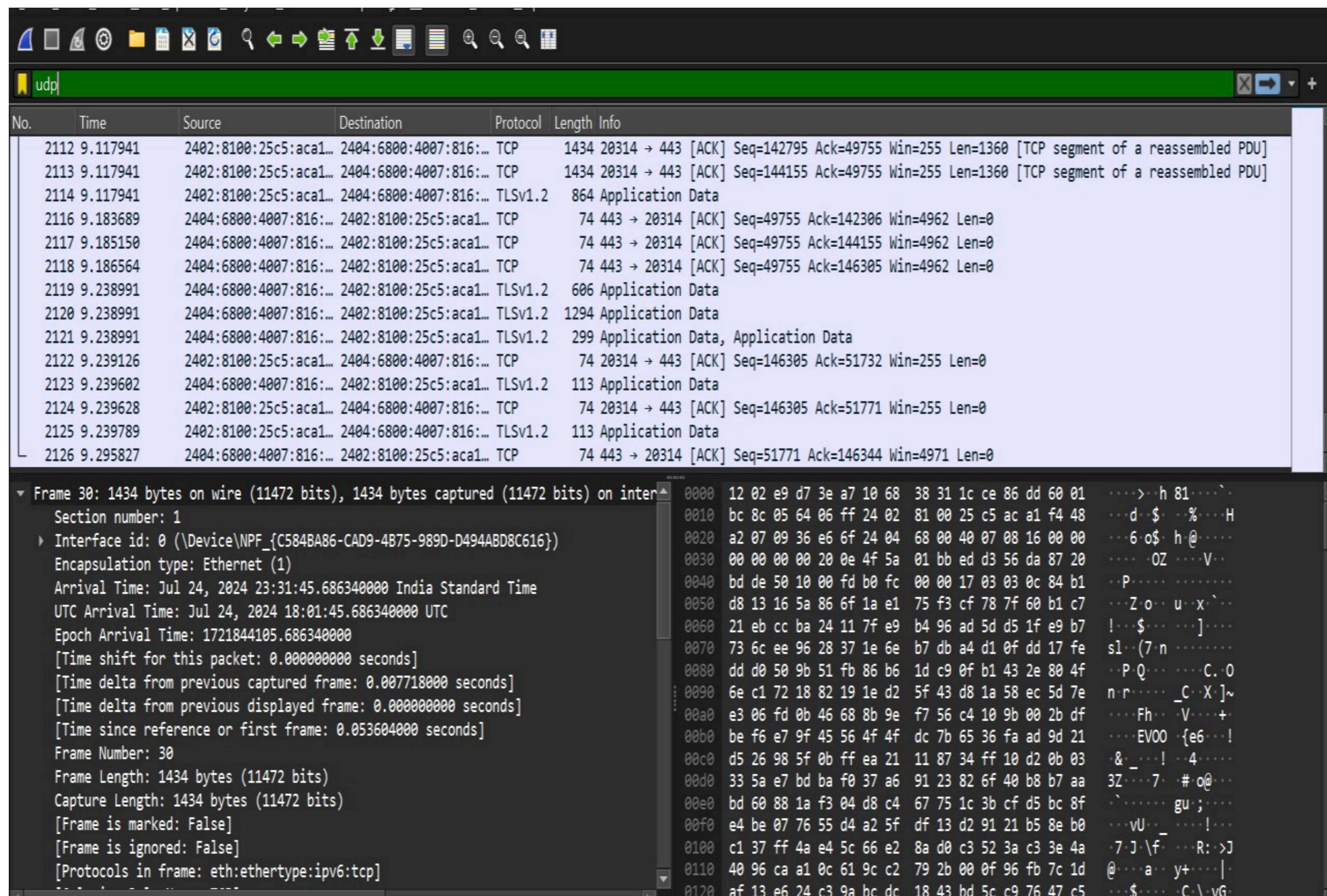
CYBER SECURITY LAB

Open search column and enter “tcp” and note down the results.

BICL404

OUTPUT:

Open search column and enter “**udp**” and note down the results.



Program 8. Installation of Rootkits and study about the varieties of options.

Rkhunter (Rootkit Hunter) is an open-source Unix/Linux based scanner tool for Linux systems released under GPL that scans backdoors, rootkits, and local exploits on your systems.

It scans hidden files, wrong permissions set on binaries, suspicious strings in the kernel, etc.

Rootkit hunter is a well-known tool for checking vulnerabilities, rootkits, back doors, and possible local exploits on a server. It is possible to use it on *any* server used for *any* purpose. When tuned and automated, it can report any suspicious activity to the system administrator. This procedure outlines the installation, tuning, and use of rootkit hunter. rkhunter is just one possible part of a hardened server setup. Use it alone or with other tools to maximize security.

Installation of rootkits and study about the variety of option

A. Steps to install Rootkits

- Open Root terminal in Kali Linux.
- Give the command “chkrootkit” and install it (if it’s not downloaded) using command “sudo apt-get install chkrootkit” .
- Give the command “rkhunter” and install it (if it’s not downloaded) using command “sudo apt-get install rkhunter” .

NOTE : if error occurs while installation, retry installing the files with the same commands!

Options of Rootkits are:

Command: “chkrootkit -h”

Output:

```
[root@kali:~]
# chkrootkit -h
Usage: /usr/sbin/chkrootkit [options] [test ...]
Options:
-h      show this help and exit
-V      show version information and exit
-l      show available tests and exit
-d      debug
-q      quiet mode
-x      expert mode
-e 'FILE1 FILE2' exclude files/dirs from results. Must be followed by a space-separated list of files/dirs.
Read /usr/share/doc/chkrootkit/README.FALSE-POSITIVES first.
-s REGEXP filter results of sniffer test through 'grep -Ev REGEXP' to exclude expected
PACKET_SNIFFERS. Read /usr/share/doc/chkrootkit/README.FALSE-POSITIVES first.
-r DIR   use DIR as the root directory
-p DIR1:DIR2:DIRN path for the external commands used by chkrootkit
-n      skip NFS mount points
-T FSTYPE skip mount points of the specified file system type
```

Command: "rkhunter -h"

Output:

```
(root㉿kali)-[~]
# rkhunter -h

Usage: rkhunter {--check | --unlock | --update | --versioncheck |
                  --propupd [{filename | directory | package name}, ...] |
                  --list [{tests | {lang | languages} | rootkits | perl | propfiles}] |
                  --config-check | --version | --help} [options]

Current options are:
  --append-log          Append to the logfile, do not overwrite
  --bindir <directory>... Use the specified command directories
  -c, --check            Check the local system
  -C, --config-check    Check the configuration file(s), then exit
  --cs2, --color-set2    Use the second color set for output
  --configfile <file>    Use the specified configuration file
  --cronjob              Run as a cron job
                         (implies -c, --sk and --nocolors options)
  --dbdir <directory>   Use the specified database directory
  --debug                Debug mode
                         (Do not use unless asked to do so)
  --disable <test>[,<test> ...] Disable specific tests
                         (Default is to disable no tests)
  --display-logfile     Display the logfile at the end
  --enable <test>[,<test> ...] Enable specific tests
                         (Default is to enable all tests)
  --hash {MD5 | SHA1 | SHA224 | SHA256 | SHA384 | SHA512 |
          NONE | <command>} Use the specified file hash function
                         (Default is SHA256)
  -h, --help              Display this help menu, then exit
  --lang, --language <language> Specify the language to use
                         (Default is English)
  --list [tests | languages |
         rootkits | perl | propfiles] List the available test names, languages,
                                         rootkit names, perl module status
                                         or file properties database, then exit
  -l, --logfile [file]    Write to a logfile
                         (Default is /var/log/rkhunter.log)
  --noappend-log          Do not append to the logfile, overwrite it
  --nocf                 Do not use the configuration file entries
                         for disabled tests (only valid with --disable)
  --nocolors              Use black and white output
  --nolog                Do not write to a logfile
  --nomow, --no-mail-on-warning Do not send a message if warnings occur
  --ns, --nosummary       Do not show the summary of check results
  --novl, --no-verbose-logging No verbose logging
  --pkgmgr {RPM | DPKG | BSD |
            BSDng | SOLARIS | NONE} Use the specified package manager to obtain
                                         or verify file property values.
                                         (Default is NONE)
  --propupd [file | directory | package]... Update the entire file properties database,
                                         or just for the specified entries
                                         Quiet mode (no output at all)
  --rwo, --report-warnings-only Show only warning messages
  --sk, --skip-keypress    Don't wait for a keypress after each test
  --summary               Show the summary of system check results
```

Program 9. Perform an experiment to Sniff Traffic using ARP poisoning.

Ettercap is a free, open-source tool that can be used for man-in-the-middle attacks on networks. As such, it can be a threat to network security. However, network administrators need to be aware of this tool to check the vulnerabilities of their systems.

Install Ettercap on Kali Linux

If you have Kali Linux, there isn't anything that you need to do to install Ettercap. It is already installed. Install Ettercap on Ubuntu Linux

Go to the command line and enter the two commands:

sudo apt update

sudo apt install ettercap-common

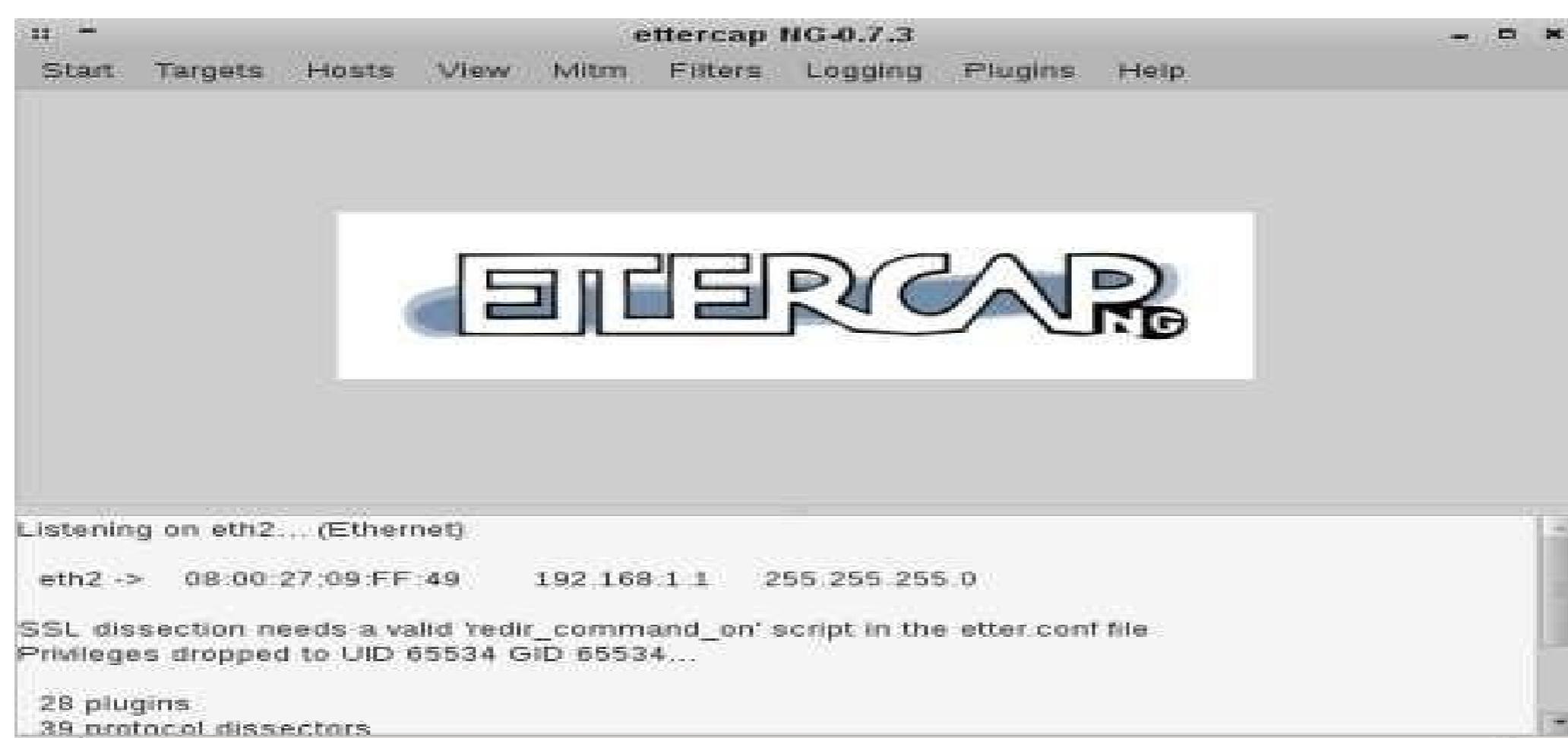
-common for console...

sudo apt install ettercap-graphical graphical for GUI

start the Ettercap

\$sudo ettercap -G

After clicking "OK" , you will be back to the main window:



Open Kali Linux software in your systems.

Start Root terminal by giving the password for it.

Inside the Root terminal give the command "ettercap -G" .

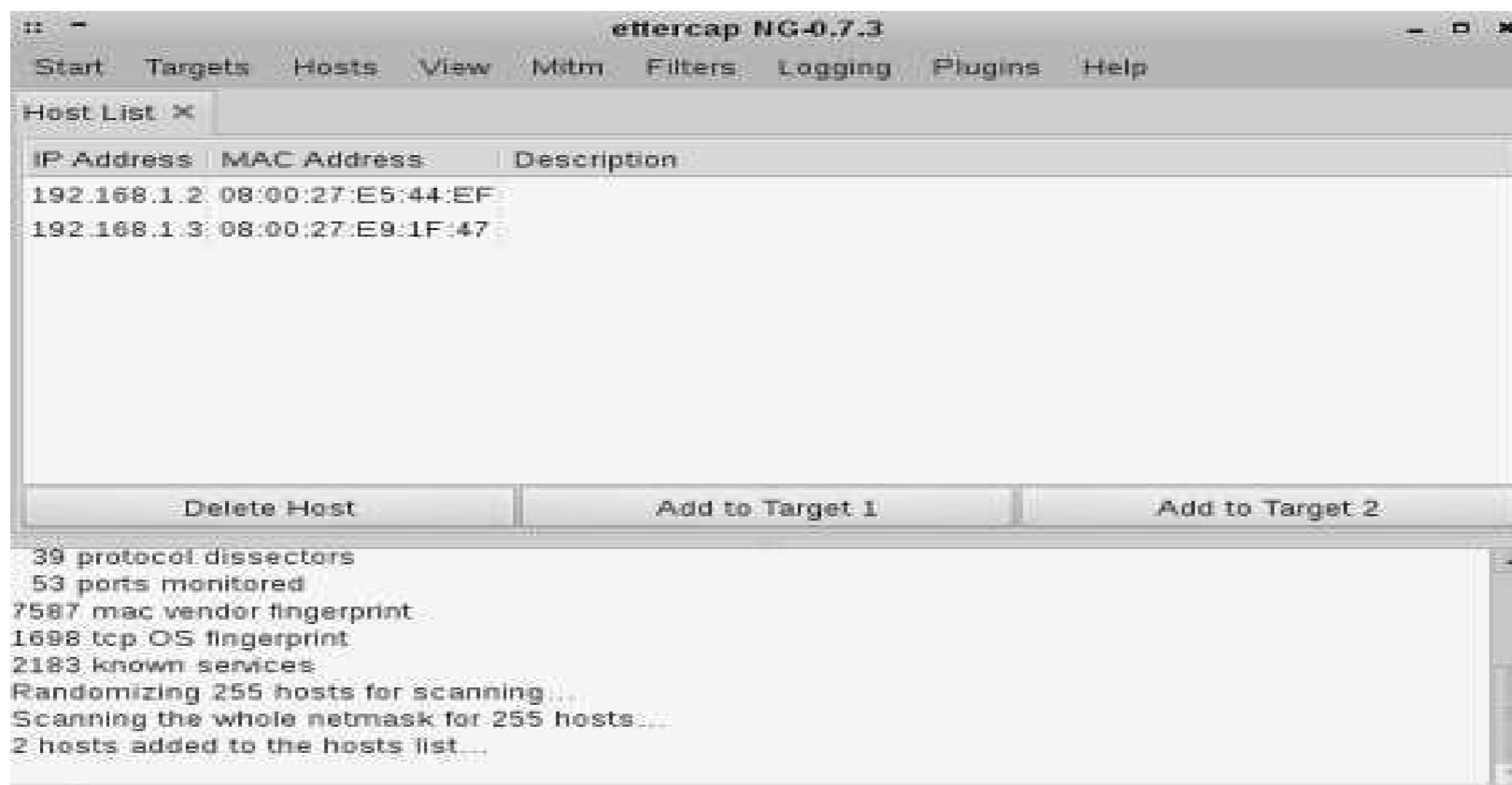
You'll get a new Window of Ettercap.

The next step is host scanning. Click the "Hosts" menu and then click "Scan for hosts" .

When the scan is finished, click the "Hosts" menu and then click on "Host List"

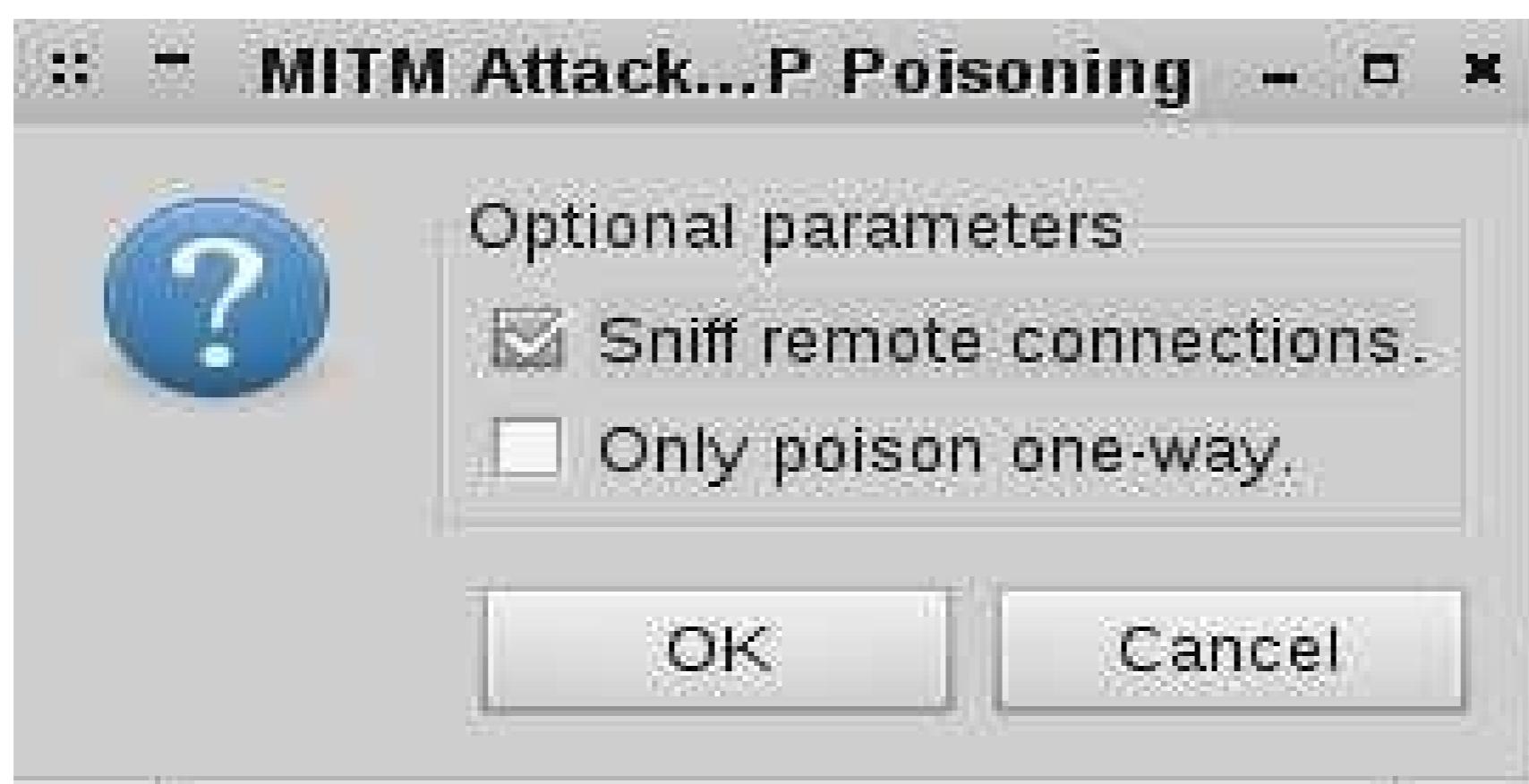
CYBER SECURITY LAB

BICL404



As you see, Ettercap found two hosts on my network. Click on the first host and click the “Add to Target 1” button and then click on the second host and click the “Add to Target 2” button.

Click the “Mitm” menu and select “Arp Poisoning” then select “Sniff Remote Connection” and click “OK”



Next, click the “Start” menu and select “start sniffing” .

In this scenario we computer seem 192.168.1.3 and when two targets communicate together, packets are sent to our computer too. You can test it via Wireshark tool. See the ARP table via “arp -a” command.



**Now inside the window, navigate to the right top corner and click on three dots.
And go for Hosts and scan for Hosts.**



**After scanning again click on hosts and open the scanned hosts.
Now select the scanned hosts and add them to target 1 and target 2.**

The screenshot shows the Ettercap interface version 0.8.3.1 (EB). At the top, there are icons for E, W, Q, and a list. The title bar reads "Ettercap 0.8.3.1 (EB)". On the left, a "Host List" tab is selected. The main area displays a table with three columns: IP Address, MAC Address, and Description. The table contains three rows of data:

IP Address	MAC Address	Description
10.0.2.2	52:54:00:12:35:02	
10.0.2.3	52:54:00:12:35:03	
10.0.2.4	52:54:00:12:35:04	

Below the table, there are three buttons: "Delete Host", "Add to Target 1" (which is highlighted in blue), and "Add to Target 2". A status message at the bottom left says "Randomizing 255 hosts for scanning... Scanning the whole netmask for 255 hosts... 8 hosts added to the hosts list... Host 10.0.2.2 added to TARGET1 Host 10.0.2.3 added to TARGET2".

Check the results above for the output.

Program 10. Demonstrate intrusion detection system using snort.

Cybersecurity is an important issue for both academics and practitioners, since successful cyberattacks may result in astronomical expenditures owing to the loss of confidentiality, integrity, or availability. Various security approaches have been proposed for detecting cyberattacks, with intrusion detection systems (IDS) and network-based intrusion detection systems (NIDS) being among the most prevalent.

Snort is a powerful and lightweight open-source network intrusion detection and prevention system(IDS/IPS) which provides network traffic analysis and packet recording in real time.

There are various features that make SNORT useful for network admins to monitor their systems and detect malicious activity. These include:

1. Real-time traffic monitor: SNORT can be used to monitor the traffic that goes in and out of a network. It will monitor traffic in real time and issue alerts to users when it discovers potentially malicious packets or threats on Internet Protocol (IP) networks.
2. Packet logging: SNORT enables packet logging through its packet logger mode, which means it logs packets to the disk. In this mode, SNORT collects every packet and logs it in a hierarchical directory based on the host network's [IP address](#).
3. Analysis of protocol: SNORT can perform protocol analysis, which is a network sniffing process that captures data in protocol layers for additional analysis.
4. Content matching: SNORT collates rules by the protocol, such as IP and TCP, then by ports, and then by those with content and those without.
5. OS fingerprinting: Operating system (OS) fingerprinting uses the concept that all platforms have a unique TCP/IP stack. Through this process, SNORT can be used to determine the OS platform being used by a system that accesses a network.
6. Can be installed in any network environment: SNORT can be deployed on all operating systems, including Linux and Windows, and a part of all network environments.

There two OS(machine)s involved in this experiment

First Machine Ubuntu _ You need to download Ubuntu for virtual box and then add this ubuntu in virtual box. Start the Ubuntu machine in virtual box and follow the below mentioned instructions for executing 10th program

Open the terminal and type ip a or ifconfig command to get the ip address ubuntu

machine. Second Machine Kali: You should also start the kali machine.

Installation of Snort in Ubuntu machine: type the following command in terminal.


```
sudo apt update  
sudo apt install  
snort
```

To get the version of the snort tool

```
$sudo snort - version
```

Run the following command to check the snort configuration is working file

```
$ snort -T -c /etc/snort/snort.conf
```

To start the snort tool, use the following command

```
$snort -q -A console -c /etc/snort/snort.conf
```

Now snort tool on ubuntu start scanning the incoming packets Next you need ping the ubuntu machine from kali machine.

@kali machine terminal

Ping 10.0.2.15 (my ubuntu machine ip address- you should give your ubuntu machine ip address)

```
File Actions Edit View Help  
Kali machine  
ping 10.0.2.15  
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.  
64 bytes from 10.0.2.15: icmp_seq=1 ttl=64 time=0.032 ms  
64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=0.029 ms  
64 bytes from 10.0.2.15: icmp_seq=3 ttl=64 time=0.028 ms  
64 bytes from 10.0.2.15: icmp_seq=4 ttl=64 time=0.030 ms  
64 bytes from 10.0.2.15: icmp_seq=5 ttl=64 time=0.030 ms  
64 bytes from 10.0.2.15: icmp_seq=6 ttl=64 time=0.028 ms  
64 bytes from 10.0.2.15: icmp_seq=7 ttl=64 time=0.034 ms  
64 bytes from 10.0.2.15: icmp_seq=8 ttl=64 time=0.054 ms  
64 bytes from 10.0.2.15: icmp_seq=9 ttl=64 time=0.028 ms  
64 bytes from 10.0.2.15: icmp_seq=10 ttl=64 time=0.033 ms  
64 bytes from 10.0.2.15: icmp_seq=11 ttl=64 time=0.027 ms  
64 bytes from 10.0.2.15: icmp_seq=12 ttl=64 time=0.031 ms  
64 bytes from 10.0.2.15: icmp_seq=13 ttl=64 time=0.029 ms  
64 bytes from 10.0.2.15: icmp_seq=14 ttl=64 time=0.068 ms  
64 bytes from 10.0.2.15: icmp_seq=15 ttl=64 time=0.031 ms  
64 bytes from 10.0.2.15: icmp_seq=16 ttl=64 time=0.029 ms  
64 bytes from 10.0.2.15: icmp_seq=17 ttl=64 time=0.052 ms  
64 bytes from 10.0.2.15: icmp_seq=18 ttl=64 time=0.066 ms  
64 bytes from 10.0.2.15: icmp_seq=19 ttl=64 time=0.024 ms  
64 bytes from 10.0.2.15: icmp_seq=20 ttl=64 time=0.063 ms  
64 bytes from 10.0.2.15: icmp_seq=21 ttl=64 time=0.051 ms  
64 bytes from 10.0.2.15: icmp_seq=22 ttl=64 time=0.025 ms  
64 bytes from 10.0.2.15: icmp_seq=23 ttl=64 time=0.027 ms
```

At Ubuntu machine

We can also add custom rules to /etc/snort/rules/local.rules file. Run the command to edit the local.rule file in /etc/snort/rules folder.

```
$sudo nano
```

/etc/snort/rules/local.rules Add the

following statement at the end

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:" ICMP DETECTED KSIT" ; sid:10000002;rev:2)
```

```

# Sid: local.rules,v 1.11 2008/07/23 20:15:44 bmc Exp $ 
#
# LOCAL RULES
#
# This file intentionally does not come with signatures. Put your local
# additions here.
alert icmp any any -> any any (msg:"ICMP detected KSIT";sid:10000002;rev:2);

```

The terminal window shows the nano editor interface with various menu options like Help, Exit, Write Out, Read File, Where Is Replace, Cut Paste, Undo, Redo, Execute Justify, and Find.

You can add any number of rules. Another rule is for port number 21.

`alert tcp any any -> $HOME_NET 21 (msg:" FTP ATTEMPT DETECTED KSIT" ; sid:10000002;rev:1;)`

Then save and exit the file by pressing **ctrl O** and then press enter to save the file. The exit the nano editor by pressing **ctrl X**.

You need to change the permission to execute the updated file. So grant read, write and execute permission to all using **chmod 777**.

`$sudo chmod 777 /etc/snort/snort.conf`

Now run the configuration. If the snort.file is updated properly, you will get any errors. Otherwise you will get errors.

Then run the snort tools using

\$ sudo snort -A console -c /etc/snort/snort.conf

```

Jul 14 14:21 user@ubuntu:~ 
can] [Priority: 3] (UDP) 192.168.56.1:50509 -> 239.255.255.258:1900 
07/14 14:19:51.099341 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::10 
07/14-14:19:51.099364 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::2 
07/14-14:19:51.097334 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::16 
07/14-14:19:51.210511 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::36 
07/14-14:19:51.0442309 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::36 
07/14-14:19:51.044241 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network 
can] [Priority: 3] (UDP) 192.168.56.1:50505 -> 239.255.255.258:1900 
07/14-14:19:51.044240 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network 
can] [Priority: 3] (UDP) 192.168.56.1:50509 -> 239.255.255.258:1900 
07/14-14:19:52.044611 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network 
can] [Priority: 3] (UDP) 192.168.56.1:50505 -> 239.255.255.258:1900 
07/14-14:19:52.044614 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network 
can] [Priority: 3] (UDP) 192.168.56.1:50509 -> 239.255.255.258:1900 
07/14-14:19:53.044604 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network 
can] [Priority: 3] (UDP) 192.168.56.1:50505 -> 239.255.255.258:1900 
07/14-14:19:53.044604 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network 
can] [Priority: 3] (UDP) 192.168.56.1:50509 -> 239.255.255.258:1900 
07/14-14:19:55.078494 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::2 
07/14-14:20:04.594379 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::2 
07/14-14:20:22.514120 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::2 
07/14-14:21:01.938173 [**] [1:10000002:2] ICMP detected KSIT [**] (Priority: 0) (IPV6-ICMP) F000::a00:27ff:fe07:b001 
FF02::2

```

You can see the custom message with author name with sid: 10000002 and revision number 2.