

Unit-2

System and Network Vulnerability

Basic Terminology

▶ IP Address

- An **Internet Protocol address** (IP address) is a numerical label assigned to each device (e.g., computer, printer) participating in a computer network that uses the **Internet Protocol for communication**.
- An IP address serves two principal functions: **host or network interface identification and location addressing**.

▶ Two Version of IP address:

- IPv4
- IPv6

▶ IPv4 uses **32-bit** for address. **Example:** 192.168.1.1

▶ IPv6 uses **128-bit** for address. **Example:** 2001:0db8:85a3:0000:0000:8a2e:0370:7334

▶ IP addresses are usually written and displayed in human-readable notations.

Public and private IP address

- ▶ **Private IP address** of a system is the IP address that is used to communicate within the same network. Using private IP data or information can be sent or received within the same network.
- ▶ **Public IP address** of a system is the IP address that is used to communicate outside the network. A public IP address is basically assigned by the ISP (Internet Service Provider).
- ▶ **Difference between Private and Public IP address:**

S.No.	PRIVATE IP ADDRESS	PUBLIC IP ADDRESS
1.	The scope of Private IP is local.	The scope of Public IP is global.
2.	It is used to communicate within the network.	It is used to communicate outside the network.

Public and private IP address

► Difference between Private and Public IP address:

- | | | |
|----|--|--|
| 3. | Private IP addresses of the systems connected in a network differ in a uniform manner. | Public IP may differ in a uniform or non-uniform manner. |
| 4. | It works only on LAN. | It is used to get internet service. |
| 5. | It is used to load the network operating system. | It is controlled by ISP. |

Public and private IP address

▶ Difference between Private and Public IP address:

- | | | |
|----|--|---|
| 6. | Private IP can be known by entering "ipconfig" on the command prompt. | Public IP can be known by searching "what is my ip" on google. |
| 7. | <p>Range:
10.0.0.0 – 10.255.255.255,
172.16.0.0 –
172.31.255.255,
192.168.0.0 –
192.168.255.255</p> <p>Example: 192.168.1.10</p> | <p>Range: Besides private IP addresses, the rest are public.</p> <p>Example: 17.5.7.8</p> |

Basic Terminology

▶ MAC Address

- ➔ A **media access control address** (MAC address) is a unique identifier assigned to network interfaces for communications on the physical network segment.
- ▶ MAC addresses are used as a **network address** for most IEEE 802 network technologies, including Ethernet, Wi-Fi & Bluetooth.
- ▶ It is also known as **physical** address or **hardware** address.
- ▶ The MAC address is a string of usually **six sets of two-digits or characters**, separated by colons.
- ▶ For example, consider a network adapter with the MAC address 01:0a:95:9d:58:36.

Basic Terminology

Difference between IP Address vs MAC Address



Film Maker

<u>IP Address</u>	<u>MAC Address</u>
1. IP stands for Internet Protocol.	1. MAC stands for Media Access Control.
2. It is a Logical Address.	2. It is a Physical Address.
3. It is provided by the Internet Service Provider(ISP)	3. It is provided by Comp. Manufacturer.
4. It can be changed by changing ISP.	4. MAC Address is fixed Address for a particular device.
5. It has various classes like A,B,C,D,E.	5. It has no class concept.
6. It is applicable on Network Layer of OSI Model	6.It is applicable on Data link Layer of OSI Model.
7. The Length of IPv4 is 32 bits. The Length of IPv6 is 128 bits.	7. The length of MAC Address is 48 bits.

Activat
Go to Set

Basic Terminology

▶ **Computer Network:**

- A computer network is a telecommunications network which allows **computers to exchange data**.
- ▶ In computer networks, networked computing devices exchange data with each other along network links (data connections).
- ▶ The connections between nodes are established using either **cable media or wireless media**.
- ▶ The best-known computer network is the **Internet**.

▶ **Computer Port:**

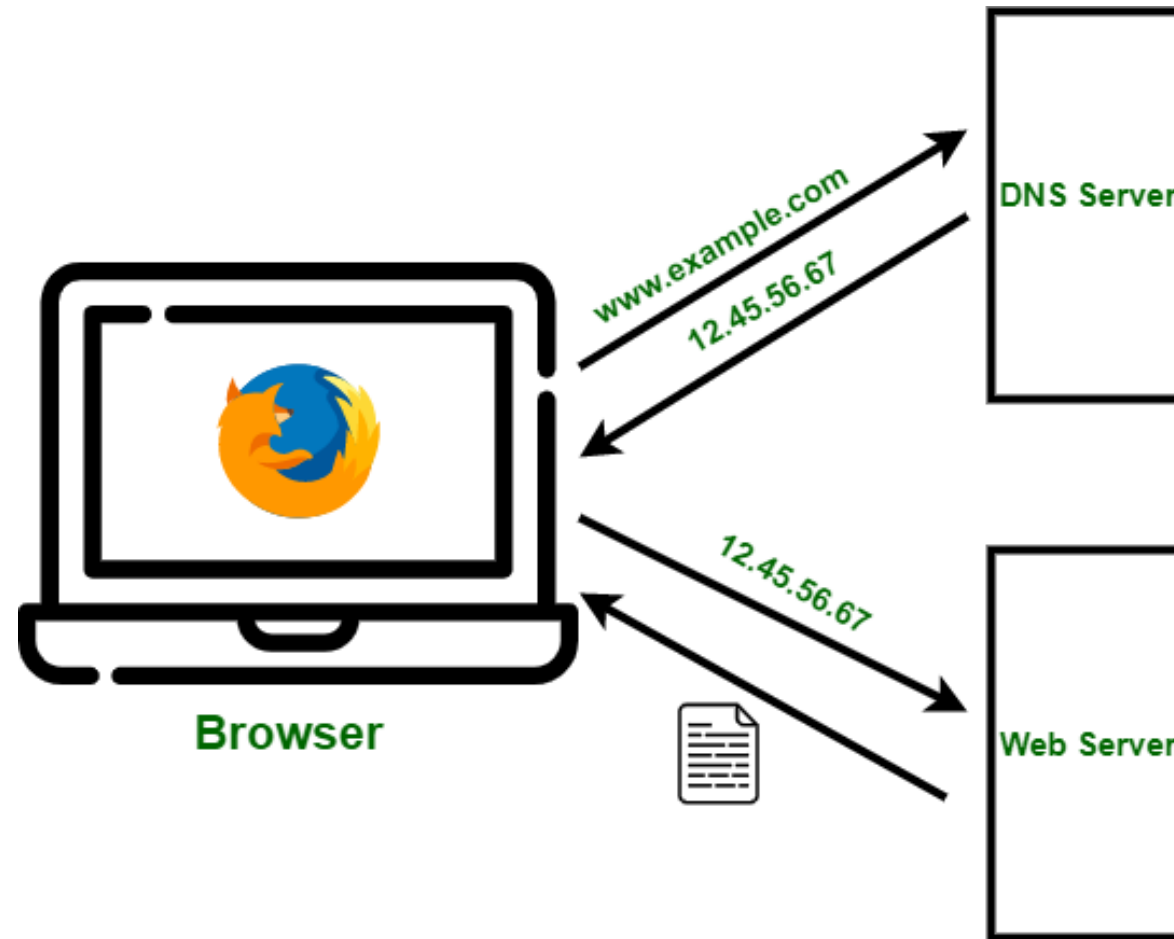
- In computer hardware, a port serves as an interface between the computer and other computers or peripheral devices.
- ▶ Computer ports have many uses, to connect a monitor, webcam, speakers, or other peripheral devices.
- ▶ On the physical layer, a computer port is a specialized interface on a piece of equipment to which a plug or cable connects.

Basic Terminology

- ▶ DNS stand for “domain name system”.
- ▶ It converting human-readable website name into **computer-readable numerical IP addresses**.
- ▶ For example:
 - ➞ If you want to visit Google, then open `www.google.com` into your web browser’s address bar instead of IP address. However, your computer does not understand where `www.google.com` is located.
- ▶ Behind the scenes, the internet and other network use numerical IP addresses. `www.google.com` is located at the IP address `73.194.39.78` on the internet.

Basic Terminology

- ▶ DNS stand for “domain name system”.



Overview of Vulnerability Scanning

- ▶ Vulnerability

- ➔ vulnerability is a **weakness** which allows an attacker to reduce a system's security.

- ▶ Vulnerability scanning usually refers to the **scanning of systems** that are connected to the **Internet**.

- ▶ Example: **A weakness in a firewall that lets hackers get into a computer network**

- ▶ It can also refer to system scanning or audits on internal networks that are not connected to the Internet in order to assess the **threat of malicious software**.

- ▶ It is possible to know the basic security measures when installing and managing network and websites. but it is not possible to catch all the vulnerabilities reside in the network and websites.

Overview of Vulnerability Scanning

When Attackers Target Vulnerabilities



Attacker creates exploits
to target software
vulnerability



OR



1. Exploits may arrive via:
- Attachment to email messages
 - Compromised websites
 - Social networking sites

2. Attacker may directly
target vulnerable servers



Users are lured into executing
the exploit via social
engineering techniques



OR



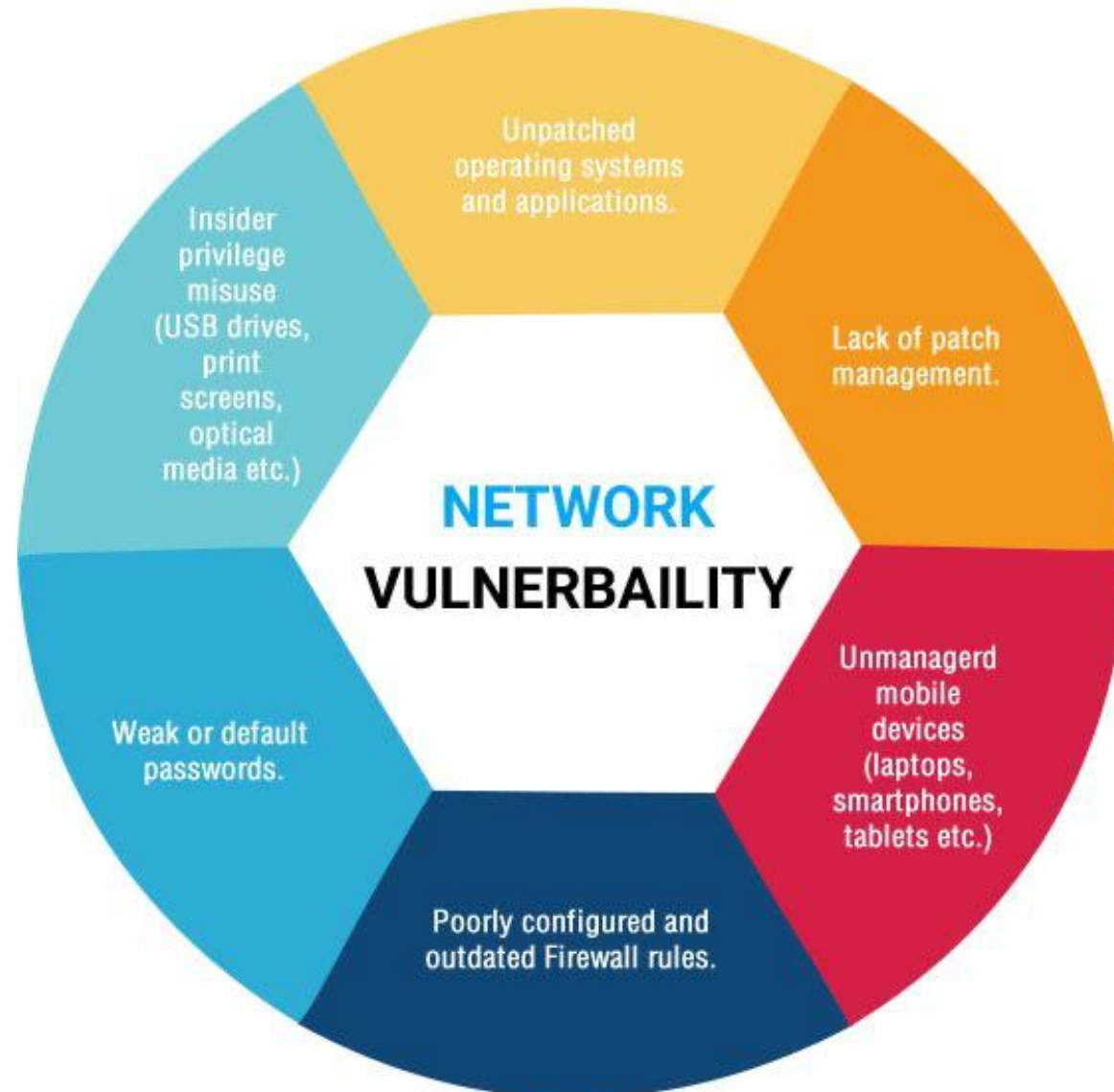
Exploits may drop malware
onto the vulnerable system or
allow attackers remote
control

Overview of Vulnerability Scanning – Cont.

- ▶ The vulnerability scanners provide you the **automate security** auditing and play an important role in your IT security.
- ▶ The vulnerability scanners can scan your network and websites for up to thousands of different security risks.
- ▶ It produces a list of those vulnerabilities, and gives steps on **how to overcome or reduce** them.
- ▶ Examples: Burp suit, Open VAS, Nmap

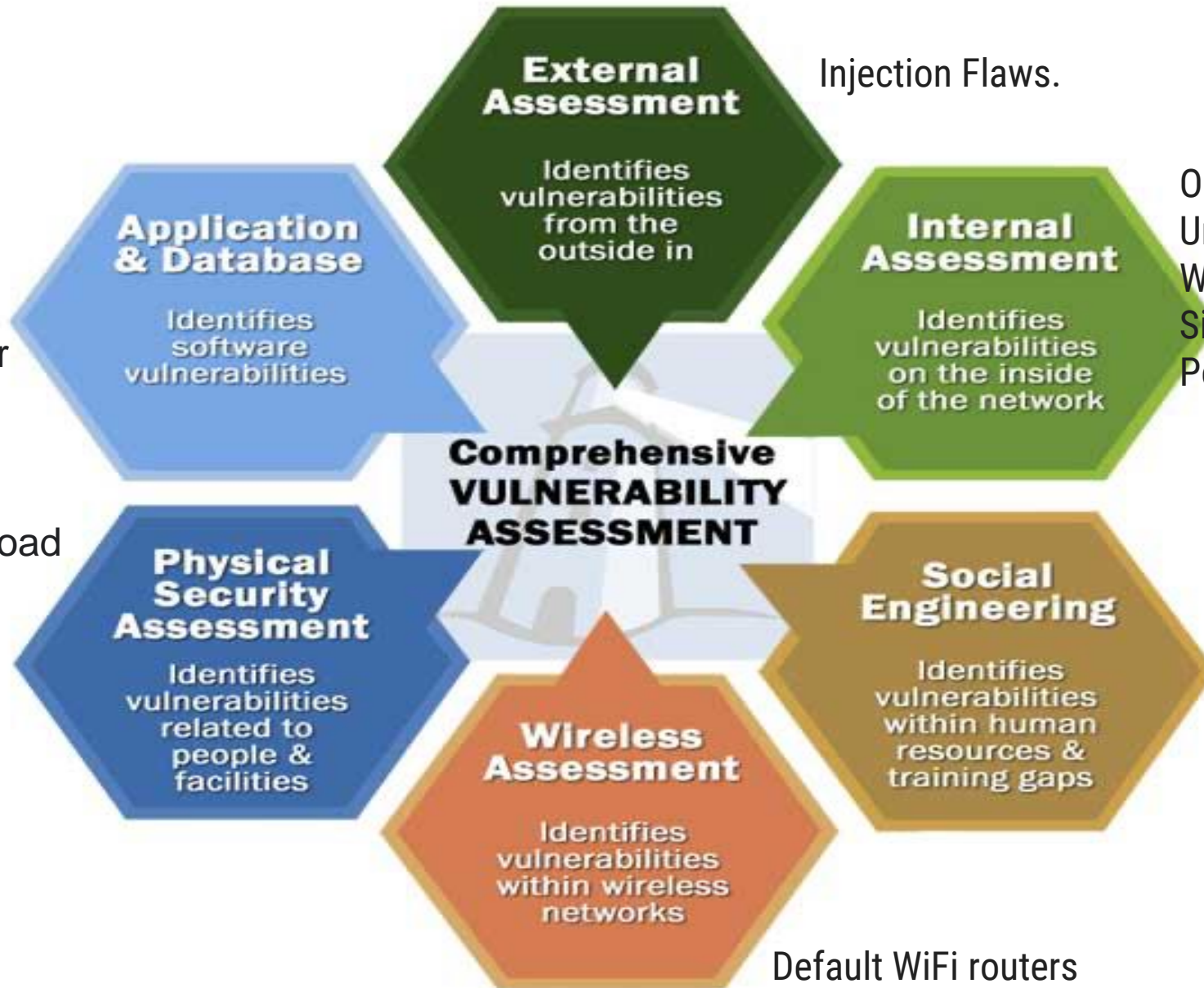
Overview of Vulnerability Scanning – Cont.

Unpatched software means **there are vulnerabilities in a program or code that a company is aware of and will not or cannot fix.** Users can also be responsible for their unpatched software if they refuse to check for and perform regular updates.



Overview of Vulnerability Scanning – Cont.

- Missing data encryption.
- OS command injection.
- SQL injection.
- Buffer overflow.
- Missing authentication for critical function.
- Missing authorization.
- Unrestricted upload of dangerous file types.



Overview of Vulnerability Scanning – Cont.

- A vulnerability assessment is the process of **defining, identifying, classifying and prioritizing** vulnerabilities in computer systems, applications and network infrastructures.

Vulnerability assessments also provide an organization with the necessary knowledge, awareness and risk backgrounds to understand and react to threats to its environment.

A vulnerability assessment process is intended to identify threats and the risks they pose.

They typically involve the use of automated testing tools, such as network security scanners, whose results are listed in a vulnerability assessment report.

Organizations of any size, or even individuals who face an increased risk of cyber attacks, can benefit from some form of vulnerability assessment.

Types of Vulnerability Scanners

- ▶ There are generally two types of vulnerability scanning tools:

1. **Network-based scanning tool:**

- ▶ Network-based scanning tools send **network traffic** to various network hosts and devices.
- ▶ **Network-based scans** are used to identify possible network security attacks. This type of scan can also detect vulnerable systems on wired or wireless networks.
- ▶ It with the goal of gathering information that will indicate whether those systems have holes that can be exploited.
- ▶ Example: OpenVAS, Wireshark, NMAP, Nikto etc.

2. **Host-based scanning tool:**

- ▶ Host-based scanning tools are **run on each host** to scan for a wide range of system problems.
- ▶ It including unauthorized software, unauthorized accounts, unprotected logins, weak passwords and inappropriate access permissions.
- ▶ Example: OSSEC

Types of Vulnerability Scanners

▶ **Cloud-Based Vulnerability Scanners**

- ▶ Used to find vulnerabilities within cloud-based systems such as web applications, WordPress, and Joomla.

▶ **Host-Based Vulnerability Scanners**

- ▶ Used to find vulnerabilities on a single host or system such as an individual computer or a network device like a switch or core-router.

▶ **Network-Based Vulnerability Scanners**

- ▶ Used to find vulnerabilities in an internal network by scanning for open ports. Services running on open ports determined whether vulnerabilities exist or not with the help of the tool.

▶ **Database-Based Vulnerability Scanners**

- ▶ Used to find vulnerabilities in database management systems. Databases are the backbone of any system storing sensitive information. Vulnerability scanning is performed on database systems to prevent attacks like SQL Injection.

False Negative & False Positive

- ▶ The vulnerability scanners use **predefined tests** to identify vulnerabilities (also called **vulns**).
- ▶ If the scanner has insufficient test then the scanner does not report the vulnerability exists on the system.
- ▶ It can be known as **false negative**.
- ▶ Example: person is declared not criminal by court but he is actually a criminal.
- ▶ If the scanner has a poorly written test then scanner reports vulnerability even if it does not exist on a system. It may produce a **false positive**.
- ▶ It wastes time as administrators must follow up to manually check the vulnerability that is actually vulnerable or not.
- ▶ Some of the free and very useful vulnerability scanners are:
 - Netcat
 - Socat
- ▶ False Positives occur when a scanner, Web Application Firewall (WAF), or Intrusion Prevention System (IPS) flags a security vulnerability that you do not have. A false negative is the opposite of a false positive, telling you that you don't have a vulnerability when, in fact, you do.

Zero-day Vulnerability

- ▶ Zero-day vulnerability refers to a **hole in software** that is unknown to the vendor.
- ▶ This security hole is then exploited by hackers before the vendor becomes aware and hurries to fix it- this exploit is called a **zero day attack**.
- ▶ A zero-day vulnerability is **a vulnerability in a system or device that has been disclosed but is not yet patched**. An exploit that attacks a zero-day vulnerability is called a zero-day exploit.
- ▶ Zero-day vulnerabilities are particularly dangerous because they represent a gap in knowledge between the attacker and defender.

Open Port / Service Identification

- ▶ Some services are very insecure. Telnet (port 23) is famous for its **lack of encryption** that leaks passwords.
- ▶ Hence **Secure Shell** (SSH) is widely accepted and reduced the presence of telnet on the Internet.
- ▶ Services do not always run on default ports, hence the scanner must rely on banners to produce a response from a listening port.
- ▶ A banner is a text displayed by a host server containing details like software type and version running in a system or server.
- ▶ Example of banner grabbing:
 - ▶ HTTP/1.1 200 OK
 - ▶ **Server: nginx/1.16.1**
 - ▶ **Date: Mon, 11 Nov 2019 13:29:13 GMT**
 - ▶ **Content-Type: text/html**
 - ▶ **Content-Length: 5683**
 - ▶ **Last-Modified: Thu, 04 Oct 2018 17:44:00**

Open Port / Service Identification

- ▶ Services do not always declare themselves. Telnet and SMTP (port 25) services return text-based banners when receives request for connection. It does not wait for particular incoming data on that connection.
- ▶ HTTP (port 80) will not respond for connection until the service receives a request that contains data.
- ▶ This way, scanners may distinguish whether an HTTP or SMTP service is listening on non-standard port.

Telnet vs. SSH

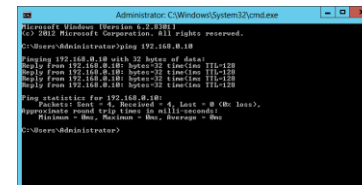
- ▶ Telnet is not secure because it does not provide encryption so our passwords can be leaked while SSH provides the encryption that's why it is more secure.
- ▶ Use Wireshark to capture password while using Telnet and SSH.
- ▶ First download Wireshark and PuTTY.
- ▶ PuTTY: <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
- ▶ Using PuTTY connect to the remote system and check entered username and password using Wireshark.
- ▶ <https://www.youtube.com/watch?v=twJCJ5Rq1o8>

Banner / Version Check

- ▶ Some services declare information about themselves without receiving particular data from a client.
- ▶ Banner Grabbing:
 - ↳ **Banner grabbing** is a technique used to gain information about a computer system on a network and the services running on its open ports.
 - ↳ Administrators can use this to take inventory of the systems and services on their network.
 - ↳ Tools commonly used to perform **banner grabbing** are Telnet, nmap, zmap and Netcat.
- ▶ Example:
 - ↳ SSH command
 - ↳ The ssh command **provides a secure encrypted connection between two hosts over an insecure network.** This connection can also be used for terminal access, file transfers.
- ▶ If you know the version of SSH and target operating system then it is very easy for someone to compromise the host.
- ▶ System administrators usually remove or change banners to make them more secure, but this doesn't remove the vulnerability.
- ▶ For banner grabbing use command: `nc -v IP PORT` or `ncat -v IP PORT`

Probe

- ▶ In Computer Security, a probe is an **attempt to gain access** to a computer and its files through a **known or probable weak point**.
- ▶ A probe is an action taken or an object used for the purpose of learning or collecting data about the state of the network.
- ▶ For example, an empty message can be sent simply to see whether the destination actually exists. Ping is a common utility for sending such a probe.
- ▶ It also could be a program or other device inserted at key junction in a network for the purpose of monitoring or collecting data about network activity -- network tap.
- ▶ Note: A network tap is an external monitoring device that mirrors the traffic that passes between two network nodes. A tap (test access point) is a hardware device inserted at a specific point in the network to monitor data.
- ▶ A network tap usually has four ports. The first two ports connect to the two network nodes at either end of the wire that the tap is monitoring. The additional ports connect to the monitoring devices that receive the mirrored packet flows.



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.0.6002]
(c) 2012 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ping 192.168.0.10

Pinging 192.168.0.10 with 32 bytes of data:
Reply from 192.168.0.10: bytes=32 time=1ms TTL=128
Reply from 192.168.0.10: bytes=32 time=1ms TTL=128
Reply from 192.168.0.10: bytes=32 time=1ms TTL=128
Reply from 192.168.0.10: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.0.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milliseconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\Administrator>
```

Two Type of Probe

1. Traffic Probe
2. Vulnerability Probe

Traffic Probe

- ▶ Traffic probe refers to the kind of information gathering. We are probing the network to gather some information.
- ▶ Some services declare information about themselves without receiving particular data from a client.
- ▶ But all services do not do that. However, lots of them will if you just ask.
- ▶ For example, a web service will not give response until it receives data from the client.
- ▶ A valid **HTTP request using the HEAD method** will provide some useful information like web server information, information about installed server operating system etc. which can be useful to compromise the host.
- ▶ The HTTP HEAD method is one of the commonly used Hypertext Transfer Protocol (HTTP) request methods. It is used to retrieve HTTP headers from the server.
- ▶ HEAD is a request method supported by HTTP used by the World Wide Web. The HEAD method asks for a response identical to that of a GET request, but without the response body. This is useful for **retrieving meta-information written in response headers, without having to transport the entire content.**
- ▶ Some other tools are also used to monitor the network like netcat, socat

Traffic Probe

Consider the example of valid HTTP request with HEAD method. To get the home page of Google:

```
echo "GET / HTTP/1.0\r\nHost:www.google.com\r\n\r\n"|nc google.com 80
```

```
HEAD / HTTP/1.1
```

```
HTTP/1.1 200 OK
```

```
MIME-Version: 1.0
```

```
Server: Apache/2.0.46 (Unix) (Red Hat/Linux)
```

```
Content-Length: 0
```

```
Cache-Control: public
```

```
Expires: Sat, 18 Jul 2020 19:00:00 GMT
```

- So here we get all the information about server.

Vulnerability Probe

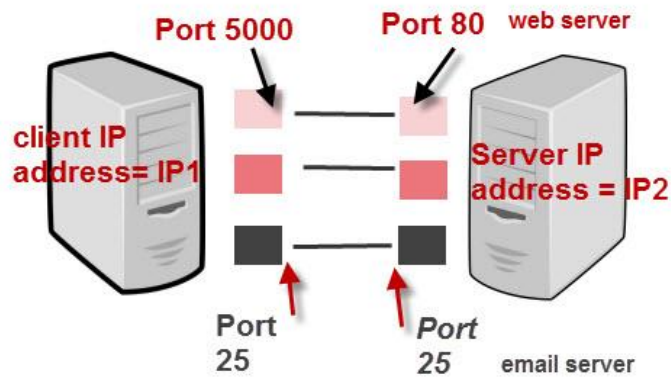
- ▶ A vulnerability probe **uses scanning technology to clean your organization's network for indicators of potential breach risk**
- ▶ Some security bugs cannot be identified without sending a payload that exploits a suspected vulnerability.
- ▶ These types of probes are more accurate—they rely on direct observation not only on port numbers or service banners.
- ▶ But they also carry more risk of interrupting the service, because the test payload must be trying to either produce or take advantage of an error in the service's code.
- ▶ An easy-to-understand example of a vulnerability probe is an HTML injection check for a web application.
- ▶ This type of injection to be performed is through the website's link. Suppose, we have PHP website's link.
- ▶ `https://www.testing123.com/books/lists.php?site=1`
- ▶ As we see, "site" is a parameter and "1" is its value. Then if for the parameter "site" instead of value "1" we would indicate any HTML code with the text to display, this indicated text would be displayed in the "Page Not Found" page. This happens only if the page is vulnerable to HTML

Vulnerability Probe

- ▶ The outcome may be to crash the software, causing a denial of service, or retrieve data, like pulling usernames and passwords from a database, or completely compromise the operating system by gaining root or administrator access.
- ▶ Exploits take many shapes. It can be simple binary shellcode or clever bits of text appended to URL parameters.
- ▶ Discovering vulnerability typically just means uncovering a software fault.
- ▶ Developing an exploit means taking advantage of that software fault to give the attacker an advantage against the system.
- ▶ We can use firewall to provide security of our internal network.
- ▶ Firewall is a device used to control the flow of traffic into and out of the network. It is security device installed between two networks, internal network to outside network (mostly internet).
- ▶ Based on the rules defined into firewall it secures our internal network.

TCP/IP Ports and Sockets

- ▶ On a TCP/IP network every device must have an IP address.
- ▶ The **IP address identifies** the **device** e.g. computer.
- ▶ However an IP address alone is **not sufficient for running network applications**, as a computer can run multiple applications and/or services.
- ▶ Just as the IP address identifies the computer, The **network port identifies** the **application** or **service** running on the computer.
- ▶ The diagram below shows a computer to computer connection and identifies the IP addresses and ports.



IP Address + Port number = Socket

- ▶ A **socket** is the **combination** of **IP address + port**
- ▶ A **connection between** two **computers uses** a **socket**.

Port Number Ranges and Well Known Ports

- ▶ A port number uses **16 bits** and so can therefore have a value from **0 to 65535 decimal**.
- ▶ Port numbers are divided into ranges as follows:
 - ↳ **Port numbers 0-1023 – Well known ports.**
 - The well known port numbers are assigned to server services by IANA which is the Internet Assigned Numbers Authority. IANA is the same group that manages the DNS Root and IP addresses.
 - **“Well-Known” ports** are port numbers that have been reserved for common applications, typically server applications.
 - The port numbers assigned to these server applications have to be known by the client’s Transport layer, so they can add the correct destination port number to messages.
 - Clients know that servers will be listening for their requests at these reserved port numbers.
 - e.g **Web servers** normally use **port 80** and **SMTP servers** use **port 25**.

Port Number Ranges and Well Known Ports

➔ Ports 1024-49151- Registered Port

- These can be registered for services with the IANA and should be treated as **semi-reserved**.
- Registered port numbers are non-well-known ports that are used by vendors for their own server applications. After all, not every possible application capability will be reflected in a well-known port, and software vendors should be free to innovate.
- **User written programs should not use these** ports.
- Examples of applications with registered port numbers include Sun's NEO Object Request Broker (port numbers 1047 and 1048) and Shockwave (port number 1626).
- An Object Request Broker (ORB) **manages the interaction between clients and servers, using the Internet InterORB Protocol (IIOP)**. It enables clients to make requests and receive responses from servers in a network-distributed environment.

Port Number Ranges and Well Known Ports

➔ Ports 49152-65535

- These are **used by client programs not by the server** and you are **free to use** these in client programs.
- When a Web browser connects to a web server the browser will allocate itself a port in this range.
- Client side port numbers are generated and assigned by the Transport layer.
- These port numbers are typically allocated for short term use and are referred to as **“Ephemeral or Dynamic Ports”**.
- They are also known as private or non-reserved ports.

Common Well Known Port Numbers

Number	Assignment
20	File Transfer Protocol (FTP) Data Transfer
21	File Transfer Protocol (FTP) Command Control
22	Secure Shell (SSH) Secure Login
23	Telnet remote login service, unencrypted text messages
25	Simple Mail Transfer Protocol (SMTP) E-mail routing
53	Domain Name System (DNS) service
67, 68	Dynamic Host Configuration Protocol (DHCP)
80	Hypertext Transfer Protocol (HTTP) used in the World Wide Web
110	Post Office Protocol (POP3)
119	Network News Transfer Protocol (NNTP)
123	Network Time Protocol (NTP)
143	Internet Message Access Protocol (IMAP) Management of digital mail
161	Simple Network Management Protocol (SNMP)
194	Internet Relay Chat (IRC)
443	HTTP Secure (HTTPS) HTTP over TLS/SSL

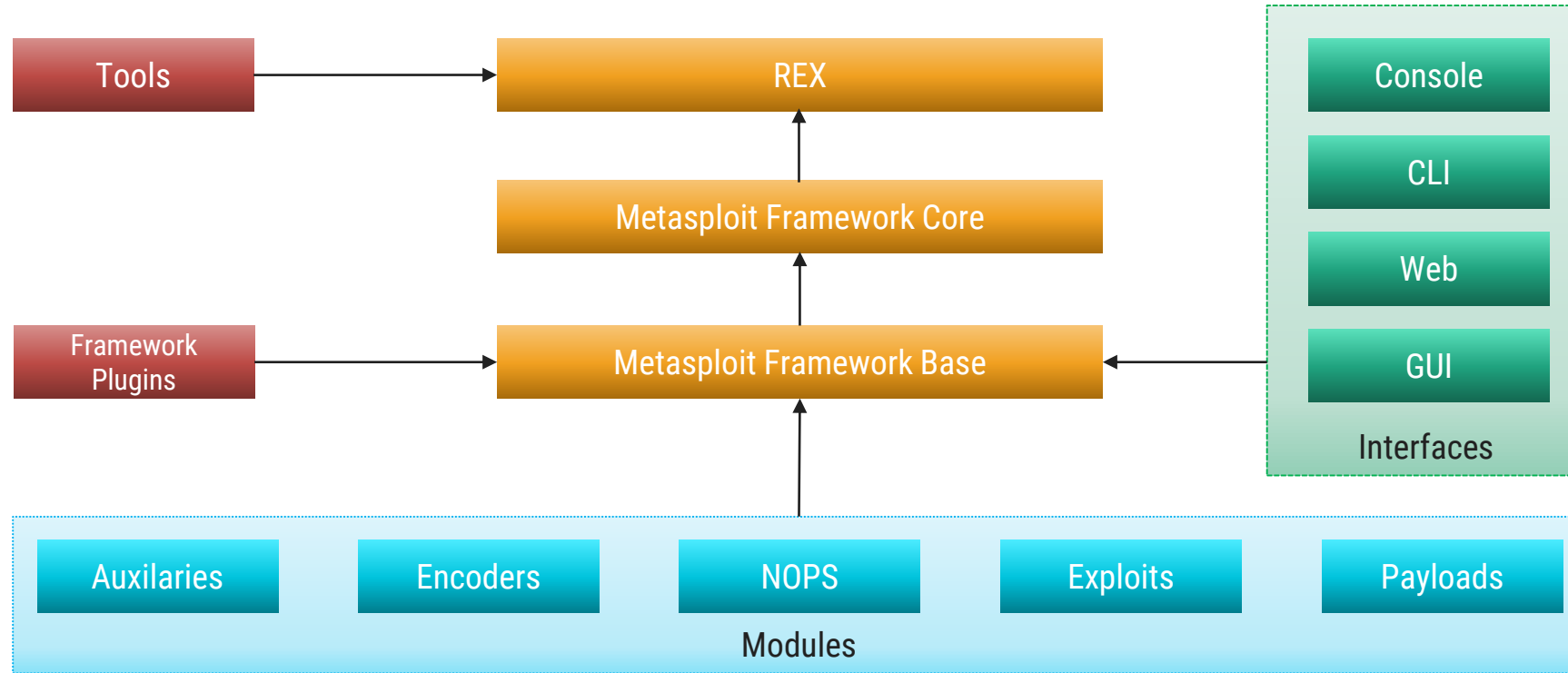
Note: the FTP command channel is used for transmitting commands as well as replies to those commands, while the FTP data channel is used for transferring data.

Port Scanning

- ▶ **Port scanner**: Software designed to probe server or host for Open ports.
- ▶ Used by administrator to verify security policy.
- ▶ Used by attacker to identify running services on host.
- ▶ **Port scan**: A process that sends a client request to server for finding active ports.
- ▶ **Open port**: Host sends a reply indicating port is active. Open means that an application on the target machine is listening for connections/packets on that port.
- ▶ **Close port**: Host sends a reply that connection will be denied.
- ▶ **Filtered**: There was no reply from the host. Filtered means that a firewall, filter, or other network obstacle is blocking the port so that Port scanner cannot tell whether it is open or closed.
- ▶ Vulnerability can be with **open ports**.

Metasploit

- ▶ Metasploit is an open-source framework used for security development and testing.
- ▶ It is best tool for developing and executing exploit code against a remote target machine.



- ▶ Modules built on top of libraries, accessed via interfaces to conduct exploitation tasks. Plugins hook directly into the framework to add commands to the interface, etc.

Metasploit

- ▶ **Vulnerability:** It is a weakness in a computer system that could be exploited by an attacker to perform unauthorized malicious actions. It can be as simple as weak or no password and as complex as a Cross-Site Scripting or buffer overflows.
- ▶ **Exploit:** An exploit is a piece of code that takes advantage of a vulnerability that is present in a computer system to cause unintended behaviour on a computer system like gaining unauthorized access to a network or getting the privilege escalated.
- ▶ **Payload:** A payload is like an engine that defines to perform specific functions for the exploit which took place. It could be installing malware such as worms or viruses which performs the malicious actions or gaining the reverse shell to the compromised system.

Modules of Metasploit Framework

- ▶ Metasploit can be used in most of the penetration testing steps. The core functionalities that Metasploit provides can be summarized by some of the modules:
- ▶ Exploits
- ▶ Payloads
- ▶ Auxiliaries
- ▶ Encoders
- ▶ **1. Exploits**
- ▶ Exploit is the program that is used to attack the vulnerabilities of the target. There is a large database for exploits on Metasploit Framework. You can search the database for the exploits and see the information about how they work, the time they were discovered, how effective they are, and so on.

Modules of Metasploit Framework

▶ 2. Payloads

- ▶ Payloads perform some tasks after the exploit runs. There are different types of payloads that you can use. For example, you could use the reverse shell payload, which basically generates a **shell/terminal/cmd** in the victim machine and connects back to the attacking machine.
- ▶ Another example of a payload would be the bind shell. This type of shell creates a listening port on the victim machine, to which the attacker machine then connects. The advantage of a reverse shell over the bind shell is that the majority of the system firewalls generally do not block the outgoing connections as much as they block the incoming ones.
- ▶ Metasploit Framework has a lot of options for payloads. Some of the most used ones are the **reverse shell**, **bind shell**, **meterpreter**, etc.

▶ 3. Auxiliaries

- ▶ These are the programs that do not directly exploit a system. Rather they are built for providing custom functionalities in Metasploit. Some auxiliaries are sniffers, port scanners, etc. These may help you scan the victim machine for information gathering purposes. For example, if you see a victim machine is running **ssh** service, but you could not find out what version of **ssh** it is using – you could scan the port and get the version of **ssh** using auxiliary modules.

Modules of Metasploit Framework

▶ 4. Encoders

- ▶ Metasploit also provides you with the option to use encoders that will encrypt the codes in such a way that it becomes unclear for the threat detection programs to interpret. They will self decrypt and become original codes when executed. However, the encoders are limited and the anti-virus has many signatures of them already in their databases. So, simply using an encoder will not guarantee anti-virus avoidance. You might get past some of the anti-viruses simply using encoders though. You will have to get creative and experiment changing the payload so it does not get detected.

Components of Metasploit Framework

▶ Metasploit is open-source and it is written in Ruby. It is an extensible framework, and you can build custom features of your likings using Ruby. You can also add different plugins. At the core of the Metasploit framework, there are some key components:

▶ msfconsole

▶ msfdb

▶ msfvenom

▶ meterpreter

▶ **1. msfconsole**

▶ This is the command line interface that is used by the Metasploit Framework. It enables you to navigate through all the Metasploit databases at ease and use the required modules. This is the command that you entered before to get the Metasploit console.

▶ **2. msfdb**

▶ Managing all the data can become a hurdle real quick, which is why Metasploit Framework gives you the option to use PostgreSQL database to store and access your data quickly and efficiently. For example, you may store and organize your scan results in the database to

Components of Metasploit Framework

▶ 3. msfvenom

- ▶ This is the tool that mimics its name and helps you create your own payloads (venoms to inject in your victim machine). This is important since your payload might get detected as a threat and get deleted by threat detection software such as anti-viruses or anti-malware.
- ▶ This happens because the threat detection systems already has stored fingerprints of many malicious payloads. There are some ways you can avoid detection.

▶ 4. meterpreter

- ▶ Meterpreter is an advanced payload that has a lot of functionalities built into it. It communicates using encrypted packets. Furthermore, **meterpreter** is quite difficult to trace and locate once in the system. It can capture screenshots, dump password hashes, and many more.

Metasploit – Cont.

- ▶ Using the built-in tools available in Metasploit, security professionals can conduct penetration tests, verify patch installations and even perform testing.
- ▶ Source code of Metasploit is in ruby.
- ▶ The tool has about 500 modules, including hundreds of remote exploits that can be useful for various releases of Windows, Linux, UNIX, and the Mac OS.
- ▶ Metasploit is very easy to use even a person who can drive a mouse or a keyboard can take over a vulnerable system.
- ▶ It uses PostgreSQL database to manage data for scans, sessions, and post-hack information.

Metasploit Hacking Session Steps

- ▶ A Metasploit hacking session progresses through several steps:
- ▶ First, you must have to identify target.
- ▶ Next, Choose an exploit to use against a vuln on the target.
- ▶ Customize the exploit to the target, which usually just requires specifying the IP address against which to run the exploit.
- ▶ Next, select a payload. Like the exploit, usually just requires specifying an IP address.
- ▶ Finally, launch the customized exploit and await the successful compromise of the target.

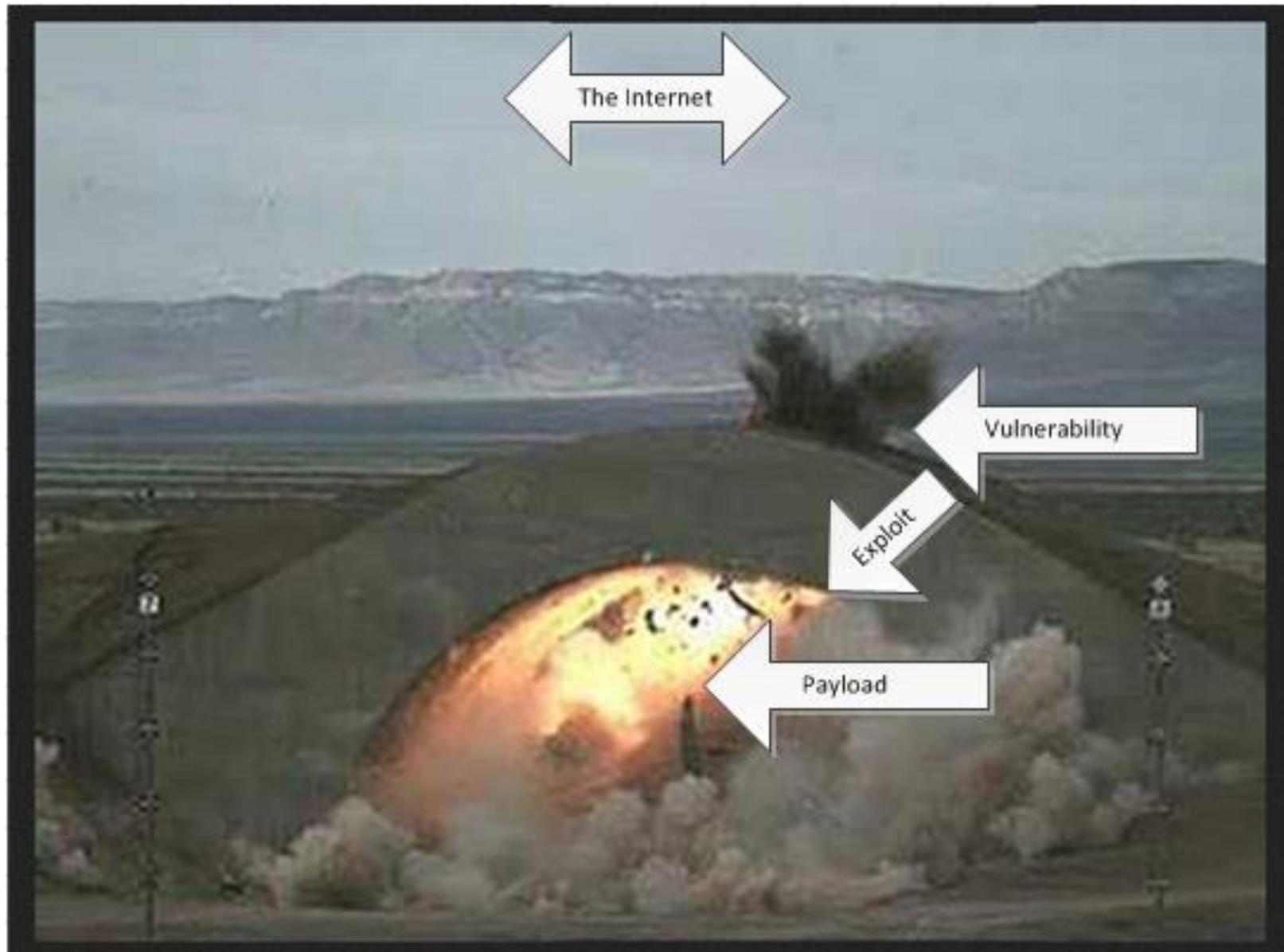
Difference between Payload and Exploits

- ▶ A **payload** refers to the part of malware which performs a malicious action.
- ▶ In the analysis of malicious software such as worms, viruses and Trojans, it refers to the software's harmful results.
- ▶ Examples of payloads include data destruction, messages with insulting text or spam e-mail messages sent to a large number of people.
- ▶ An **exploit** (meaning "using something to one's own advantage") is a piece of software, a chunk of data, or a sequence of commands that takes advantage of a bug or vulnerability in order to cause unexpected behaviour to occur on computer software, hardware, or something electronic.
- ▶ Such behaviour includes things like gaining control of a computer system or a denial-of-service attack.
- ▶ The exploit is what delivers the payload.

Example: Payload and Exploits

- ▶ Take a missile as an analogy. You have the rocket and fuel and everything else in the rocket, and then you have the warhead that does the actual damage.
- ▶ Without the warhead, the missile doesn't do very much when it hits.
- ▶ Additionally, a warhead isn't much use if it goes off in your bunker without a rocket delivering it.
- ▶ The delivery system (missile) is the **exploit** and the **payload** (warhead) is the code that actually does something.
- ▶ **Exploits** give you the ability to 'pop a shell/run your **payload** code'.
- ▶ Example payloads are things like keyloggers, reverse shells etc.
- ▶ Payloads are only referred to when code execution is possible.

Example: Payload and Exploits



Network Vulnerability Scanning - Netcat

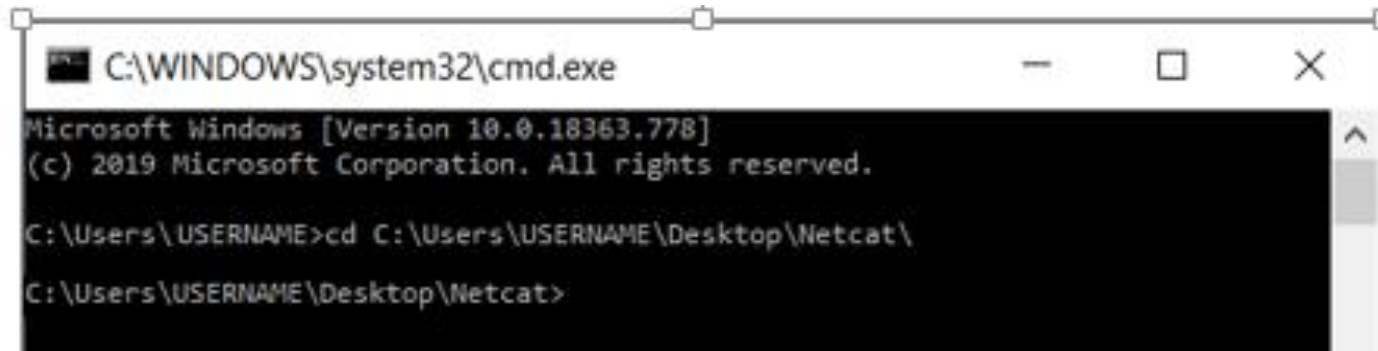
- ▶ Vulnerability scanning is **the process of identifying security weaknesses and flaws in systems and software running on them**. This is an integral component of a vulnerability management program, which has one goal – to protect the organization from breaches and the exposure of sensitive data.
- ▶ Tools:
- ▶ Download for windows: <https://eternallybored.org/misc/netcat/>
- ▶ Netcat is a **command line tool** responsible for reading and writing data in the network.
- ▶ To exchange data, Netcat uses the network protocols TCP/IP and UDP.
- ▶ The tool originally comes from the world of Unix but is now **available for all platforms**.
- ▶ Due to its universal usability, Netcat is often called the “Swiss army knife for TCP/IP”.
- ▶ For instance, it allows you to diagnose faults and problems that expose the functionality and security of a network.
- ▶ Port scans, data streaming or simple data transfers can also be performed by Netcat.

Uses of Netcat

- ▶ Hackers have come up with hundreds of ways to use Netcat.
- ▶ Some of the uses of Netcat are given here in detail:
 - Obtain Remote Access to a Shell
 - Perform Basic Port Scanning
 - Identify more information about ports
 - Communicate with UDP Services
 - For IP Spoofing
 - Hijack a Service

How do I use Netcat?

- ▶ Netcat can be used on all platforms via the **command line**.
- ▶ The command line tool is usually pre-installed on Linux and macOS. Windows users need to download the program from the internet.
- ▶ Special installation steps are not necessary; downloading the program file (*nc.exe*) is enough for use on Windows.
- ▶ You can then use Netcat with **command prompt** (*cmd.exe*) to carry out various network tasks. Start the command prompt.
- ▶ To start the program file (*nc.exe*), you also need to **switch to the storage location**. If the *nc.exe* is saved in the “netcat” folder on the Windows desktop, the syntax will look like this:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\USERNAME>cd C:\Users\USERNAME\Desktop\Netcat\
C:\Users\USERNAME\Desktop\Netcat>
```

Netcat syntax

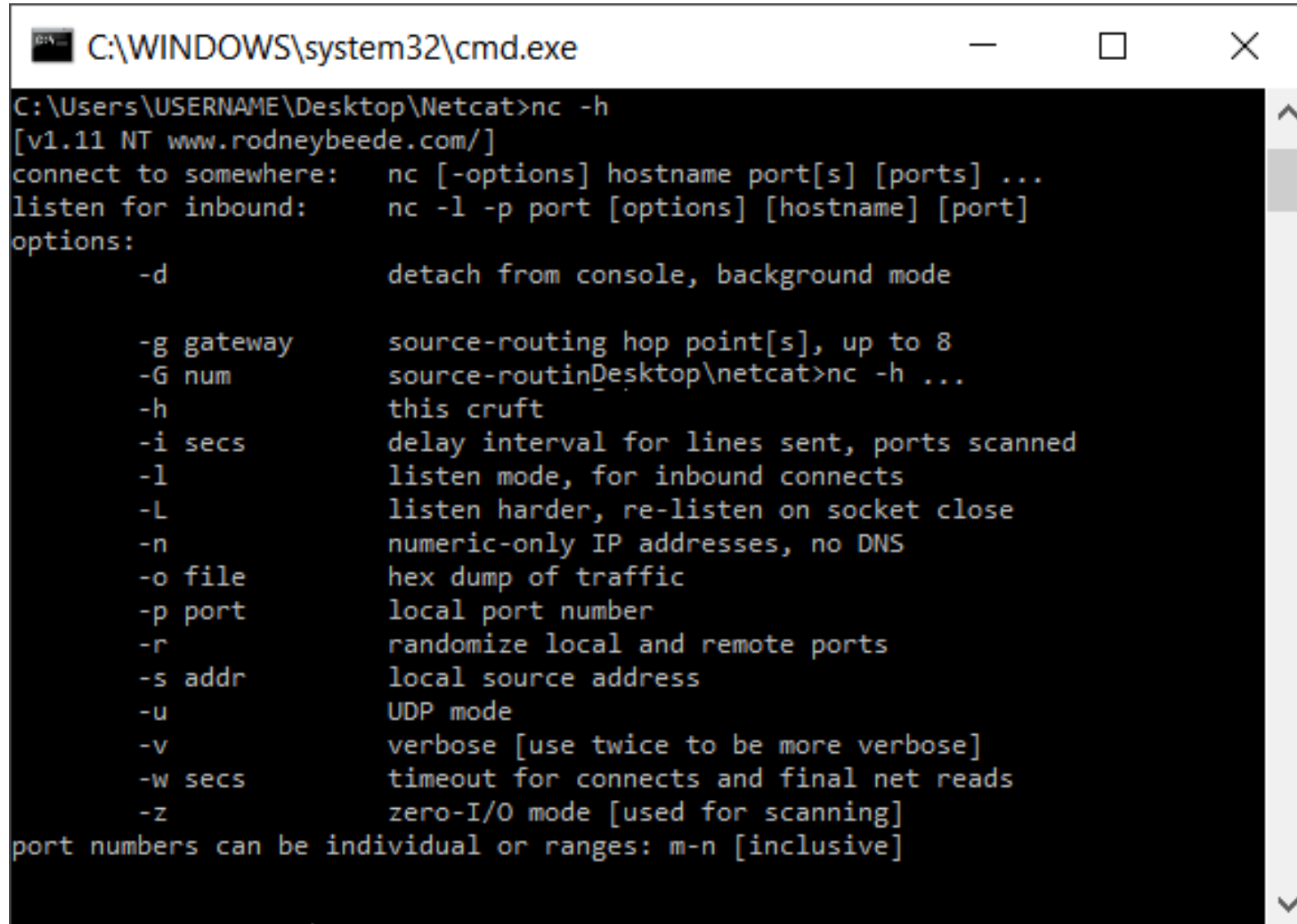
- nc followed by various options as below:

SWITCHES :

-l : LISTEN
-v : VERBOSITY
-p : PORT
-w : TIMEOUT
-n : Don't RESOLVE
-e : EXECUTE COMMANDS
-z : PORT SCAN

Netcat syntax

- nc followed by various options as below:

A screenshot of a Windows command prompt window titled "C:\WINDOWS\system32\cmd.exe". The window shows the output of the command "nc -h" run from the directory "C:\Users\USERNAME\Desktop\Netcat". The output displays the Netcat version "v1.11 NT www.rodneybeede.com/" and provides a list of options for connecting to a remote host or listening for inbound connections. The options listed include -d (detach), -g (gateway), -G (num), -h (help), -i (secs), -l (listen), -L (listen harder), -n (numeric-only), -o (file), -p (port), -r (randomize), -s (addr), -u (UDP), -v (verbose), -w (secs), and -z (zero-I/O). A note at the bottom states that port numbers can be individual or ranges (m-n inclusive).

```
C:\WINDOWS\system32\cmd.exe
C:\Users\USERNAME\Desktop\Netcat>nc -h
[v1.11 NT www.rodneybeede.com/]
connect to somewhere:  nc [-options] hostname port[s] [ports] ...
listen for inbound:   nc -l -p port [options] [hostname] [port]
options:
    -d                detach from console, background mode

    -g gateway        source-routing hop point[s], up to 8
    -G num            source-routinDesktop\netcat>nc -h ...
    -h                this cruft
    -i secs           delay interval for lines sent, ports scanned
    -l                listen mode, for inbound connects
    -L                listen harder, re-listen on socket close
    -n                numeric-only IP addresses, no DNS
    -o file           hex dump of traffic
    -p port           local port number
    -r                randomize local and remote ports
    -s addr           local source address
    -u                UDP mode
    -v                verbose [use twice to be more verbose]
    -w secs           timeout for connects and final net reads
    -z                zero-I/O mode [used for scanning]
port numbers can be individual or ranges: m-n [inclusive]
```

Netcat syntax

- ▶ For instance, if you want to define a **server** or a **client** in the network for data transmission, the following syntax applies:

- ▶ Client mode (connect to somewhere):

```
nc [options] [IP address/host name] [port]
```

- ▶ Server mode (listen for inbound):

```
nc -l -p port [options] [host name] [port]
```

- ▶ The fundamental structure for **running a port scan** is as follows:

```
nc [options] [host] [port]
```

Netcat syntax

- ▶ In order to **detect any errors and security issues**, you can run a scan and identify **open ports**. In the following example, the computer has the IP address *192.168.11.1*. After the IP address, individual ports (e.g. 1), multiple ports (1, 2, 3 etc.) or a whole range (1-1024) can be entered for the scan:

```
C:\Users\hp\Desktop\netcat-1.11>nc -z -v 8.8.8.8 20-80
dns.google [8.8.8.8] 80 (http): TIMEDOUT
dns.google [8.8.8.8] 79 (finger): TIMEDOUT
dns.google [8.8.8.8] 78 (?): TIMEDOUT
dns.google [8.8.8.8] 77 (?): TIMEDOUT
```

```
nc -w 2 -z 192.168.10.1 1-1024
```

Netcat as a simple chat program

- ▶ Netcat can also set up a **simple TCP or UDP connection** between two computers and open up a communication channel.
- ▶ In the example below, the recipient is first installed on the **remote system** and set to listening mode.
- ▶ The recipient then acts as “listener” and uses the port 1605 to receive messages. They can be reached at the IP address *192.168.11.1*.

- ▶ *Run this command in windows*

```
nc -l -p 1605  
ENTER
```

- ▶ A **connection** is then established by the local computer (sending PC) **with the message recipient** using the following command: run this command in kali, ip is of windows os.

```
nc 192.168.11.1 1605  
ENTER
```

- ▶ If the connection is successfully established, messages can be sent in **both directions**.

Netcat as a simple chat program

```
C:\Users\hp\Downloads\netcat\netcat-1.11>nc -vlp 12345
listening on [any] 12345 ...
connect to [192.168.242.131] from DESKTOP-G4MJ83C [192.168.242.131] 21340
hello
good morning
how r u
```

```
C:\Users\hp\Downloads\netcat\netcat-1.11>nc 192.168.242.131 12345
hello
how r u
good morning
```

Socat

- ▶ *Socat* is a flexible, multi-purpose relay tool. **Its purpose is to establish a relationship between two data sources**, where each data source can be a file, a Unix socket, UDP, TCP, or standard input. This tool is regarded as the advanced version of netcat. They do similar things, but **socat has more additional functionality, such as permitting multiple clients to listen on a port, or reusing connections.**
- ▶ **Use Case of socat**
- ▶ *Socat* is useful for **connecting applications inside separate boxes**. Imagine we have Box A and Box B, and inside Box A, there's a database server application running. Furthermore, Box A is closed to the public, but Box B is open. Our network will allow a connection from Box B to Box A.
- ▶ Now, let's say a user wants to read the database log. We don't want the user to enter Box A, but we're fine if the user wants to get inside Box B.
- ▶ *Socat* can connect the database log in Box A with a text reader in Box B. That way, the user can read the log in Box B. We don't have to compromise the security of Box A in order for the user to do the job.
- ▶ *Socat* can work in both directions. The user in Box B might want to send some database queries to the database running in Box A. Then the database running in Box A can send the results back to the user in Box B.

Socat

▶ Installing *socat*

- ▶ *socat* is available in most Linux distros.
- ▶ To install *socat* on Debian-based Linux (such as Ubuntu), we use *apt-get*:
- ▶ `$ apt-get install -y socat`

▶ Connecting Two Stream Sources

- ▶ Let's connect *nc* with the Transmission Control Protocol (TCP) and **stream data from both directions**. We'll need two console terminals to conduct this experiment.
- ▶ On the first terminal, let's run *nc* in listening mode:
- ▶ `$ nc -l localhost 1234`
- ▶ The `-l` flag indicates *nc* is in the listening mode. It listens on *localhost* port *1234*.
- ▶ On the second terminal, let's run *socat* to connect *STDIO* to *localhost* with port *1234* using the TCP protocol:
- ▶ `$ socat STDIO TCP4:localhost:1234`

Socat

- ▶ In the command above, the first argument is the standard input, represented with the keyword *STDIO*. The second argument is a string with a special syntax.
- ▶ As we can see, the string is divided into three parts with a colon delimiter.
- ▶ The first part is the address format, *TCP4*. The second part is the server or the IP address, *localhost*.
- ▶ The last part is the port, *1234*.
- ▶ The *socat* application connects the stream from the first argument (*STDIO*) to the one mentioned in the second argument (*TCP4:localhost:1234*).
- ▶ In this case, we can switch the order of the arguments and it doesn't matter because it's bidirectional.

Socat

▶ Testing the Streams

- ▶ Now, our standard input on the second terminal is connected to the *nc* server. We can type something on it:
- ▶ Good morning!
- ▶ Then, we go back to the first terminal. We can see that **the text we typed on the second terminal was printed on the first terminal**:
- ▶ Good morning!
- ▶ Then, we can type something below *“Good morning!”* on the first terminal:
- ▶ Life is beautiful.
- ▶ Then, we go back to the second terminal. We can see that *“Life is beautiful.”* was printed there as well:
- ▶ **Life is** beautiful.

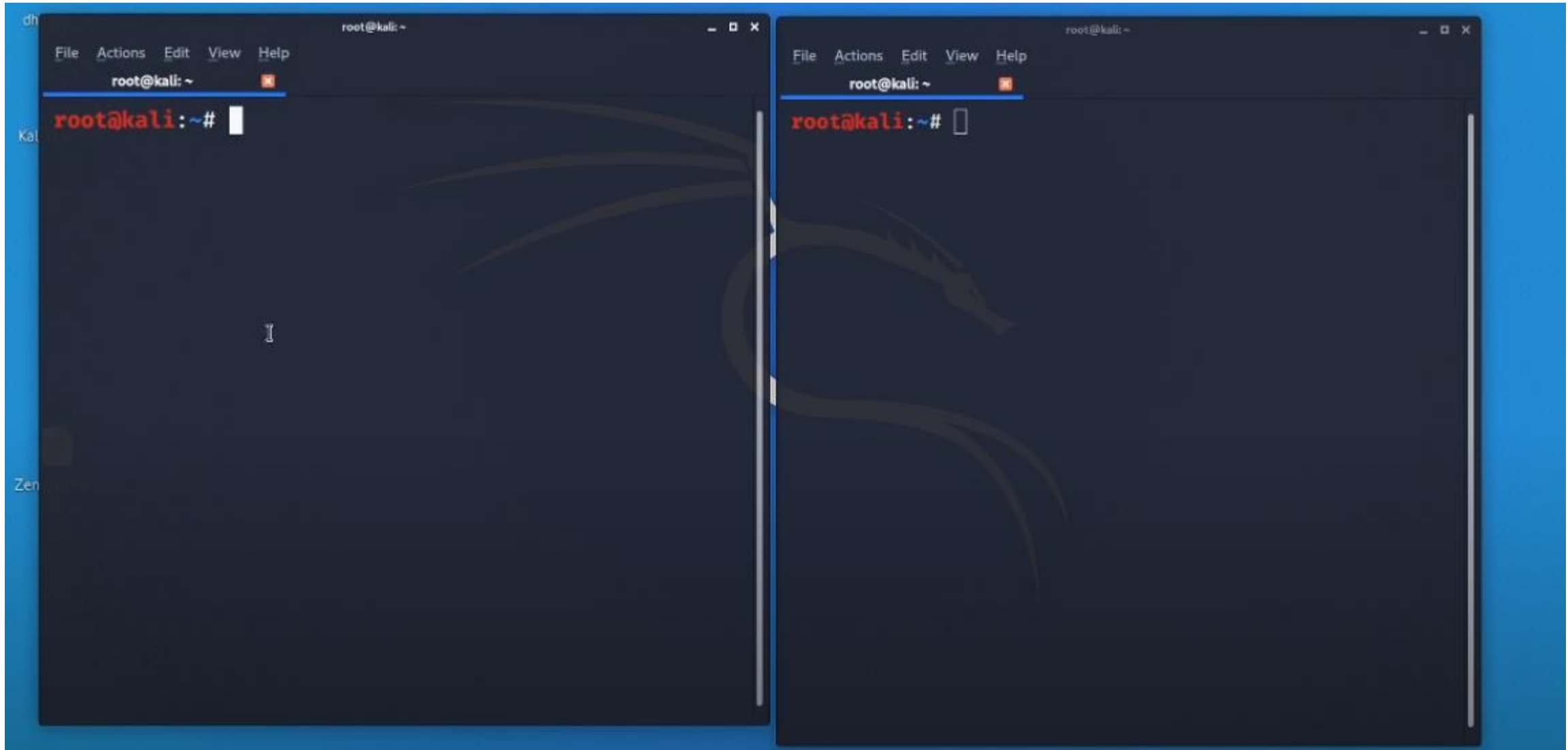
Socat

▶ Forwarding a Stream

- ▶ Instead of handling the data itself, *socat* can **forward the data that will be received by another *socat* application**. This time, we need three console terminals.
- ▶ On the first terminal, we run *nc* as usual:
- ▶ `$ nc -l localhost 1234`
- ▶ On the second terminal, we run *socat*. But this time, we don't connect the standard input with the *nc* application on the first terminal. Instead, we listen on another port:
- ▶ `$ socat TCP4-LISTEN:4321 TCP4:localhost:1234`
- ▶ We used a different string syntax, *TCP4-LISTEN:4321*. It means we listen to the data that comes from the second argument, *TCP4:localhost:1234*, and write it to port *4321*. When we used the *TCP4-LISTEN* address format, we only added the port. This is different from the *TCP4* address format where we have to fill the server address.
- ▶ On the third terminal, we'll run *socat*, and this time, we connect the standard input to the *socat* application on the second terminal:
- ▶ `$ socat STDIO TCP4:localhost:4321`

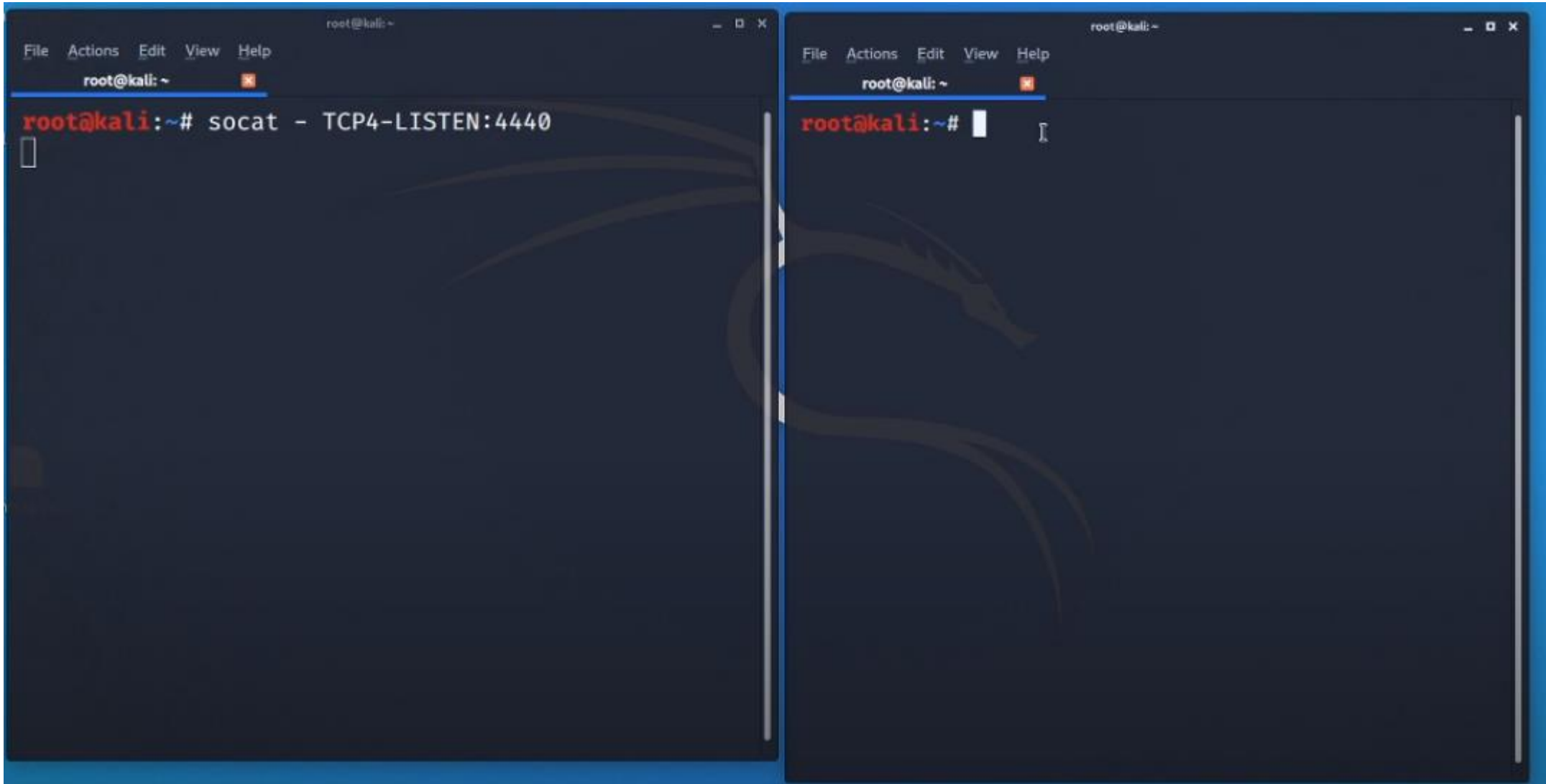
Socat Practical

- ▶ 1. open two terminal in kali linux



Socat Practical

- ▶ 2. on first terminal write following command: it is set to listen mode

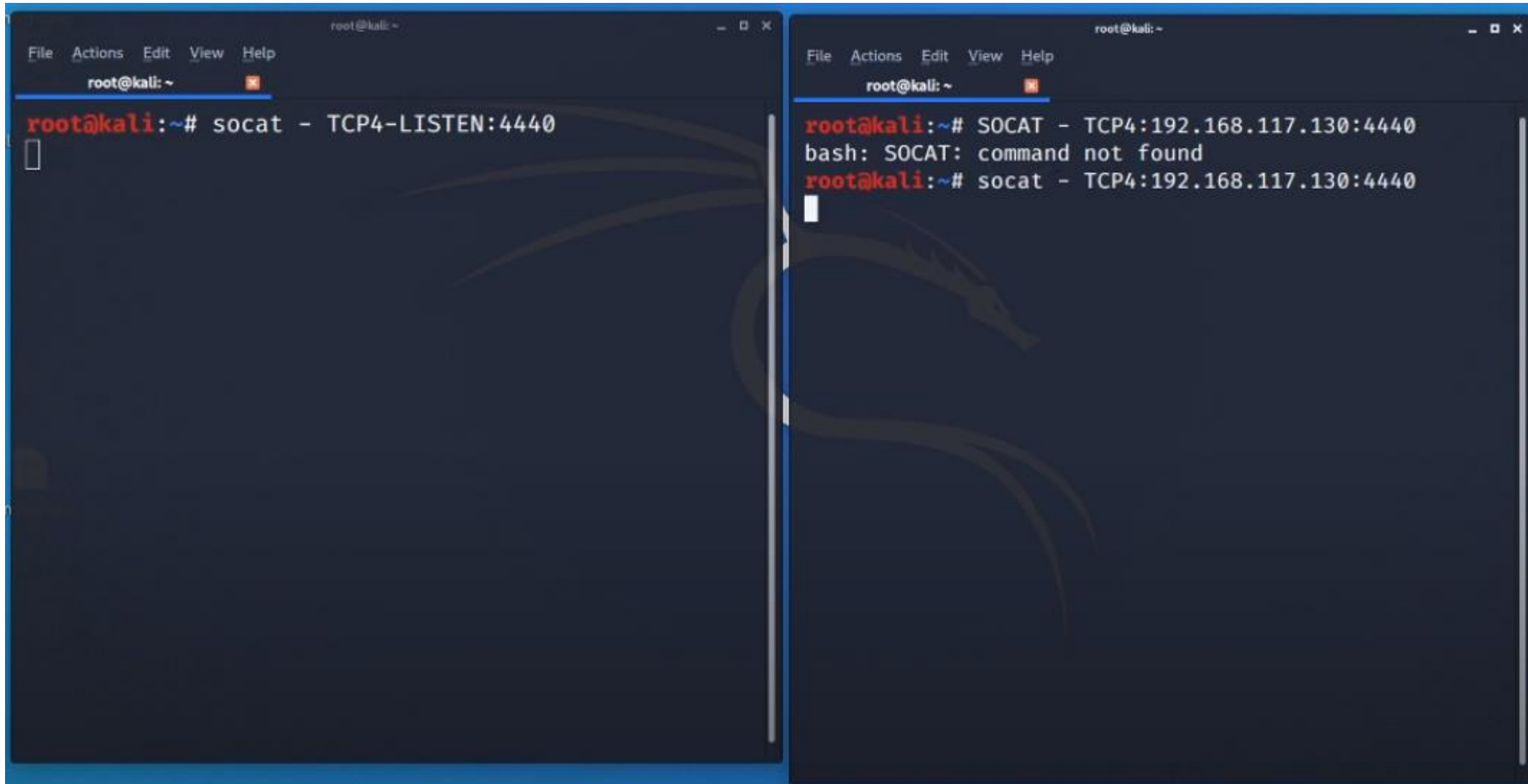


```
root@kali: ~  
File Actions Edit View Help  
root@kali: ~  
root@kali:~# socat - TCP4-LISTEN:4440  
[ ]
```

```
root@kali: ~  
File Actions Edit View Help  
root@kali: ~  
root@kali:~# [ ]
```


Socat Practical

- ▶ 3. on second terminal write following command:

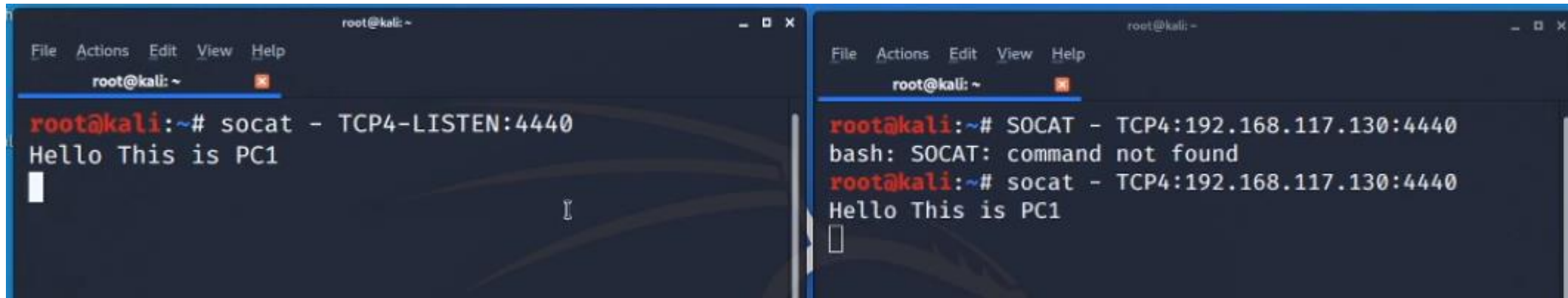


The image shows two terminal windows side-by-side. The left window has a menu bar with 'File', 'Actions', 'Edit', 'View', and 'Help'. Below the menu bar, the prompt is 'root@kali: ~'. The command 'root@kali:~# socat - TCP4-LISTEN:4440' has been entered, and a cursor is visible on the next line. The right window also has a menu bar with 'File', 'Actions', 'Edit', 'View', and 'Help'. Below the menu bar, the prompt is 'root@kali: ~'. The command 'root@kali:~# SOCAT - TCP4:192.168.117.130:4440' has been entered, followed by the error message 'bash: SOCAT: command not found'. The second command 'root@kali:~# socat - TCP4:192.168.117.130:4440' has been entered, and a cursor is visible on the next line.

```
root@kali:~# socat - TCP4-LISTEN:4440
root@kali:~# SOCAT - TCP4:192.168.117.130:4440
bash: SOCAT: command not found
root@kali:~# socat - TCP4:192.168.117.130:4440
```

Socat Practical

- ▶ 4. now both systems are connected here, if type any message in first terminal then then it will be reflected in second terminal.



The image shows two terminal windows side-by-side, both running on a Kali Linux system. The left terminal window has the title 'root@kali: ~' and shows the command 'root@kali:~# socat - TCP4-LISTEN:4440' being executed. Below the command, the text 'Hello This is PC1' is displayed, and a cursor is visible on the line. The right terminal window also has the title 'root@kali: ~' and shows the command 'root@kali:~# SOCAT - TCP4:192.168.117.130:4440'. Below this, the text 'bash: SOCAT: command not found' is shown. Then, the command 'root@kali:~# socat - TCP4:192.168.117.130:4440' is entered, followed by 'Hello This is PC1' and a cursor. This demonstrates a successful connection between the two terminals.

```
root@kali:~# socat - TCP4-LISTEN:4440
Hello This is PC1

```

```
root@kali:~# SOCAT - TCP4:192.168.117.130:4440
bash: SOCAT: command not found
root@kali:~# socat - TCP4:192.168.117.130:4440
Hello This is PC1

```

FPipe

- ▶ Download Fpipe for windows :
- ▶ <https://www.softpedia.com/get/Network-Tools/Misc-Networking-Tools/FPipe.shtml#download>
- ▶ It implement port redirection techniques (it is redirected to you on different port exa: 80->443) natively in windows. It adds UDP protocol and outbound source port number support, which does not in datapipe.
- ▶ FPipe is a TCP source port forwarder/redirector. It can create a TCP / UDP stream with a source port of your choice. This is useful for getting past firewalls that allow traffic with source ports of 23, to connect with internal servers.
- ▶ Fpipe runs on windows operating system. There is no need of priviledge user account and support from dynamic link library.
- ▶ Fpipe can run on local host of the application that you are trying to use to get inside firewall.
- ▶ When you start Fpipe, it will wait for a client to connect on its listening port.
- ▶ It makes a listening connection is made a new connection to the destination machine and port with the specified local source port will be made.
- ▶ When the full connection has been established, Fpipe forwards all the data received on its

Fpipe

fpipe \Rightarrow supports both
TCP and UDP.

\Rightarrow for Windows
Specific

`fpipe.exe` $-l$ 1028 $-r$ 80
listener \swarrow \nearrow remote
www.google.com
redirect

\Rightarrow open browser \Rightarrow 192.168.117.134:1028

FPipe

- ▶ Download Fpipe for windows :
- ▶ <https://www.softpedia.com/get/Network-Tools/Misc-Networking-Tools/FPipe.shtml#download>
- ▶ fpipe2-1.zip file will be downloaded. Unzip it and Copy it to c:\users\ljmca\fpipe2-1
- ▶ Now open command prompt, run following command

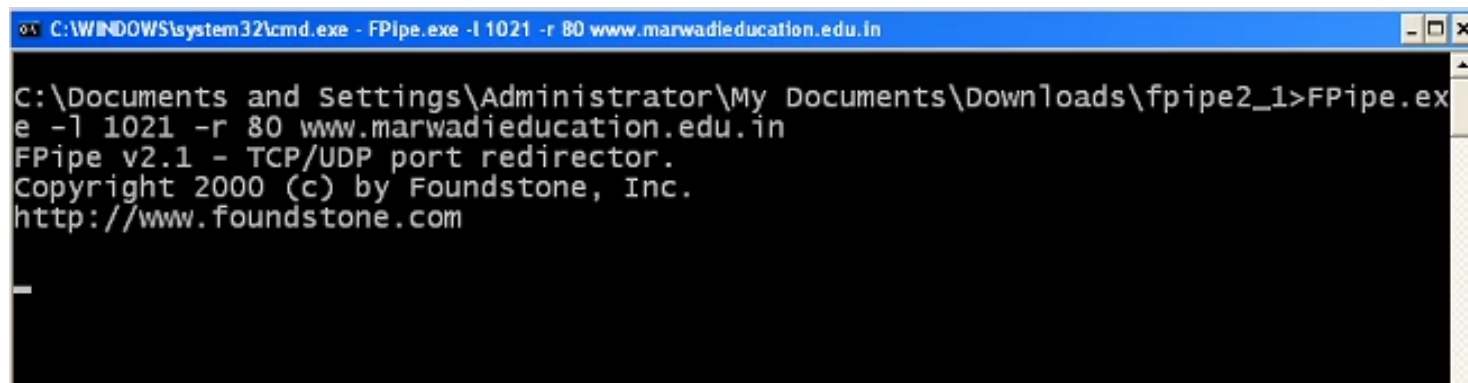
```
C:\Users\hp\Downloads\fpipe2_1>fpipe.exe -l 1234 -r 80 www.ljku.edu.in
FPipe v2.1 - TCP/UDP port redirector.
Copyright 2000 (c) by Foundstone, Inc.
http://www.foundstone.com

Pipe connected:
  In:  192.168.242.131:15278 --> 192.168.242.131:1234
  Out: 192.168.242.131:15280 --> 208.109.9.101:80
```

- ▶ -l for listening to 1028 port number, -r remote and 80 is the remote port and we want to connect to ljku.edu.in
- ▶ So when we open browser and write ip address of our computer in browser like 192.168.20.40:1021, it will open ljku.edu.in website

Fpipe Option

Sr No.	Option	Description
1	-? Or -h	Display Help
2	-c	Max. allows simultaneous TCP connections. Default 32 connections are allowed.
3	-i	Listening interface IP address
4	-l	Listening port number
5	-r	Remote port number
6	-s	Source port used for outbound traffic
7	-u	It support UDP mode
8	-v	For verbose mode



```
C:\WINDOWS\system32\cmd.exe - FPipe.exe -l 1021 -r 80 www.marwadieducation.edu.in

C:\Documents and Settings\Administrator\My Documents\Downloads\fpipe2_1>FPipe.exe -l 1021 -r 80 www.marwadieducation.edu.in
FPipe v2.1 - TCP/UDP port redirector.
Copyright 2000 (c) by Foundstone, Inc.
http://www.foundstone.com

_
```

Datapipe

- ▶ This tool is for unix operating system. And it can specifically handles TCP traffic only.
- ▶ A port redirection tool passes TCP/IP traffic received by the tool on one port to another port to which the tool points.
- ▶ A port redirection tool functions as a channel for TCP/IP connections.
- ▶ For example, you could place a datapipe on a system between a browser and a web server.
- ▶ If you pointed the browser to the listening port of the system with the redirection tool, the browser would see the contents of the web server without having to directly access the web server's IP address.
- ▶ Datapipe is a Unix-based port redirection tool. It runs on the UNIX OS.
 - ➞ `$./datapipe`
 - ➞ `./datapipe localhost localport remotehost remoteport`

Datapipe

Datapipe \Rightarrow { port redirection tool }

NAT}

80 $\xrightarrow{\text{redirecting to a different}}$ 443

proxy will redirect you to google.com
website with your ip-address and
port

`./datapipe` `<ip-address> 8090` { `www.google.com 80` } ✓

Datapipe – Cont.

- ▶ The **localhost** argument indicates the IP address on which to open the listening port.
- ▶ It may be the localhost interface (i.e., 127.0.0.1) or the address of a network interface on the
- ▶ local system from which the **datapipe** command is being executed.
- ▶ The **localport** argument indicates the listening port on the local system; connections will be made to this port number.
- ▶ On UNIX systems, you must have root privileges to open a listening port below 1024.
- ▶ If you receive an error similar to “bind: Permission denied,” your account may not have privileges to open a reserved port.
- ▶ The **remoteport** argument indicates the port to which data is to be forwarded.
- ▶ For example, in most cases if the target is a web server, the remoteport value will be 80.
- ▶ The **remotehost** argument indicates the hostname or IP address of the target.
- ▶ The easiest conceptual example of port redirection is forwarding HTTP traffic.

Datapipe – Cont.

- ▶ Here we set up a datapipe to listen on a high port, 9080 in this example, that redirects to a web site of our choice:
 - ➔ `$./datapipe my.host 9080 80 www.google.com`
- ▶ Now, we enter this URL into a web browser:
 - ➔ `http://my.host:9080/`
 - ➔ You should see Google's home page.
- ▶ Datapipe performs a basic function, but with a little creativity you can make it a powerful tool.
- ▶ Port redirection forwards traffic between TCP ports only.
- ▶ It does not perform protocol conversion or any other data manipulation.
- ▶ Redirecting web traffic from port 80 to port 443 will not change HTTP connections to encrypted HTTPS connections.

Datapipe – Cont.

► First install datapipe

```
root@kali:~# wget https://github.com/bovine/datapipe.git
--2020-09-26 00:00:30-- https://github.com/bovine/datapipe.git
Resolving github.com (github.com)... 13.234.176.102
Connecting to github.com (github.com)|13.234.176.102|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/bovine/datapipe [following]
--2020-09-26 00:00:31-- https://github.com/bovine/datapipe
Reusing existing connection to github.com:443.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'datapipe.git'

datapipe.git           [  <=>  ]
2020-09-26 00:00:32 (404 KB/s) - 'datapipe.git' saved [112736]
```

Datapipe – Cont.

- ▶ Now run this command to create datapipe executable

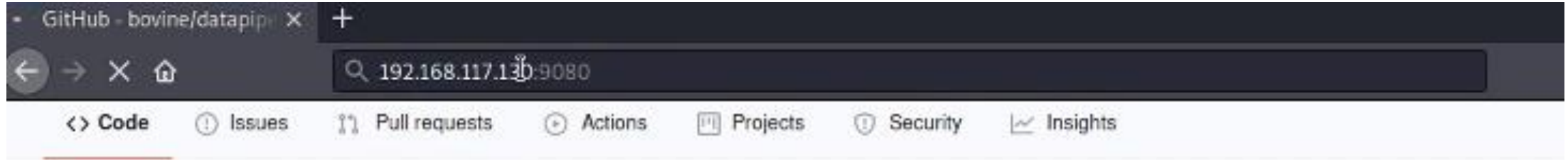
```
root@kali: ~/datapipe
root@kali:~# cd datapipe/
root@kali:~/datapipe# ls
a.out  datapipe  datapipe.c  datapipe.txt  license.txt  README
root@kali:~/datapipe# gcc -o datapipe datapipe.c
```

- ▶ Now run this command

```
root@kali:~/datapipe# ./datapipe 192.168.117.130 9080 www.ignou.ac.in 80
```

Datapipe – Cont.

- ▶ Now open browser and type this ip, it will open ignou.ac.in



Winrelay

- ▶ Winrelay is windows based port redirection tool.
- ▶ The most recent version improves on datapipe and Fpipe by providing support for **IPv6** networks.
- ▶ Winrelay is a TCP/UDP forwarder/redirector that works with both IPv4 and IPv6.
- ▶ You can choose the port and IP it will listen on, the source port and IP that it will connect from, and the port and IP that it will connect to.
- ▶ Some antivirus software consider as malicious software.
- ▶ Online games use datapipe and fpipe tools.
- ▶ Port redirection tools are useful for assigning the alternative port to a service.
- ▶ Source:
 - ↳ www.ntsecurity.nu/toolbox/winrelay/

Network Reconnaissance

- ▶ Reconnaissance attack is a kind of information gathering on network system and services. This enable the attacker to discover vulnerabilities or weaknesses on the network.
- ▶ Network reconnaissance is a testing done for finding potential vulnerabilities in a computer network. It is the process of acquiring information about a network or doing a preliminary survey to gain information.
- ▶ Hackers use reconnaissance as the first step in an effective attack.
- ▶ Hackers find as much information about the target as possible before launching the first attack.
- ▶ Generally, goals of reconnaissance on a target network are to discover:
 - Locate the network and identify IP addresses of hosts.
 - Find out accessible UDP and TCP ports.
 - Identify open ports and underlying applications.
 - Identify OS type in each hosts.
 - Identify active machines.

Network Reconnaissance

- ▶ Nmap and THC-Amap are examples of tools designed to do Network Reconnaissance.
- ▶ Tools are:
 - ↳ AMAP: Application Mapper, uses the results from Nmap to mine for more information.
 - ↳ Nessus: It is vulnerability scanner.
 - ↳ Scanrand: It is fast network scanner.
 - ↳ Paratrace: TCP traceroute that utilizes selected TTL messages.
- ▶ Intruders are increasingly making use of compromised hosts to launch reconnaissance against target networks.

Network Reconnaissance

- ▶ Reconnaissance attack can be active or passive.
- ▶ Passive reconnaissance is an attempt to gain information about targeted computers and networks without actively engaging with the systems. In active reconnaissance, in contrast, the attacker engages with the target system, typically conducting a port scan to determine find any open ports.
- ▶ Methods of passive reconnaissance include: **War driving to detect vulnerable wireless networks**. Looking for information stored on discarded computers and other devices.
- ▶ Wardriving involves **attackers searching for wireless networks with vulnerabilities while moving around an area in a moving vehicle**.
- ▶ The term *reconnaissance* comes from its military use to describe an information-gathering mission.
- ▶ Both types of reconnaissance are sometimes referred to as *passive attacks* because the purpose is simply to obtain information, rather than to actively exploit the target.
- ▶ However, reconnaissance is often a preliminary step towards an active attempt to exploit the target system.

Network Reconnaissance

- ▶ Both active and passive reconnaissance are also used for ethical hacking, in which white hat hackers use attack methods to determine system vulnerabilities so that problems can be taken care of before the system falls prey to a real attack.
- ▶ The simplest way to protect yourself from port scan attacks or reconnaissance attacks is to use a good firewall and intrusion prevention system (IPS).
- ▶ An intrusion prevention system (IPS) is **a network security tool that continuously monitors a network for malicious activity and takes action to prevent it.**
- ▶ Firewall filters traffic based on IP and port number and An intrusion prevention system (IPS) is a network security tool (which can be a hardware device or software) that continuously monitors a network for malicious activity and takes action to prevent it, including reporting, blocking, or dropping it, when it does occur.
- ▶ The firewall controls which ports are exposed and to whom they are visible, while the IPS will detect port scans in progress and shut them down before they are able to gain a full map of your network.

NMAP

- ▶ Nmap (“Network Mapper”) is a free and open source (license) utility for network discovery and security auditing.
- ▶ Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime.
- ▶ Nmap uses raw IP packets in novel ways:
 - To determine what hosts are available on the network.
 - Available services (application name and version) those hosts are offering.
 - Operating systems (and OS versions) they are running.
 - Type of packet filters/firewalls are in use.
- ▶ It was designed to rapidly scan large networks, but works fine against single hosts.
- ▶ Nmap started as a Linux utility and was ported to other systems including Windows, macOS etc.
- ▶ Nmap is used to discover hosts and services on a computer network by sending packets and analyzing the responses.
- ▶

NMAP

- ▶ Nmap runs on all major computer operating systems, and official binary packages are available for Linux, Windows, and Mac OS X.
- ▶ In addition to the classic command-line Nmap executable, the Nmap suite includes:
 - An advanced GUI and results viewer (Zenmap).
 - A flexible data transfer, redirection, and debugging tool (Ncat).
 - A utility for comparing scan results (Ndiff).
 - A packet generation and response analysis tool (Nping).

What Nmap can do?

- ▶ Identify Hosts on the Network, Scan for TCP and UDP Ports
- ▶ Port scanning, Scan for Protocols
- ▶ Identify a Target's Operating System, Scriptable interaction with the target
- ▶ Version detection, Camouflage the Scan (Nmap includes options that hide its scanning process from network security and monitoring devices like firewall.)
- ▶ Nmap can provide further information on targets, device types, and MAC addresses.

NMAP Characteristics and Source

- ▶ Flexible
- ▶ Powerful
- ▶ Portable
- ▶ Easy
- ▶ Free
- ▶ Well Documented
- ▶ Supported
- ▶ Acclaimed
- ▶ Popular

- ▶ Source:
 - ↳ <http://nmap.org/>

NMAP - Identify Hosts on the Network

- ▶ To determine which hosts (i.e., IP addresses) on a network are live, use the Ping scanning method.
- ▶ It sends ICMP (**Internet Control Message Protocol**) echo requests to the specified range of IP addresses and awaits a response. Based on the response, information about the network can be retrieved.
- ▶ Nmap applies the ICMP probing concepts to TCP ports as well.
- ▶ For example, by sending SYN, ACK packets to a TCP port nmap can assume whether a host is live or not based on the response received.
- ▶ If it receives any response then Nmap assumes the host has responded and it is live.
- ▶ If it receives nothing, the host is assumed to not be live, not currently on the network, or ignoring connections to the target port.

NMAP - Scan for TCP Ports

- ▶ The basic method of TCP port scanning is to call a TCP connect function for the port and wait for a response. This is called “TCP connect” because it is based on the Unix system function used for network communications.
- ▶ The connect function conducts the TCP three-way handshake and try to establish a connection.
- ▶ The table given below represents the possible assumptions made by nmap after getting the reply for various requests.



Nmap Sends Packet with TCP Flag	Nmap Receives Packet with TCP Flag	Nmap Sends Follow-up Packet with TCP Flag	Nmap Assumes
SYN	SYN-ACK	ACK followed by RST	Port is open; host is alive.
SYN	RST	–	Port is closed; host is alive.
SYN	No response	–	Port is blocked by firewall or host is not present.
ACK	RST	–	Port is not firewall- protected; port may be open or closed; host is alive.
ACK	No response <i>or</i> ICMP unreachable	–	Port is blocked by firewall or host is not present.
FIN	Nothing		Port is open if host is alive and not firewall-protected.
FIN	RST		Port is closed; host is alive.

NMAP - Scan for TCP Ports

Nmap Sends Packet with TCP Flag	Nmap Receives Packet with TCP Flag	Nmap Sends Follow-up Packet with TCP Flag	Nmap Assumes
SYN	SYN-ACK	ACK followed by RST	Port is open; host is alive.
SYN	RST	–	Port is closed; host is alive.
SYN	No response	–	Port is blocked by firewall or host is not present.
ACK	RST	–	Port is not firewall- protected; port may be open or closed; host is alive.
ACK	No response <i>or</i> ICMP unreachable	–	Port is blocked by firewall or host is not present.
FIN	Nothing		Port is open if host is alive and not firewall-protected.
FIN	RST		Port is closed; host is alive.

NMAP - Scan for UDP Ports

- ▶ Scanning for UDP services is more error-prone than scanning for TCP services because UDP does not support the same statehandling of connection handshakes, resets, re-requests, and so on.
- ▶ **Scan for Protocols**
- ▶ This is used to identify whether a port is supporting a particular type of protocol or not.
- ▶ For example if we make an attempt to connect to a UDP port the following conclusion can be obtained.

Nmap Sends
to Target Port

Empty UDP
packet

Empty UDP
packet

Nmap Receives from
Target Port

Nothing

ICMP port
unreachable

Nmap Assumes

The port is open if the host responds to the Ping (host is alive); however, the port may be closed if the target's network blocks ICMP responses.

The port is closed.

NMAP - Scan for UDP Ports

▶ Identify a Target's Operating System

- ▶ One of Nmap's most useful features is the capability to determine a host's operating system based on its responses to specific packets.
- ▶ Depending on the operating system(OS), Nmap may even provide a particular version and patch level information.
- ▶ Nmap sends a series of TCP and UDP packets to the remote host and examines practically every bit in the responses.
- ▶ After performing dozens of tests Nmap compares the results to its nmap-os-db database of more than 2,600 known OS fingerprints and prints out the OS details if there is a match.
- ▶ Each fingerprint includes a freeform textual description of the OS, and a classification which provides the vendor name (e.g. Sun), underlying OS (e.g. Solaris), OS generation (e.g. 10), and device type (general purpose, router, switch, game console, etc).

NETWORK SNIFFERS AND INJECTION TOOLS

- ▶ Sniffers are effective debugging tools and equally effective hacking tools which can monitor traffic present anywhere on the communication channel.
- ▶ *Network Sniffing* - Process of capturing, decoding, and analyzing network traffic is called Network Sniffing.
- ▶ A network sniffer can listen and record any raw data that passes through it. Network sniffing is a tool that can help us locate network problems by allowing us to capture and view packet level data on our network.
- ▶ Wireless sniffers are also commonly referred to as wireless packet sniffers or wireless network sniffers.
- ▶ Sniffers work differently depending on the type of network they are in.
 - Shared Ethernet
 - Switched Ethernet
- ▶ Sniffers are useful tool for system and network administrators.
- ▶ The sniffer typically operates on the Data Link Layer of the OSI model so it does not have to play by the rules of any higher level protocols.

NETWORK SNIFFERS AND INJECTION TOOLS

- ▶ One way to limit the impact of sniffers is to employ encrypted channels for communicating with services.
- ▶ Examples of sniffers
 - Tcpdump
 - Windump
 - Wireshark
 - Ettercap
 - Hping
 - Kismet

TCPdump

- ▶ TcpDump is primarily a sniffer that runs under the command line.
- ▶ It allows the user to display TCP/IP and other packets being transmitted or received over a network to which the computer is attached.
- ▶ TcpDump is free software and is a highly configurable, commandline packet sniffer for Unix.
- ▶ It works well and is present by default on most Unix-based systems.
- ▶ It's long been a part of the Unix due to its usefulness in debugging networks and services.
- ▶ Tcpdump was made strictly for
 - network monitoring
 - traffic analysis and testing
 - packet inspection
- ▶ It captures a lot of useful low level information about a packets passing on the network, and it can help diagnose all kind of network problems.

TCPdump

- ▶ Tcpdump filters enables us to extract any combination of network packets. But it does not extract detailed information from higher level protocols like HTTP, SNMP, or DNS into more human readable formats.
- ▶ TCPdump uses the libpcap library to capture packets. It can be used for intercepting and displaying the communications of another user or computer.
- ▶ **Installing tcpdump tool in Linux:** Many Operating Systems have tcpdump command pre-installed but to install it,

For Ubuntu/Debian OS

apt install tcpdump

- ▶ Source:
 - ➞ <http://www.tcpdump.org>

TCPdump Commands

- ▶ TCPdump can only be used by root user. It can decode and monitor the header data of
 - Internet protocol (IP)
 - Transmission Control Protocol (TCP)
 - User Datagram Protocol (UDP)
 - Internet Control Message Protocol (ICMP)

Working with tcpdump command : This will capture the packets from the current interface of the network through which the system is connected to the internet.

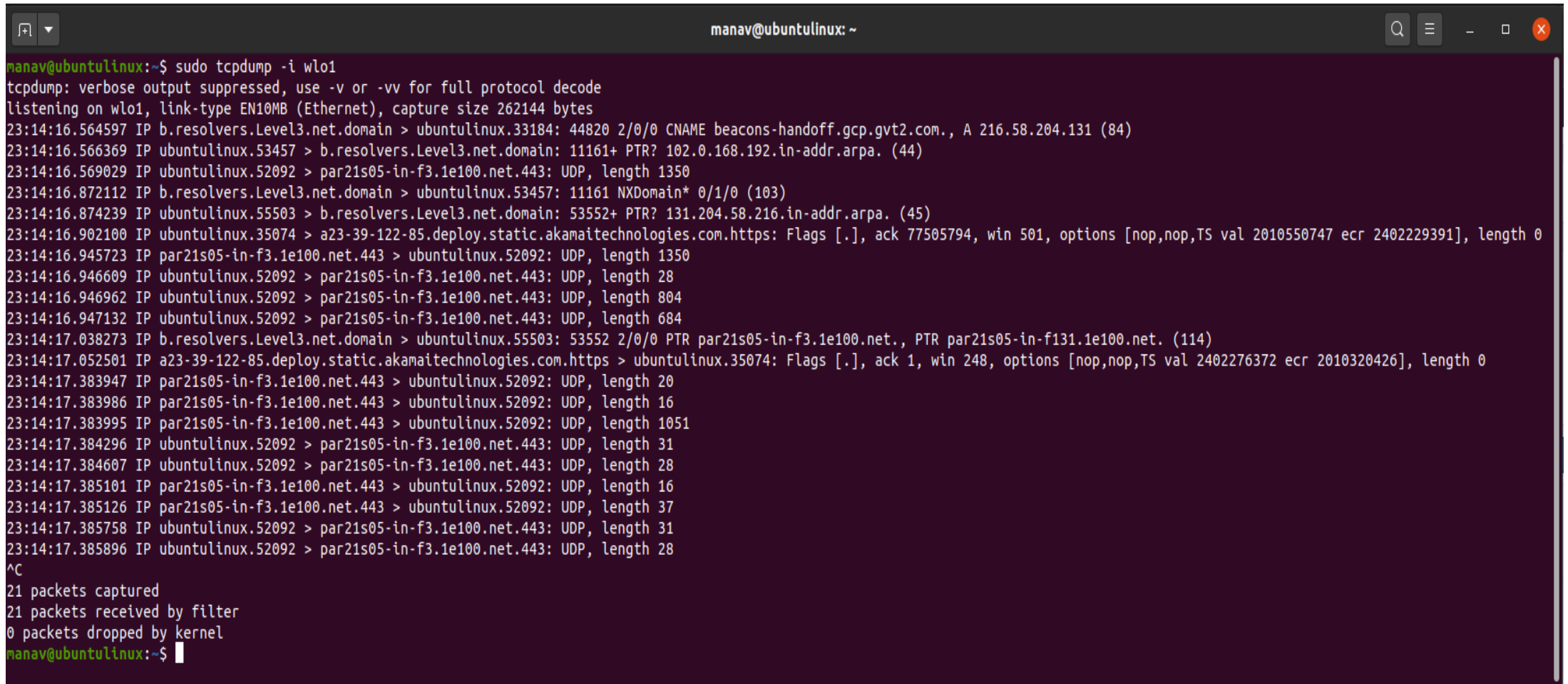
```
sudo tcpdump
```

[illegible]

TCPdump Commands

2. To capture packets from a specific network interface

```
sudo tcpdump -i wlo1
```

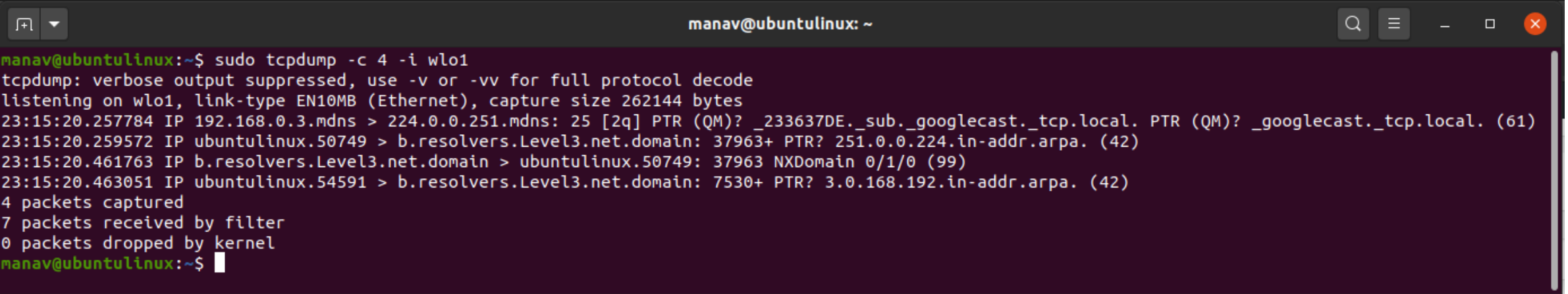
A terminal window titled 'manav@ubuntu: ~' with standard Ubuntu window controls. It shows the execution of 'sudo tcpdump -i wlo1'. The output displays network traffic details including timestamps, IP addresses, protocols, and lengths. It lists several DNS queries and responses, as well as an HTTPS request. The capture ends with a summary: 21 packets captured, 21 received by filter, and 0 dropped by kernel.

```
manav@ubuntu:~$ sudo tcpdump -i wlo1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlo1, link-type EN10MB (Ethernet), capture size 262144 bytes
23:14:16.564597 IP b.resolvers.Level3.net.domain > ubuntu: 44820 2/0/0 CNAME beacons-handoff.gcp.gvt2.com., A 216.58.204.131 (84)
23:14:16.566369 IP ubuntu:53457 > b.resolvers.Level3.net.domain: 11161+ PTR? 102.0.168.192.in-addr.arpa. (44)
23:14:16.569029 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 1350
23:14:16.872112 IP b.resolvers.Level3.net.domain > ubuntu:53457: 11161 NXDomain* 0/1/0 (103)
23:14:16.874239 IP ubuntu:55503 > b.resolvers.Level3.net.domain: 53552+ PTR? 131.204.58.216.in-addr.arpa. (45)
23:14:16.902100 IP ubuntu:35074 > a23-39-122-85.deploy.static.akamaitechnologies.com.https: Flags [.] , ack 77505794, win 501, options [nop,nop,TS val 2010550747 ecr 2402229391], length 0
23:14:16.945723 IP par21s05-in-f3.1e100.net.443 > ubuntu:52092: UDP, length 1350
23:14:16.946609 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 28
23:14:16.946962 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 804
23:14:16.947132 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 684
23:14:17.038273 IP b.resolvers.Level3.net.domain > ubuntu:55503: 53552 2/0/0 PTR par21s05-in-f3.1e100.net., PTR par21s05-in-f131.1e100.net. (114)
23:14:17.052501 IP a23-39-122-85.deploy.static.akamaitechnologies.com.https > ubuntu:35074: Flags [.] , ack 1, win 248, options [nop,nop,TS val 2402276372 ecr 2010320426], length 0
23:14:17.383947 IP par21s05-in-f3.1e100.net.443 > ubuntu:52092: UDP, length 20
23:14:17.383986 IP par21s05-in-f3.1e100.net.443 > ubuntu:52092: UDP, length 16
23:14:17.383995 IP par21s05-in-f3.1e100.net.443 > ubuntu:52092: UDP, length 1051
23:14:17.384296 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 31
23:14:17.384607 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 28
23:14:17.385101 IP par21s05-in-f3.1e100.net.443 > ubuntu:52092: UDP, length 16
23:14:17.385126 IP par21s05-in-f3.1e100.net.443 > ubuntu:52092: UDP, length 37
23:14:17.385758 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 31
23:14:17.385896 IP ubuntu:52092 > par21s05-in-f3.1e100.net.443: UDP, length 28
^C
21 packets captured
21 packets received by filter
0 packets dropped by kernel
manav@ubuntu:~$
```


TCPdump Commands

3. To capture specific number of packets

```
sudo tcpdump -c 4 -i wlo1
```

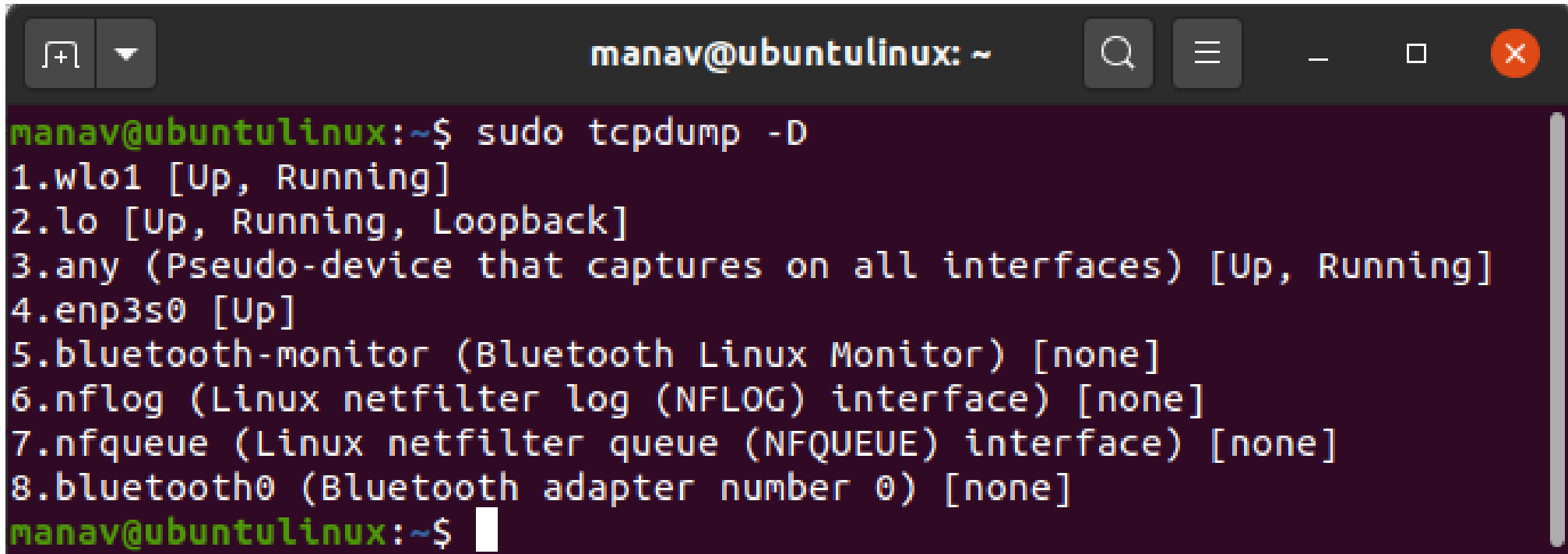
A terminal window titled 'manav@ubuntulinux: ~' with standard window controls. The terminal shows the execution of 'sudo tcpdump -c 4 -i wlo1'. The output indicates that verbose output is suppressed, the interface is wlo1, and 4 packets were captured. It then displays four network packets with their timestamps, source/destination IPs, and protocols. The terminal ends with the prompt 'manav@ubuntulinux:~\$' and a cursor.

```
manav@ubuntulinux:~$ sudo tcpdump -c 4 -i wlo1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlo1, link-type EN10MB (Ethernet), capture size 262144 bytes
23:15:20.257784 IP 192.168.0.3.mdns > 224.0.0.251.mdns: 25 [2q] PTR (QM)? _233637DE._sub._googlecast._tcp.local. PTR (QM)? _googlecast._tcp.local. (61)
23:15:20.259572 IP ubuntulinux.50749 > b.resolvers.Level3.net.domain: 37963+ PTR? 251.0.0.224.in-addr.arpa. (42)
23:15:20.461763 IP b.resolvers.Level3.net.domain > ubuntulinux.50749: 37963 NXDomain 0/1/0 (99)
23:15:20.463051 IP ubuntulinux.54591 > b.resolvers.Level3.net.domain: 7530+ PTR? 3.0.168.192.in-addr.arpa. (42)
4 packets captured
7 packets received by filter
0 packets dropped by kernel
manav@ubuntulinux:~$
```

TCPdump Commands

To display all available interfaces

```
sudo tcpdump -D
```

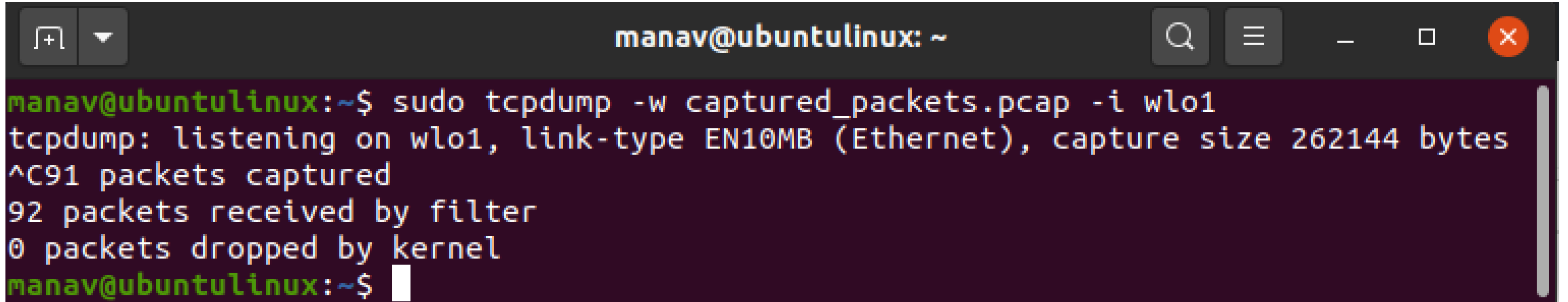
A terminal window titled 'manav@ubuntulinux: ~' with standard window controls. The terminal shows the command 'sudo tcpdump -D' being executed, which lists available network interfaces. The output is as follows:

```
manav@ubuntulinux:~$ sudo tcpdump -D
1.wlo1 [Up, Running]
2.lo [Up, Running, Loopback]
3.any (Pseudo-device that captures on all interfaces) [Up, Running]
4.enp3s0 [Up]
5.bluetooth-monitor (Bluetooth Linux Monitor) [none]
6.nflog (Linux netfilter log (NFLOG) interface) [none]
7.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
8.bluetooth0 (Bluetooth adapter number 0) [none]
manav@ubuntulinux:~$
```

TCPdump Commands

To save captured packets into a file

```
sudo tcpdump -w captured_packets.pcap -i wlo1
```

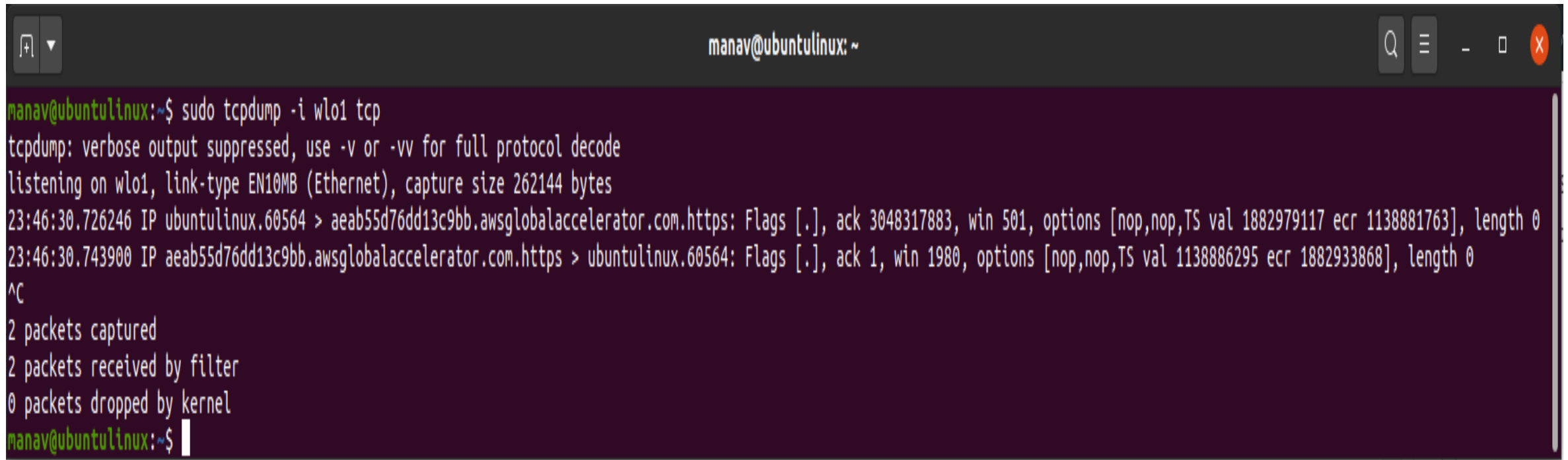
A terminal window titled 'manav@ubuntulinux: ~' with standard window controls (search, menu, zoom, close). The terminal output shows the execution of the 'sudo tcpdump' command, followed by status messages from tcpdump and a final prompt.

```
manav@ubuntulinux:~$ sudo tcpdump -w captured_packets.pcap -i wlo1
tcpdump: listening on wlo1, link-type EN10MB (Ethernet), capture size 262144 bytes
^C91 packets captured
92 packets received by filter
0 packets dropped by kernel
manav@ubuntulinux:~$
```

TCPdump Commands

To capture only TCP packets

```
sudo tcpdump -i wlo1 tcp
```

A terminal window titled 'manav@ubuntulinux: ~' with standard Ubuntu window controls. The terminal shows the execution of 'sudo tcpdump -i wlo1 tcp'. It displays two captured packets: one from 'ubuntulinux.60564' to 'aeab55d76dd13c9bb.awsglobalaccelerator.com.https' and another from 'aeab55d76dd13c9bb.awsglobalaccelerator.com.https' to 'ubuntulinux.60564'. After pressing Ctrl-C, it shows statistics: 2 packets captured, 2 packets received by filter, and 0 packets dropped by kernel.

```
manav@ubuntulinux:~$ sudo tcpdump -i wlo1 tcp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on wlo1, link-type EN10MB (Ethernet), capture size 262144 bytes
23:46:30.726246 IP ubuntulinux.60564 > aeab55d76dd13c9bb.awsglobalaccelerator.com.https: Flags [.], ack 3048317883, win 501, options [nop,nop,TS val 1882979117 ecr 1138881763], length 0
23:46:30.743900 IP aeab55d76dd13c9bb.awsglobalaccelerator.com.https > ubuntulinux.60564: Flags [.], ack 1, win 1980, options [nop,nop,TS val 1138886295 ecr 1882933868], length 0
^C
2 packets captured
2 packets received by filter
0 packets dropped by kernel
manav@ubuntulinux:~$
```

TCPdump Commands Example

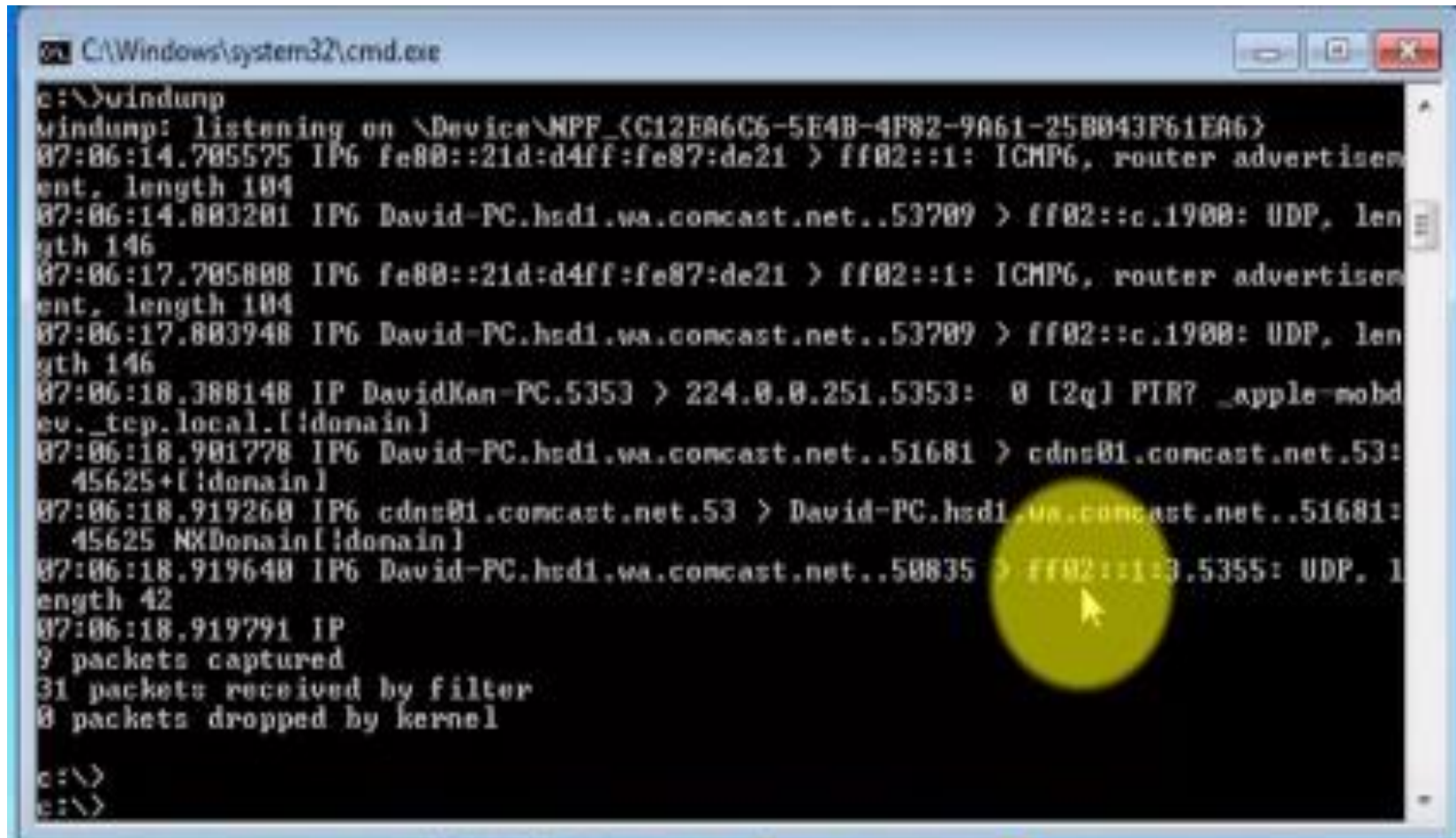
- ▶ To print all packets arriving at or departing from *sundown*:
 - ➔ **\$ tcpdump host sundown**
- ▶ To print traffic between *helios* and either *hot* or *ace*:
 - ➔ **\$ tcpdump host helios and \(hot or ace \)**
- ▶ To print all IP packets between *ace* and any host except *helios*:
 - ➔ **\$ tcpdump ip host ace and not helios**

Windump

- ▶ It is a free version of TCPdump for windows. Windump comes in two parts.
 1. WinPcap: It is a set of network capture drivers which uses to obtain packet-level access to network interfaces in the computer.
 2. Windump a program itself is invoked from the command line after installing the WinPcap library.
- ▶ Windump supports all TCPdump's flags, parameters and settings.
- ▶ WinDump can run under Windows 95 98 ME, NT, 2000 XP, 2003 and Vista.
- ▶ Source:
 - ↳ <https://www.winpcap.org/>

Windump

- ▶ Copy windump.exe into for example c:\ then run command from that location.
- ▶ Download and install WinPcap
- ▶ Type following commands: Its start capturing packets, press ctrl+c to stop capture.



```
C:\Windows\system32\cmd.exe
c:\>windump
windump: listening on \Device\NPF_{C12EA6C6-5E4B-4F82-9A61-25B043F61EA6}
07:06:14.705575 IP6 fe80::21d:d4ff:fe87:de21 > ff02::1: ICMP6, router advertisement, length 104
07:06:14.803201 IP6 David-PC.hsd1.wa.comcast.net..53709 > ff02::c:1900: UDP, length 146
07:06:17.705808 IP6 fe80::21d:d4ff:fe87:de21 > ff02::1: ICMP6, router advertisement, length 104
07:06:17.803948 IP6 David-PC.hsd1.wa.comcast.net..53709 > ff02::c:1900: UDP, length 146
07:06:18.388148 IP DavidKan-PC.5353 > 224.0.0.251.5353: 0 [2q] PTR? _apple-mobd.ev._tcp.local.[!donain]
07:06:18.901778 IP6 David-PC.hsd1.wa.comcast.net..51681 > cdns01.comcast.net.53: 45625+[!donain]
07:06:18.919260 IP6 cdns01.comcast.net.53 > David-PC.hsd1.wa.comcast.net..51681: 45625 NXDonain[!donain]
07:06:18.919640 IP6 David-PC.hsd1.wa.comcast.net..50835 > ff02::1:3.5355: UDP, length 42
07:06:18.919791 IP
9 packets captured
31 packets received by filter
0 packets dropped by kernel

c:\>
c:\>
```

Windump

- ▶ Windump -d , list all the available interfaces on system.
- ▶ Windump -i 1 , to listen to particular interface.
- ▶ Windump -i 1 -w c:\test\mycap.pcap, it will write every packet captured into mycap.pcap file we can also analysis this file into wireshark.

Windump Example

- ▶ See all packets in the capture file
 - ➔ `windump -n -r filename.pcap`
- ▶ Show only the first 2 packets
 - ➔ `windump -n -r filename.pcap -c 2`
- ▶ Tracking host by source MAC address
 - ➔ `windump -n -r filename.pcap -e "ether src 00:a0:cc:3b:bf:fa"`
- ▶ Tracking host by destination MAC address
 - ➔ `windump -n -r filename.pcap -e "ether dst 00:a0:cc:3b:bf:fa"`
- ▶ Tracking host by IP, whether that IP is source or destination
 - ➔ `windump -n -r filename.pcap "host 192.168.0.1"`
- ▶ Track host by source IP
 - ➔ `windump -n -r filename.pcap "src host 192.168.0.1"`
- ▶ Track host by destination IP
 - ➔ `windump -n -r filename.pcap "dst host 192.168.0.1"`

Wireshark

- ▶ Wireshark is an open-source packet analyzer, which is used for **education, analysis, software development, communication protocol development, and network troubleshooting**.
- ▶ It is used to track the packets so that each one is filtered to meet our specific needs. It is commonly called as a **sniffer, network protocol analyzer, and network analyzer**. It is also used by network security engineers to examine security problems.
- ▶ Wireshark is a free to use application which is used to apprehend the data back and forth. It is often called as a free packet sniffer computer application. It puts the network card into an unselective mode, i.e., to accept all the packets which it receives.
- ▶ Wireshark can be used in the following ways:
 - ▶ It is used by network security engineers to examine security problems.
 - ▶ It allows the users to watch all the traffic being passed over the network.
 - ▶ It is used by network engineers to troubleshoot network issues.
 - ▶ It also helps to troubleshoot latency issues and malicious activities on your network.
 - ▶ It can also analyze dropped packets.
 - ▶ It helps us to know how all the devices like laptop, mobile phones, desktop, switch, routers, etc., communicate in a local network or the rest of the world.

Wireshark

- ▶ It has a great GUI. It is free and open source software.
- ▶ Wireshark is very similar to tcpdump, but has a graphical front-end, plus some integrated sorting and filtering options.
- ▶ It offers network monitoring on almost all types of network standards (ethernet, wlan, Bluetooth etc)
- ▶ All the necessary components for monitoring, analyzing and documenting the network traffic are present. It is free to use.
- ▶ You can use it to review traffic captured by tools like tcpdump or WinDump or use it to capture traffic directly.

Ettercap

- ▶ Ettercap is an open-source tool that can be used to support man-in-the-middle attacks on networks.
- ▶ It is capable of **intercepting traffic on a network segment, capturing passwords, and conducting active eavesdropping against a number of common protocols.**
- ▶ Ettercap can capture packets and then write them back onto the network.
- ▶ Ettercap enables the diversion and alteration of data virtually in real-time.
- ▶ Ettercap can also be used for the protocol analysis necessary to analyze network traffic.
- ▶ Ettercap has a nice Graphical User Interface (UI) as well as a command line interface.
- ▶ While Ettercap can support network traffic analysis, the most frequent use of Ettercap is to set up man-in-the-middle attacks using ARP poisoning.
- ▶ Penetration testing you can emulate includes man-in-the-middle attacks, credentials capture, dns spoofing, and DoS attack.

Ettercap

- ▶ A penetration test (pen test) is **an authorized simulated attack performed on a computer system to evaluate its security**.
- ▶ Penetration testers use the same tools, techniques, and processes as attackers to find and demonstrate the business impacts of weaknesses in a system.
- ▶ Ettercap also supports deep analysis of many protocols and includes many features for network and host analysis.
- ▶ Ettercap can also detect a switched local area network (LAN) and use the OS fingerprints to determine the total geometry of the LAN.

Ettercap - Modes of Operation

- ▶ Ettercap offers four modes of operation.
- ▶ These are as follows:
 - ➔ **IP-based:** packets are filtered based on IP source and destination.
 - ➔ **MAC-based:** packets are filtered based on MAC address, useful for sniffing connections through a gateway.
 - ➔ **ARP-based:** uses ARP poisoning to sniff on a switched LAN between two hosts.
 - ➔ Note: sniffing is **the act of intercepting and monitoring traffic on a network**
 - ➔ **Note: Address Resolution Protocol** is one of the most important protocols of the network layer in the OSI model which helps in finding the MAC(Media Access Control) address given the IP address of the system i.e. the main duty of the ARP is to convert the 32-bit IP address(for IPv4) to 48-bit address i.e. the MAC address.
 - ➔ **ARP broadcasts a request packet to all the machines on the LAN and asks if any of the machines are using that particular IP address.** When a machine recognizes the IP address as its own, it sends a reply so ARP can update the cache for future reference and proceed with the communication.
 - ➔ ARP Poisoning (also known as ARP Spoofing) is a type of cyber attack carried out over a Local Area Network (LAN) that involves sending malicious ARP packets to a default gateway on a LAN in order to change the pairings in its IP to MAC address table.
 - ➔ **PublicARP-based:** uses ARP poisoning to sniff on a switched LAN from a victim host to all other hosts.

Features of Ettercap

- ▶ It supports in setting up a filter that searches for a particular string in the TCP or UDP payload and replaces it with a custom string or drops the entire packet.
- ▶ It can determine the OS of the victim host and its network adapter.
- ▶ It can kill connections of choices from the connection-list.
- ▶ It can hijack DNS requests.
- ▶ Help videos:
 - MITM With Ettercap - ARP Poisoning
 - <https://www.youtube.com/watch?v=3UD738uE7Tg>
 - Ettercap as cyber security tools
 - <https://www.youtube.com/watch?v=ejSK5rm3ITl>

Hping

- ▶ Hping is a free packet generator and analyzer for the TCP/IP protocol. It is one of the tools for security auditing and testing of firewalls and networks. It is preinstalled on kali linux.
- ▶ It was used to exploit the idle scanning technique and now implemented in the NMAP security scanner.
- ▶ The new version of hping, hping3, is scriptable using the tcl language and implements an engine for string based, human readable description of TCP/IP packets, so that the programmer can write scripts related to low level TCP/IP packet manipulation and analysis in very short time.
- ▶ Hping also has a listen mode, enabling it to be used as an unsophisticated backdoor for covert remote access or file transfers.
- ▶ Hping's "listen" mode can be used for receiving data.
- ▶ When hping is in listen mode, it monitors traffic for a special "signature" that indicates it should capture the data to follow.

Use of Hping

- ▶ Determining a Host's Status When Ping Doesn't Work.
- ▶ Testing Firewall Rules.
- ▶ Stealth Port Scanning.
- ▶ Note: Stealth scan types are those where packet flags cause the target system to respond without having a fully established connection. Stealth scanning is used by hackers to avoid the intrusion detection system (IDS), making it a significant threat. Therefore, it's important for system administrators to run stealth scans on their systems to penetration test the firewall and the functionality of the IDS.
- ▶ Remote OS Fingerprinting.
- ▶ Hping3 tutorial
- ▶ <https://www.youtube.com/watch?v=ud0rIWyhUU>

Use of Hping

- ▶ Simple command similar to ping, to check host is alive.

```
root@kali: ~  
root@kali: ~  
(rootkali) - [~]  
# hping3 192.168.128.132  
HPING 192.168.128.132 (eth0 192.168.128.132): NO FLAGS are set, 40 headers + 0 data bytes  
len=46 ip=192.168.128.132 ttl=64 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=7.7 ms  
len=46 ip=192.168.128.132 ttl=64 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=4.1 ms  
len=46 ip=192.168.128.132 ttl=64 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=7.2 ms  
^C  
--- 192.168.128.132 hping statistic ---  
3 packets transmitted, 3 packets received, 0% packet loss  
round-trip min/avg/max = 4.1/6.3/7.7 ms
```

Use of Hping

- ▶ Below hping3 command will scan 1-1024 ports of 192.168.128.132 (which is the IP of metasploitable)

```
(rootkali)-[~]
# hping3 --scan 1-1024 192.168.128.132
Scanning 192.168.128.132 (192.168.128.132), port 1-1024
1024 ports to scan, use -y to see all the replies
+-----+-----+-----+-----+-----+-----+
|port| serv name | flags |ttl| id  | win | len |
+-----+-----+-----+-----+-----+-----+
All replies received. Done.
Not responding ports: (21 ftp) (22 ssh) (23 telnet) (25 smtp) (53 domain) (80 http) (111 sunrpc) (13
9 netbios-ssn) (445 microsoft-d) (512 exec) (513 login) (514 shell)
```

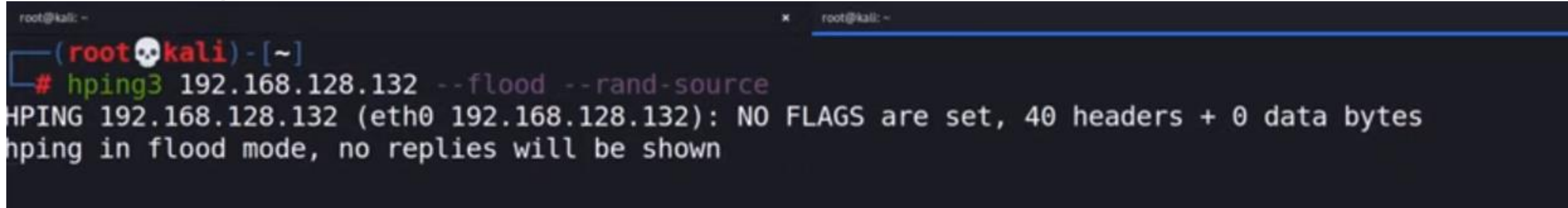
Use of Hping

- ▶ Below hping3 command will scan 1-1024 ports of 192.168.128.132 (which is the IP of metasploitable) with -S option (SYN flag)

```
root@kali: ~  
# hping3 --scan 1-1024 192.168.128.132 -S  
Scanning 192.168.128.132 (192.168.128.132), port 1-1024  
1024 ports to scan, use -V to see all the replies  
+-----+-----+-----+-----+-----+-----+  
|port| serv name | flags  |ttl| id  | win | len |  
+-----+-----+-----+-----+-----+-----+  
21 ftp      : .S..A... 64    0  5840  46  
22 ssh      : .S..A... 64    0  5840  46  
23 telnet   : .S..A... 64    0  5840  46  
25 smtp     : .S..A... 64    0  5840  46  
53 domain   : .S..A... 64    0  5840  46  
80 http     : .S..A... 64    0  5840  46  
111 sunrpc  : .S..A... 64    0  5840  46  
139 netbios-ssn: .S..A... 64    0  5840  46  
445 microsoft-d: .S..A... 64    0  5840  46  
512 exec    : .S..A... 64    0  5840  46  
513 login   : .S..A... 64    0  5840  46  
514 shell   : .S..A... 64    0  5840  46
```

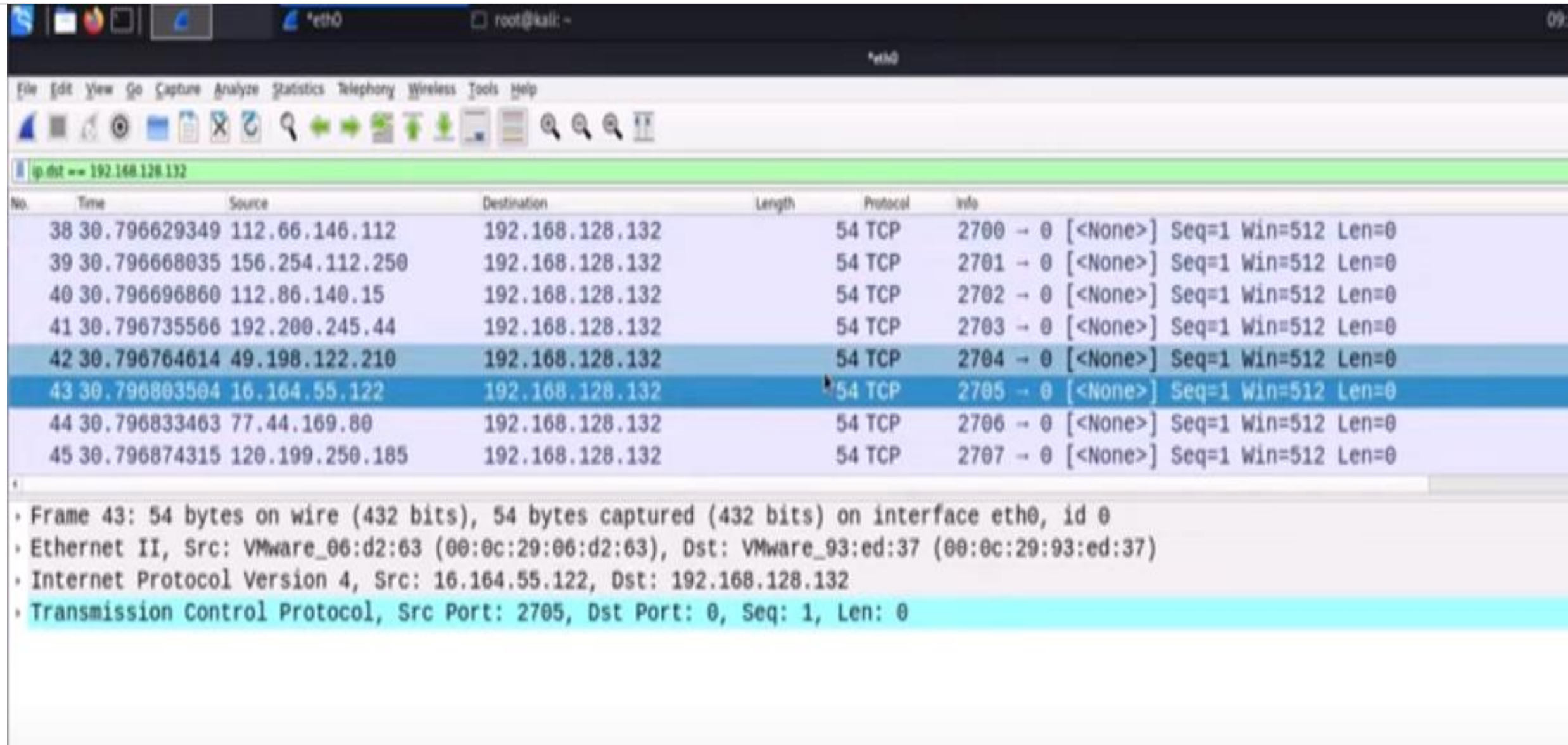
Use of Hping

- ▶ This tool is used to perform denial of service attack. Below command flood 192.168.128.132 with lots of traffic from random sources. (sender IP is from various random source due to random-source argument, so no one easily detect this attack.)
- ▶ We can check this using Wireshark. Start packet capturing and then run following command and then filter IP address=192.168.128.132, then we can see random source IP.

A terminal window screenshot from a Kali Linux machine. The prompt is root@kali: ~. The user has entered the command # hping3 192.168.128.132 --flood --rand-source. The output shows HPING 192.168.128.132 (eth0 192.168.128.132): NO FLAGS are set, 40 headers + 0 data bytes. A follow-up line states hping in flood mode, no replies will be shown.

```
root@kali: ~  
(root@kali) - [~]  
# hping3 192.168.128.132 --flood --rand-source  
HPING 192.168.128.132 (eth0 192.168.128.132): NO FLAGS are set, 40 headers + 0 data bytes  
hping in flood mode, no replies will be shown
```


Use of Hping



The image shows a Wireshark packet capture on interface eth0. The filter is set to 'ip.dst == 192.168.128.132'. The packet list shows several TCP SYN packets from various source IP addresses to the destination 192.168.128.132. Packet 43 is selected, and its details are shown in the packet details pane.

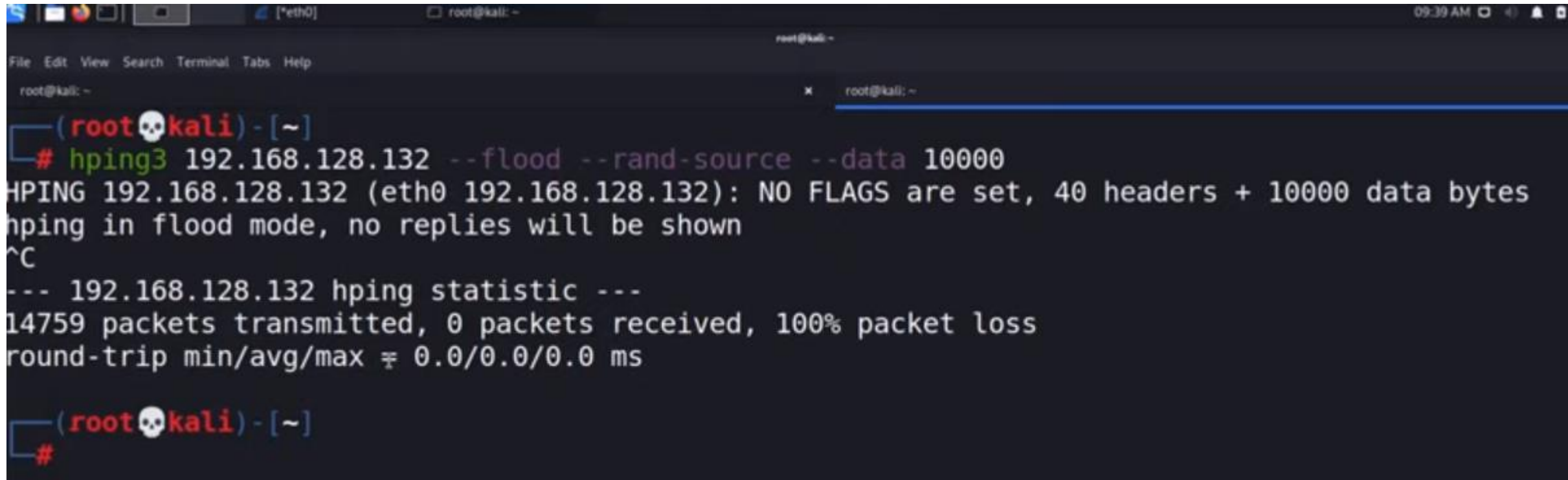
No.	Time	Source	Destination	Length	Protocol	Info
38	30.796629349	112.66.146.112	192.168.128.132	54	TCP	2700 → 0 [<None>] Seq=1 Win=512 Len=0
39	30.796668035	156.254.112.250	192.168.128.132	54	TCP	2701 → 0 [<None>] Seq=1 Win=512 Len=0
40	30.796696860	112.86.140.15	192.168.128.132	54	TCP	2702 → 0 [<None>] Seq=1 Win=512 Len=0
41	30.796735566	192.200.245.44	192.168.128.132	54	TCP	2703 → 0 [<None>] Seq=1 Win=512 Len=0
42	30.796764614	49.198.122.210	192.168.128.132	54	TCP	2704 → 0 [<None>] Seq=1 Win=512 Len=0
43	30.796803504	16.164.55.122	192.168.128.132	54	TCP	2705 → 0 [<None>] Seq=1 Win=512 Len=0
44	30.796833463	77.44.169.80	192.168.128.132	54	TCP	2706 → 0 [<None>] Seq=1 Win=512 Len=0
45	30.796874315	120.199.250.185	192.168.128.132	54	TCP	2707 → 0 [<None>] Seq=1 Win=512 Len=0

Frame 43 details:

- Frame 43: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0
- Ethernet II, Src: VMware_06:d2:63 (00:0c:29:06:d2:63), Dst: VMware_93:ed:37 (00:0c:29:93:ed:37)
- Internet Protocol Version 4, Src: 16.164.55.122, Dst: 192.168.128.132
- Transmission Control Protocol, Src Port: 2705, Dst Port: 0, Seq: 1, Len: 0

Use of Hping

- ▶ Denial of service attack with `-data` field. It can slow down the server.

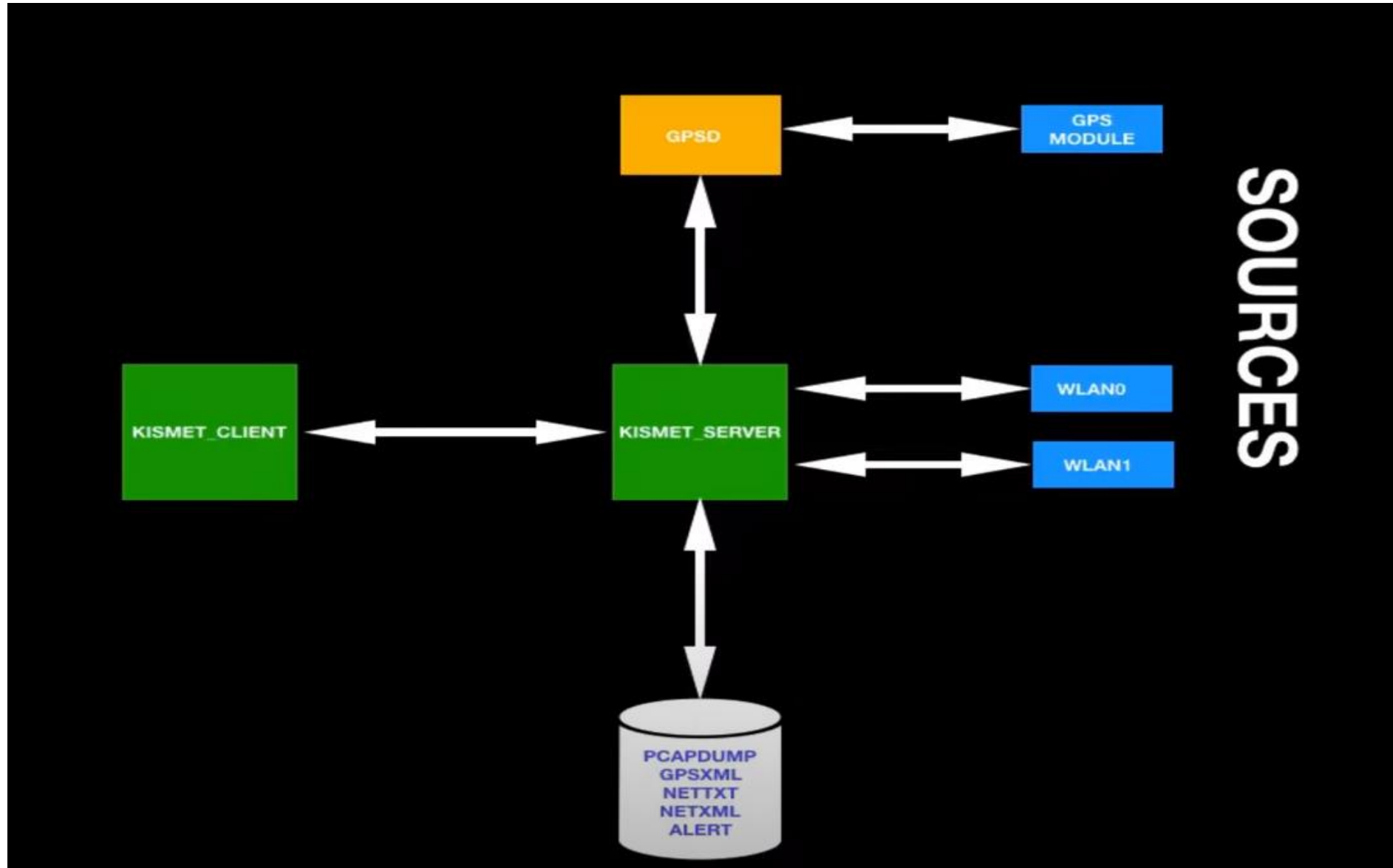


```
root@kali: ~  
File Edit View Search Terminal Tabs Help  
root@kali: ~  
(root👤kali) - [~]  
# hping3 192.168.128.132 --flood --rand-source --data 10000  
HPING 192.168.128.132 (eth0 192.168.128.132): NO FLAGS are set, 40 headers + 10000 data bytes  
hping in flood mode, no replies will be shown  
^C  
--- 192.168.128.132 hping statistic ---  
14759 packets transmitted, 0 packets received, 100% packet loss  
round-trip min/avg/max = 0.0/0.0/0.0 ms  
(root👤kali) - [~]  
#
```

Kismet

- ▶ Kismet is a free software and it is network detector, packet sniffer and intrusion detection system for 802.11 wireless LANs.
- ▶ Kismet will work with any wireless card which supports raw monitoring mode and can sniff 802.11a, 802.11b, 802.11g and 802.11n traffic.
- ▶ This runs under Linux, FreeBSD, NetBSD, openBSD, and mac OS X, Microsoft windows.
- ▶ Kismet has three separate parts.
- ▶ These are as follows:
 - A drone: it can be used to collect packets and then pass them on to a server for interpretation.
 - A server: it can either be used in conjunction with a drone or on its own, interpreting packet data and extrapolating wireless information and organizing it.
 - The client: it communicates with the server and displays the information the server collects.

Kismet



Features of Kismet

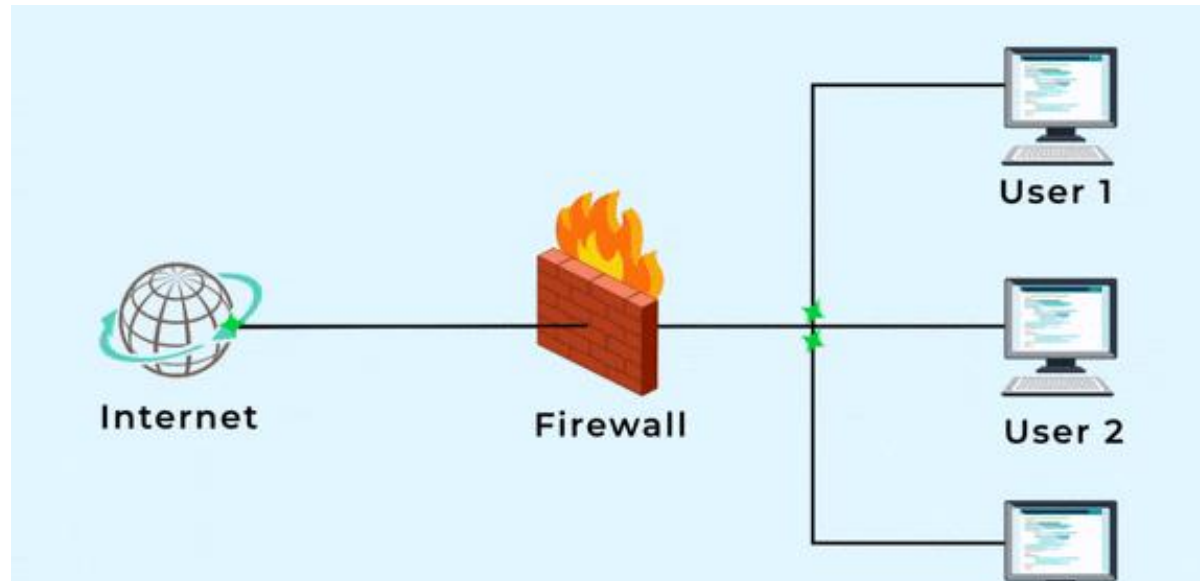
- ▶ Kismet differs from other wireless network detector in working passively.
- ▶ It is able to detect the presence of both wireless access and wireless client.
- ▶ Kismet also includes basic wireless IDS features such as detecting active wireless sniffing programs including NetStumbler, as well as a number of wireless network attacks.
- ▶ It has the ability to log all sniffed packets and save them in a tcpdump/wireshark compatible file format.
- ▶ It has ability to detect default or not configured networks, probe requests, and determine what level of wireless encryption is used on a given access point.

Features of Kismet – Cont.

- ▶ Kismet supports channel hopping.
- ▶ Channel hopping **avoids busy channels, thereby reducing the ping time, increasing robustness.**
- ▶ Kismet also supports logging of the geographical coordinates of the network if the input from a GPS receiver is additionally available.
- ▶ Discover wireless networks with kismet
- ▶ <https://www.youtube.com/watch?v=IBzUIgTwuYE>

Firewall Basics

- ▶ A **firewall** is a device which is used to **control** the **flow of traffic** into and out-of network. In other words, it is a **security device** which **installed between** two networks, **internal network** to **outside network** (more often the internet).
- ▶ **Based on** the **rule define** in the firewall **data will be passed** to one network to other network.
- ▶ Example: Block all traffic by default and explicitly enable only specific traffic to known services. ...
- ▶ The primary job of a firewall is to **secure the inside network from the internet**.
- ▶ Systems on one side of the firewall are protected from systems on the other side.



Firewall Basics

- ▶ Firewalls can be implemented as both hardware and software, or a combination of both. It's a part of almost all operating systems.
- ▶ At its core, firewall examines traffic on a network interface to determine whether packets should be allowed to enter or leave the interface. T
- ▶ Thus, firewall software blocks inbound connections to a system's services that shouldn't be exposed to other systems on a public network and can also be used to block outbound traffic from a system to a network.
- ▶ Firewalls block traffic to known malware sites to try and limit the potential damage of downloading an infected file.

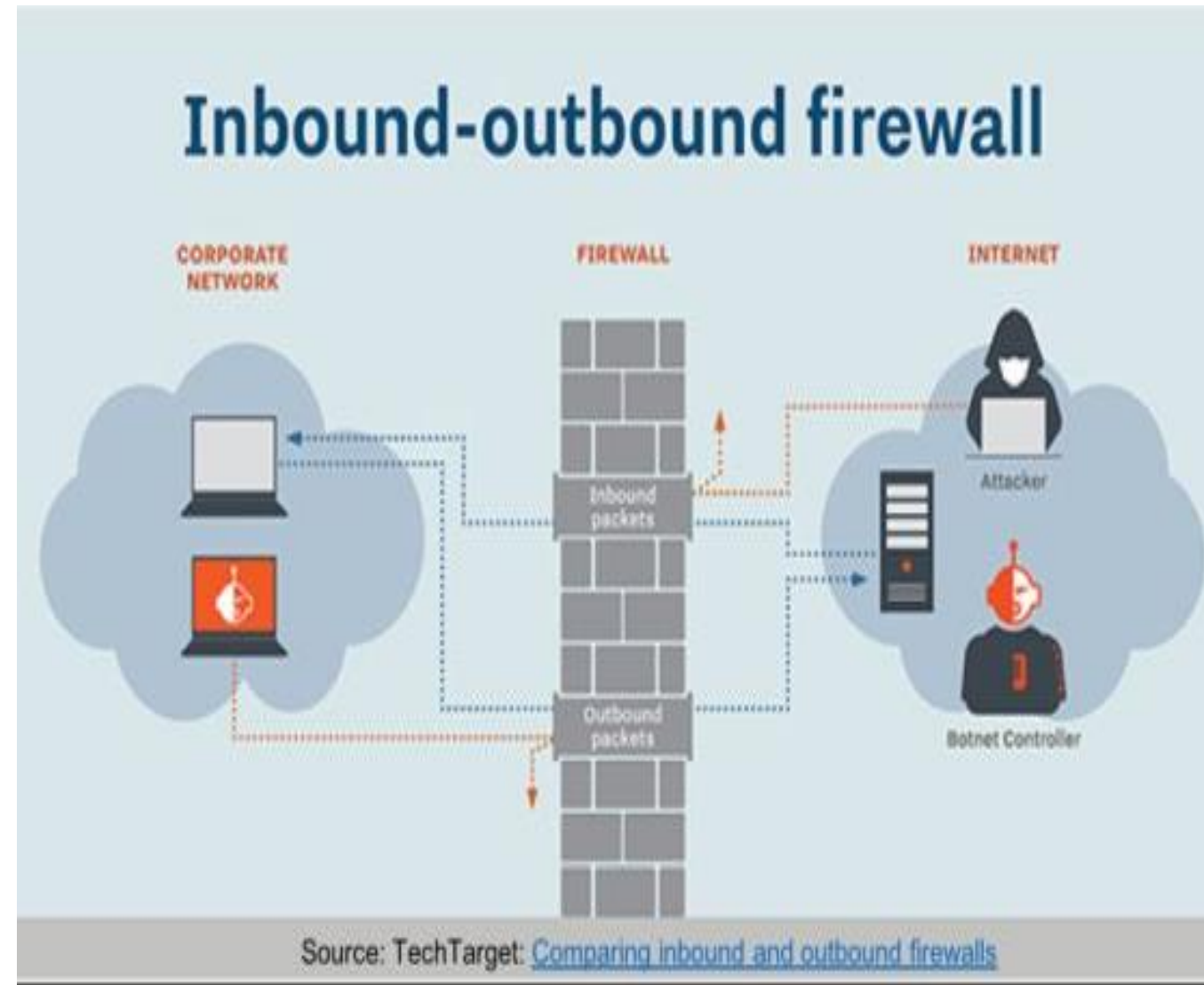
Firewall Basics

► For example:

- Consider **LAN** is corporate or our campus network and **WAN** is internet.
- If we place firewall between the two networks then it will control the flow of the whole traffic and based on rule define into firewall.
- It will **allow or deny** the traffic.

► Firewalls generally **filter** traffic based on **two methodologies**:

1. A firewall can **allow any traffic** except what is specified as restricted part. It depends on the type of firewall used, the source, the destination addresses, and the ports.
2. A firewall can **deny any traffic** that does not meet the specific criteria based on the network layer on which the firewall operates.



HOW A FIREWALL PROTECTS A NETWORK

- ▶ Firewalls are only as effective as the rules they're configured to enforce. Most firewalls have three ways to enforce a rule for network traffic
- ▶ **Accept the packet** and pass it on to its intended destination.
- ▶ **Deny the packet and indicate the denial** with an Internet Control Message Protocol (ICMP) message or similar acknowledgment to the sender. This provides explicit feedback that such traffic is not permitted through the firewall.
- ▶ **Drop the packet without any acknowledgment.** This ends the packet's life on the network. No information is sent to the packet's sender. This method minimizes the sender's ability to deduce information about the protected network, but it may also adversely impact network performance for certain types of traffic. For example, a client may repeatedly attempt to connect to a service because it hasn't received an explicit message that the service isn't available.
- ▶ Most firewalls drop packets as their default policy for traffic that isn't permitted. When building a ruleset, start with the concept of least privilege or deny all. It's safer to start with a firewall that rejects every incoming connection and open only the necessary holes for services we want to expose, rather than to start with an open firewall that exposes all of your network's resources.

Firewall Types

- ▶ Firewall is the first destination for the traffic coming to your internal network.
- ▶ So, **anything** which **comes** to your **internal network** **passes through** the **firewall** and any outgoing traffic will also pass through the firewall before leaving your network completely.
- ▶ This is the reason that sometimes this type of firewall filter is also called **screening routers**.
- ▶ Firewall types the way a firewall provides greater protection relies on the firewall itself, and on the policies that are configured on it.
- ▶ The Following types of firewall are:
 1. Packet-Filter Firewall
 2. Circuit-Level Gateways
 3. Stateful Packet-Inspection (SPI)
 4. Proxy Firewall
 5. Application Gateways
 6. Next-Gen firewalls
 7. Software Firewall
 8. Hardware Firewall
 9. Cloud Firewall

Packet Filtering Firewall

- ▶ As the most “basic” and oldest type of firewall architecture,
- ▶ **Packet-filtering** firewalls basically **create** a **checkpoint** at a traffic router or switch.
- ▶ The firewall performs a simple **check** of the **data packets** coming through the router—inspecting information such as the **destination** and **origination IP address, packet type, port number**, and other surface-level information **without opening** up the **packet** to inspect its contents.
- ▶ If the information **packet doesn't pass** the **inspection**, it is **dropped**.
- ▶ The **good thing** about these firewalls is that they **aren't very resource-intensive**.
- ▶ This means they don't have a **huge impact** on **system performance** and are **relatively simple**.
- ▶ However, they're also **relatively easy** to **bypass** compared to firewalls with more robust inspection capabilities.

Circuit-Level Gateways

- ▶ As another simplistic firewall type that is meant to quickly and easily approve or deny traffic without consuming significant computing resources.
- ▶ Circuit-level gateways work by **verifying** the **transmission control protocol (TCP) handshake**.
- ▶ Note: Transmission Control Protocol (TCP) provides a secure and reliable connection between two devices using the 3-way handshake process. TCP uses the full-duplex connection to synchronize (SYN) and acknowledge (ACK) each other on both sides. There are three steps for both establishing and closing a connection. They are – SYN, SYN-ACK, and ACK.
- ▶ This TCP handshake check is designed to **make sure** that the session the **packet** is **from legitimate**.
- ▶ While **extremely resource-efficient**, these firewalls **does** not **check the packet** itself.
- ▶ So, if a **packet held malware**, but had the right TCP handshake, it **would pass** right through.
- ▶ This is why circuit-level gateways are **not enough** to **protect** your **business** by themselves.

Stateful Inspection Firewalls

- ▶ These firewalls **combine both packet inspection technology** and **TCP handshake** verification to create a level of protection greater than either of the previous two architectures could provide alone.
- ▶ However, these firewalls do put **more** of a strain on **computing resources** as well. This may **slow down** the **transfer** of legitimate packets compared to the other solutions.

Proxy Firewalls

- ▶ **Proxy** firewalls operate at the **application layer** to **filter incoming traffic** between your network and the traffic source—hence, the name “**application-level gateway**.”
- ▶ Rather than letting traffic connect directly, the proxy firewall first establishes a connection to the source of the traffic and **inspects** the **incoming data packet**.
- ▶ This check is similar to the stateful inspection firewall in that it looks at both the packet and at the TCP handshake protocol.
- ▶ However, proxy **firewalls** may also **perform deep-layer packet inspections**, **checking** the **actual contents** of the information packet to **verify** that it contains **no malware**.
- ▶ Once the check is complete, and the packet is approved to connect to the destination, the proxy sends it off.
- ▶ This creates an extra layer of separation between the “client” (the system where the packet originated) and the individual devices on your network—obscuring them to **create additional privacy** and **protection** for your network.
- ▶ It's that they can create **significant slowdown** because of the extra steps.

Application Level Firewall

- ▶ These firewalls operate at the application level.
- ▶ In other words, they filter the **traffic only** with regards to the **application** (or **service**) for which they are intended.
- ▶ For example, a firewall for monitoring traffic to all the web applications your network uses.
- ▶ Types of application level firewall:
 - ▶ Network-Based Application Firewalls: Scan and monitor network-based traffic destined for the for any specific application.
 - ▶ Host-Based Application Firewalls: Monitor all the incoming and outgoing traffic initiated by an application, system or host.

Next-Generation Firewalls

- ▶ Many of the **most recently-released** firewall products are being advertised as “**next-generation**” architectures.
- ▶ Some common features of next-generation firewall architectures **include deep-packet inspection** (checking the actual contents of the data packet), **TCP handshake checks**, and **surface-level packet inspection**.
- ▶ Next-generation firewalls may include other technologies as well, such as **intrusion prevention systems (IPSs)** that work to **automatically stop attacks** against your network.

Software Firewalls

- ▶ **Software firewalls** include any type of firewall that is **installed** on a **local device rather than** a **separate** piece of **hardware**.
- ▶ The big benefit of a software firewall is that it's highly useful for creating defense in depth by **isolating individual network endpoints** from one another.
- ▶ However, **maintaining** individual software firewalls on different devices can be **difficult** and **time-consuming**.
- ▶ Furthermore, not every device on a network may be compatible with a single software firewall, which may mean having to use several different software firewalls to cover every asset.
- ▶ **Advantages:**
 - ▶ Helpful in blocking particular sites
 - ▶ Juniors and parental controls can be supervised
 - ▶ Ease in maintenance
 - ▶ Valuable for home users
 - ▶ Assignment of different levels of access and permissions to the user can be done with ease

Software Firewalls

▶ **Disadvantages**

- ▶ Installation and up-gradation are required on individual computers.
- ▶ Slow Performance of the system.
- ▶ Due to its installation, system resources are consumed.
- ▶ Does not work on smart TVs, gaming consoles, etc.

Hardware Firewalls

- ▶ It is physical piece of equipment planned to perform firewall duties.
- ▶ A hardware firewall can be a computer or a dedicated piece of equipment which serve as a firewall.
- ▶ Hardware firewall are incorporated into the router that is situated between the computer and the internet gateway.
- ▶ **Advantages:**
 - ▶ Independently run so less prone to cyber-attacks.
 - ▶ Installation is external so resources are free from the server.
 - ▶ Increased bandwidth enables the handling of more data packets per second.
 - ▶ Reduced latency.
 - ▶ VPN connection is also supported for increased security and encryption.
- ▶ **Disadvantages:**
 - ▶ Hardware devices can take extra space

Cloud Firewalls

- ▶ Whenever a cloud solution is used to deliver a firewall, it can be called a cloud firewall, or **firewall-as-a-service (FaaS)**.
- ▶ A cloud Firewall is nothing but a Firewall that is deployed in the cloud and these cloud Firewalls form a virtual barrier, to prevent malicious network traffic in the cloud, they function as same as traditional Firewalls, but the only difference is the cloud firewall is hosted in a cloud platform.
- ▶ Cloud Firewalls act as a security product that acts as a shield and protects from unauthorized network traffic and this protection is provided to different cloud components like Cloud CRM, Cloud Database, Email Cloud.