

## **PRACTICAL -1**

**Aim : Introduction Virtualization Environment configuration and Cyber Lab setup (Kali, VM ware and Oracle Virtual Box).**

### **Virtualization**

- Virtualization is technology that allows you to create multiple simulated environments or dedicated resources from a single, physical hardware system.
- Software called a hypervisor connects directly to that hardware and allows you to split one system into separate, distinct, and secure environments known as virtual machines (VMs).
- These VMs rely on the hypervisor's ability to separate the machine's resources from the hardware and distribute them appropriately.
- Virtualization helps you get the most value from previous investments.
- The physical hardware, equipped with a hypervisor, is called the host, while the many VMs that use its resources are guests.
- These guests treat computing resources—like CPU, memory, and storage—as a pool of resources that can easily be relocated.
- Operators can control virtual instances of CPU, memory, storage, and other resources, so guests receive the resources they need when they need them.

### **Types of Virtualizations**

There are six areas of IT where virtualization is making headway:

1. **Network virtualization** is a method of combining the available resources in a network by splitting up the available bandwidth into channels, each of which is independent from the others and can be assigned -- or reassigned -- to a particular server or device in real time.  
-The idea is that virtualization disguises the true complexity of the network by separating it into manageable parts, much like your partitioned hard drive makes it easier to manage your files.
2. **Storage virtualization** is the pooling of physical storage from multiple network storage devices into what appears to be a single storage device that is managed from a central console. Storage virtualization is commonly used in storage area networks.
3. **Server virtualization** is the masking of server resources -- including the number and identity of individual physical servers, processors and operating

systems -- from server users. The intention is to spare the user from having to understand and manage complicated details of server resources while increasing resource sharing and utilization and maintaining the capacity to expand later.

- The layer of software that enables this abstraction is often referred to as the hypervisor.
- The most common hypervisor -- Type 1 -- is designed to sit directly on bare metal and provide the ability to virtualize the hardware platform for use by the virtual machines. KVM virtualization is a Linux kernel-based virtualization hypervisor that provides Type 1 virtualization benefits like other hypervisors. KVM is licensed under open source.
- A Type 2 hypervisor requires a host operating system and is more often used for testing and labs.

4. **Data virtualization** is abstracting the traditional technical details of data and data management, such as location, performance, or format, in favor of broader access and more resiliency tied to business needs.
5. **Desktop virtualization** is virtualizing a workstation load rather than a server. This allows the user to access the desktop remotely, typically using a thin client at the desk.
  - Since the workstation is essentially running in a data center server, access to it can be both more secure and portable. The operating system license does still need to be accounted for as well as the infrastructure.
6. **Application virtualization** is abstracting the application layer away from the operating system. This way, the application can run in an encapsulated form without being depended upon by the operating system underneath. This can allow a Windows application to run on Linux and vice versa, in addition to adding a level of isolation.

## **Virtualization Environment**

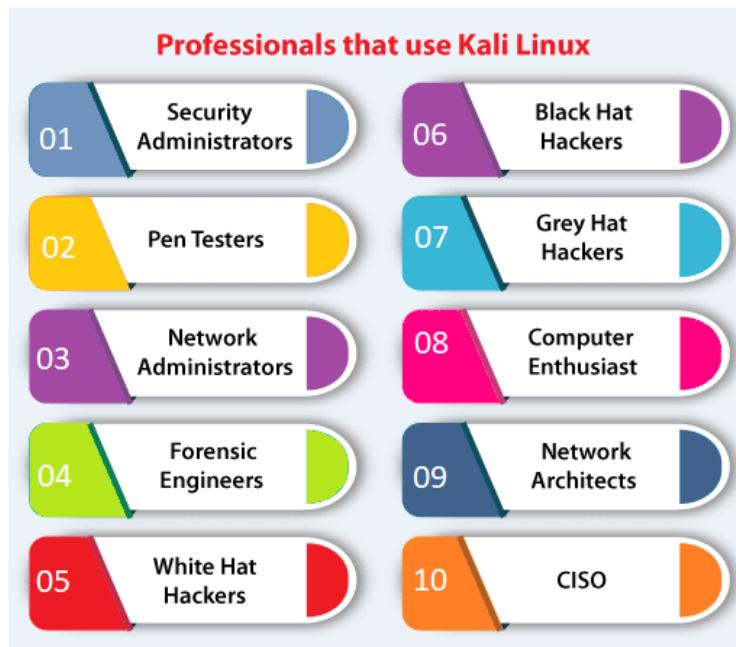
- A virtualization environment allows the creation of isolated and independent virtual machines (VMs) that run multiple operating systems (OS) or applications.
  - In a virtualization environment, a hypervisor is installed on the host machine, which creates multiple virtual machines with their own resources, such as CPU, memory, storage, and network interfaces.
  - Each virtual machine can run its own operating system and applications, and is isolated from other virtual machines on the same host machine.

Virtualization environments can be used in cybersecurity for various purposes, including:

1. Testing and development: Cybersecurity professionals can use virtualization environments to test and develop new security technologies and configurations without risking damage to the production environment.
2. Malware analysis: Virtualization environments can be used to analyze and study malware in a controlled environment. Malware can be executed in a virtual machine without affecting the host machine or other virtual machines.
3. Forensics: Virtualization environments can be used to conduct forensic investigations on compromised systems. Investigators can create a copy of the compromised system in a virtual machine and analyze it without affecting the original system.
4. Sandboxing: Virtualization environments can be used to isolate potentially dangerous applications and websites in a sandboxed environment. If an application or website is found to be malicious, it can be terminated without affecting the host machine or other virtual machines.

## Kali Linux

- Kali Linux is a Debian-based Linux distribution that is designed for digital forensics and penetration testing.
- It is funded and maintained by Offensive Security, an information training company. Kali Linux was developed through the rewrite of BackTrack by Mati Aharoni and Devon Kearns of Offensive Security.
- Kali Linux comes with a large number of tools that are well suited to a variety of information security tasks, including penetration testing, computer forensics, security research, and reverse engineering.
- Kali Linux is a one-of-a-kind operating system since it is one of the few platforms that are freely utilized by both good and bad guys.
- This operating system is widely used by both Security Administrators and Black Hat Hackers.
- One is responsible for detecting and preventing security breaches, while the other is responsible for identifying and perhaps exploiting security breaches.
- The number of tools configured and preinstalled on the operating system makes Kali Linux a Swiss Army Knife in any security professional's toolbox.



## VM ware

- VMware is a company that provides virtualization software that enables users to create and manage virtual machines. VMware's virtualization software allows multiple operating systems and applications to run on a single physical server, which can help businesses save on hardware costs and improve efficiency.



VMware's virtualization products include:

1. **VMware Workstation:** This is a desktop virtualization product that allows users to create and run multiple virtual machines on a single physical computer.
  2. **VMware Fusion:** This is a desktop virtualization product for Mac computers that allows users to run Windows, Linux, and other operating systems on a Mac without having to reboot.
  3. **VMware vSphere:** This is a server virtualization product that allows users to create and manage virtual machines on a large scale.
  4. **VMware ESXi:** This is a hypervisor that allows users to create and run multiple virtual machines on a single physical server.
- VMware's virtualization software is widely used in enterprise computing environments, where it can help businesses reduce hardware costs, increase efficiency, and improve flexibility. Virtualization technology has also become

increasingly popular in cloud computing environments, where it enables cloud service providers to create and manage virtual machines on a large scale.

## Oracle Virtual Box

- Oracle VM VirtualBox is cross-platform virtualization software.
- It allows users to extend their existing computer to run multiple operating systems including Microsoft Windows, Mac OS X, Linux, and Oracle Solaris, at the same time.
- Designed for IT professionals and developers, Oracle VM VirtualBox is ideal for testing, developing, demonstrating, and deploying solutions across multiple platforms from one machine.
- Oracle VM VirtualBox can display virtual machines remotely, meaning that a virtual machine can execute on one computer even though the virtual machine will be displayed on a second computer.
- The virtual machine can be controlled from the second computer, as if the virtual machine was running on that computer.



- Oracle VirtualBox is a type-2 hypervisor that allows users to create and run virtual machines on a host operating system. From a cybersecurity perspective, VirtualBox can be a valuable tool for testing and developing software, as well as for isolating potentially risky applications or operating systems from the host environment.

Here are some key points to consider when assessing VirtualBox from a cybersecurity perspective:

1. Virtualization can increase security: Virtual machines can be isolated from the host operating system, providing an additional layer of security. Virtual machines can also be easily backed up and restored, which can be useful in the event of a security breach.

2. Proper configuration is key: As with any software, VirtualBox should be properly configured to ensure maximum security. This includes setting strong passwords, enabling encryption, and configuring firewalls.
3. VirtualBox vulnerabilities: As with any software, VirtualBox is not immune to vulnerabilities. Users should stay up-to-date with security patches and updates, and regularly review security advisories and bulletins.
4. Shared folders and networking: VirtualBox allows for shared folders and networking between the host and guest operating systems. While this can be convenient, it also introduces potential security risks. Users should carefully configure shared folders and networking to minimize these risks.

## Configuration/Installation of Virtual Environment in Oracle Virtual Box .

### 1. Download and Install VirtualBox



The screenshot shows the official Oracle VirtualBox download page. At the top left is the Oracle logo, which is a blue cube with the word "ORACLE" on one face and "VirtualBox" on another. To the right of the logo, the word "VirtualBox" is written in large blue letters, followed by a "Download VirtualBox" button. Below the logo, there's a navigation menu with links like "About", "Screenshots", "Downloads", "Documentation", "End-user docs", "Technical docs", "Contribute", and "Community". A red box highlights a list of "VirtualBox binaries" under the "Downloads" section, which includes links for Windows hosts, OS X hosts, Linux distributions, Solaris hosts, and Solaris 11 IPS hosts. The text below the list states that the binaries are released under the terms of the GPL version 2. It also mentions a "changelog" for what has changed and provides links for SHA256 and MD5 checksums.

Here you will find links to VirtualBox binaries and its source code.

### VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.0 packages, see [VirtualBox 6.0 builds](#).

If you're looking for the latest VirtualBox 5.2 packages, see [VirtualBox 5.2 builds](#).

### VirtualBox 6.1.22 platform packages

- [Windows hosts](#)
- [OS X hosts](#)
- [Linux distributions](#)
- [Solaris hosts](#)
- [Solaris 11 IPS hosts](#)

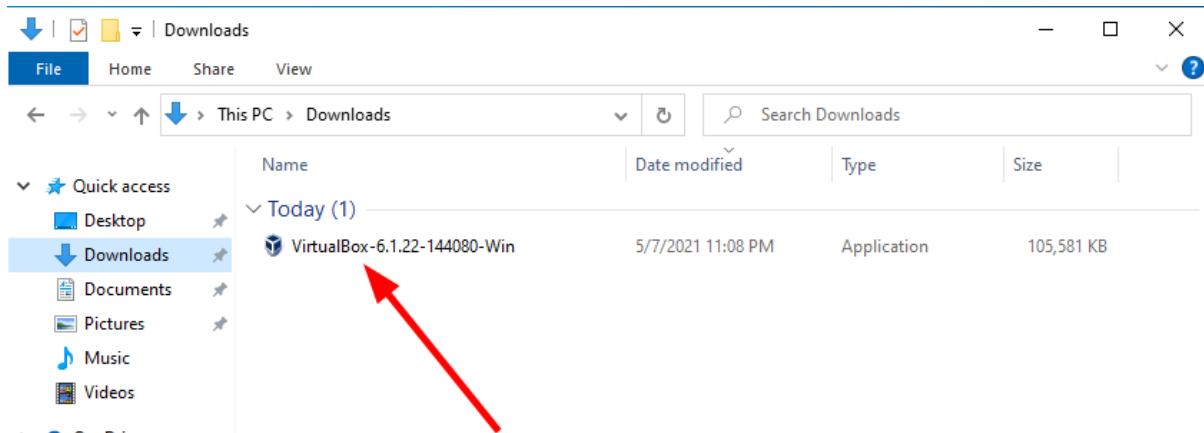
The binaries are released under the terms of the GPL version 2.

See the [changelog](#) for what has changed.

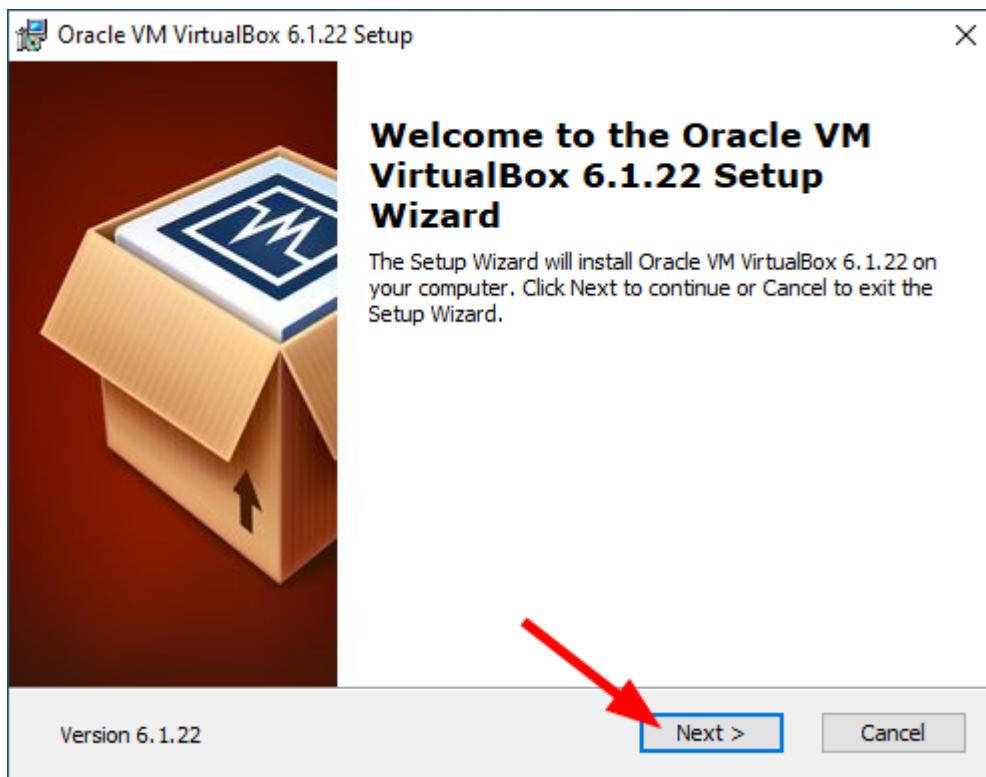
You might want to compare the checksums to verify the integrity of downloaded packages:

- [SHA256 checksums](#), [MD5 checksums](#)

Once the download is completed, double-click on the installation file to launch the VirtualBox installer.

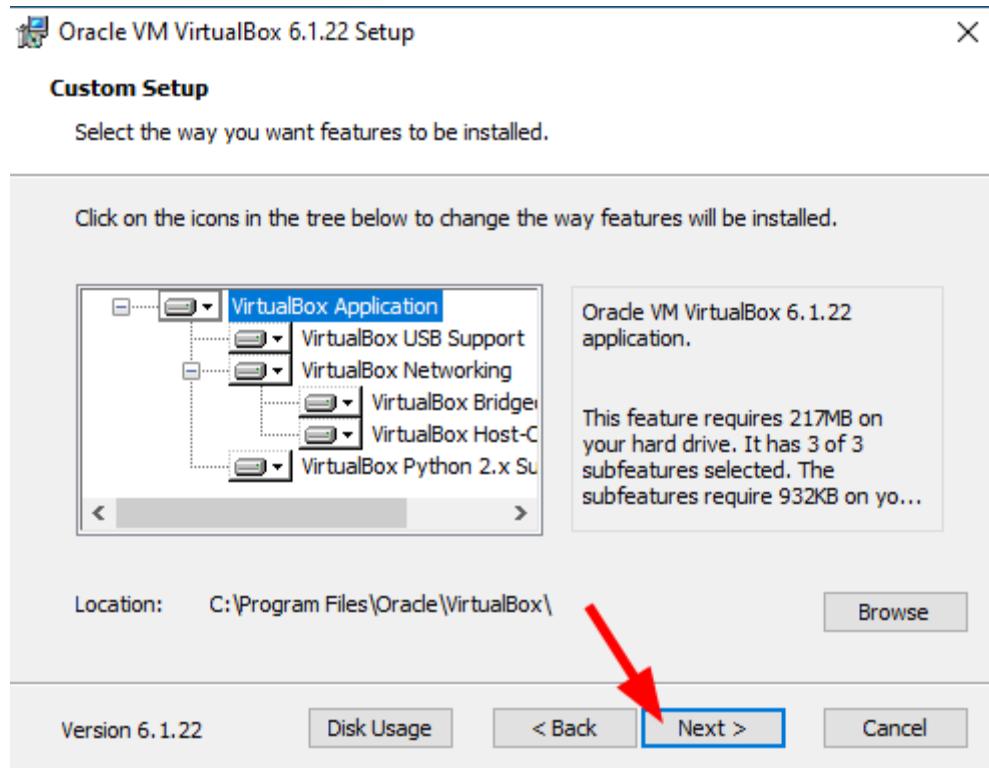


On the VirtualBox Welcome window, click the Next button

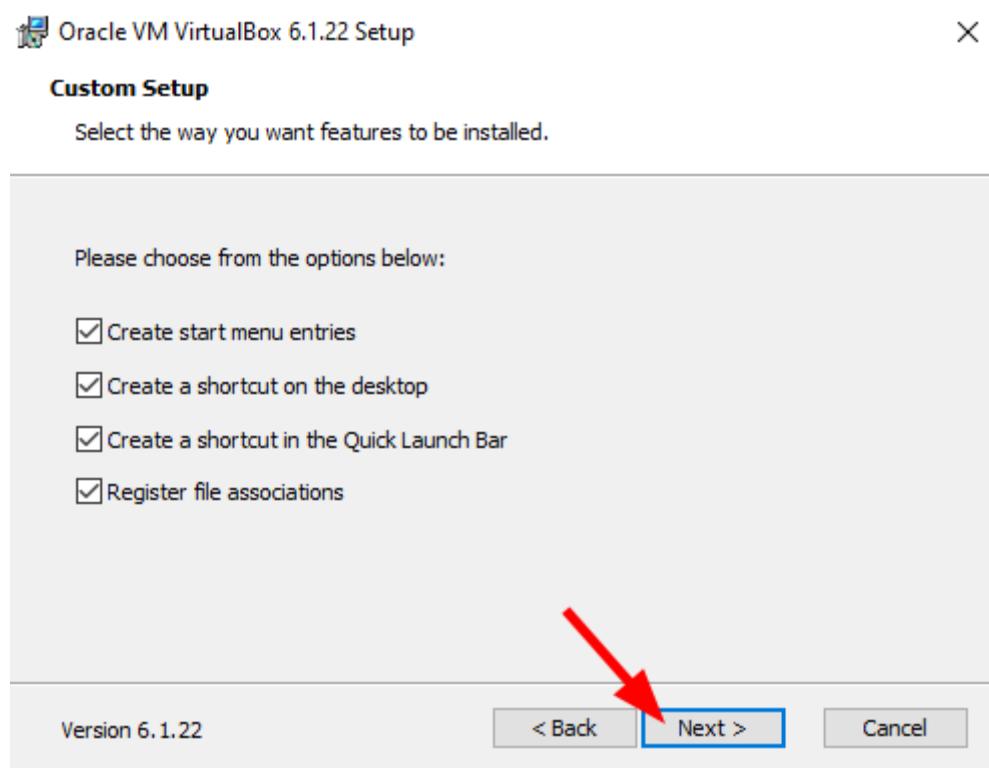


On the VirtualBox Customer Setup window, click the Next button

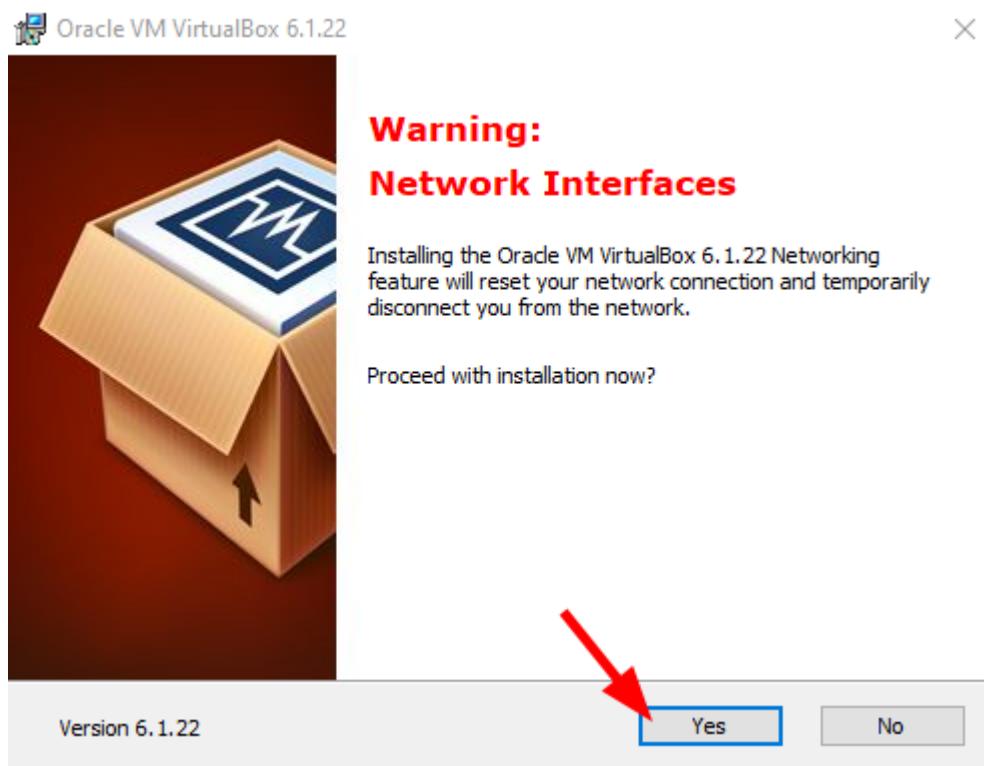
Alternatively, you can browse for a new location on your machine to install the VirtualBox or customize your installation further



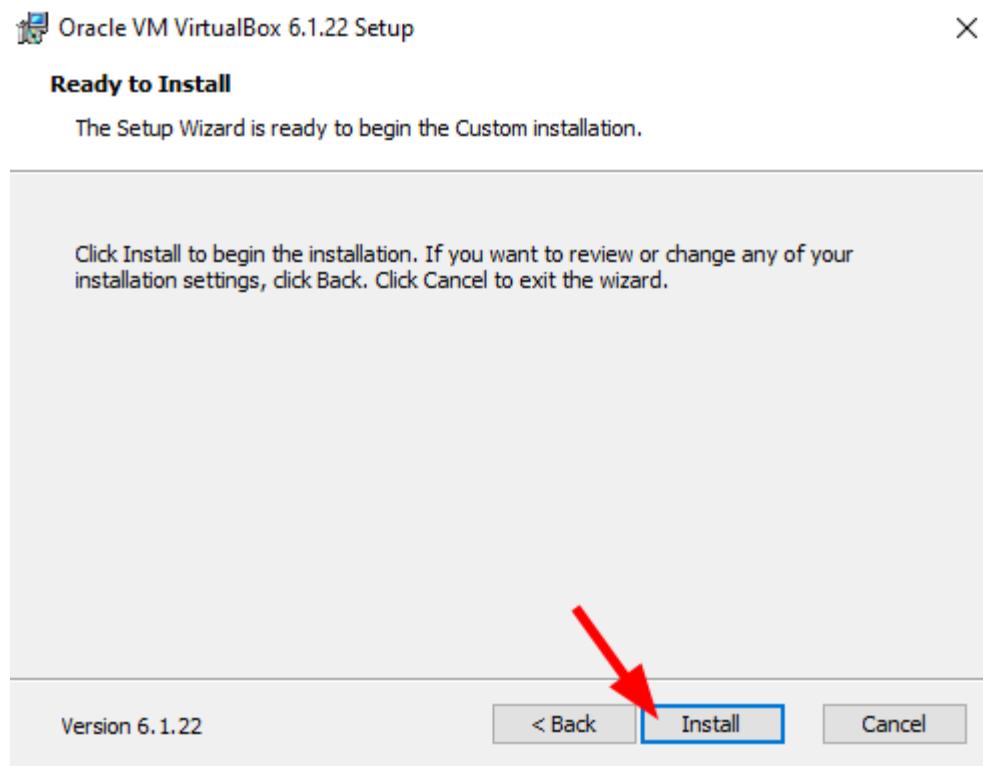
Check/uncheck the shortcut options or leave the default selection then click the Next button



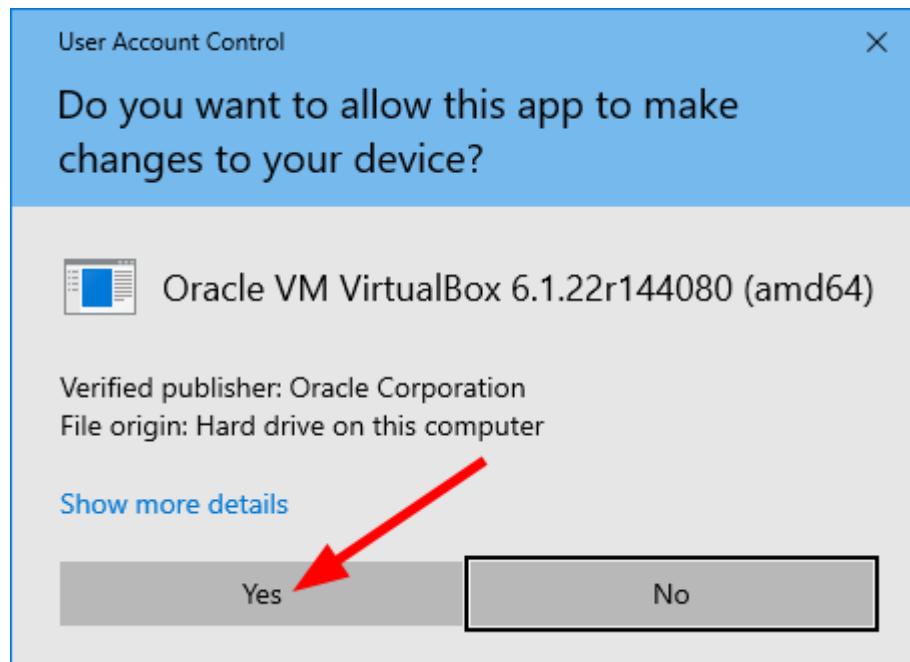
The VirtualBox installer will proceed to configure the virtual network interface on your VM therefore the network connection will reset shortly. Click Yes to continue the installation.



On the Ready to Install window click Install to start the VirtualBox installation.

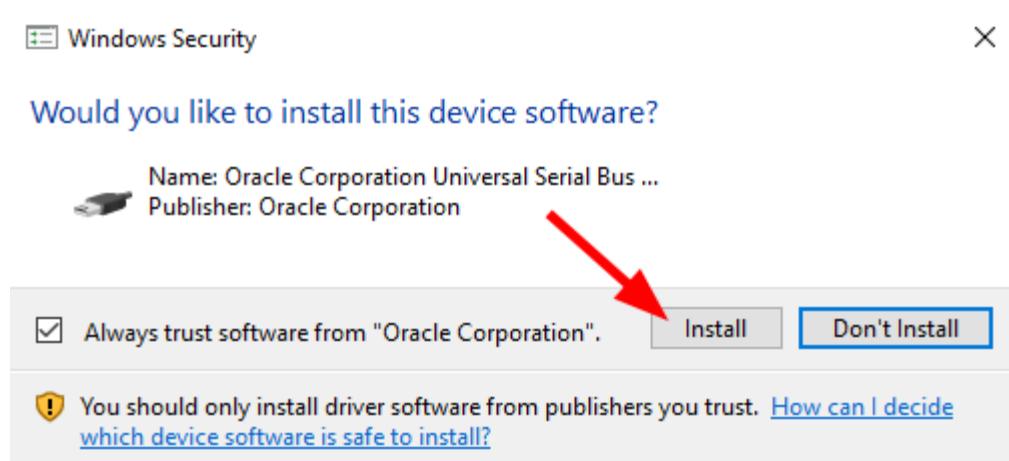


On the User Account Control window, click Yes [Figure 1.8].



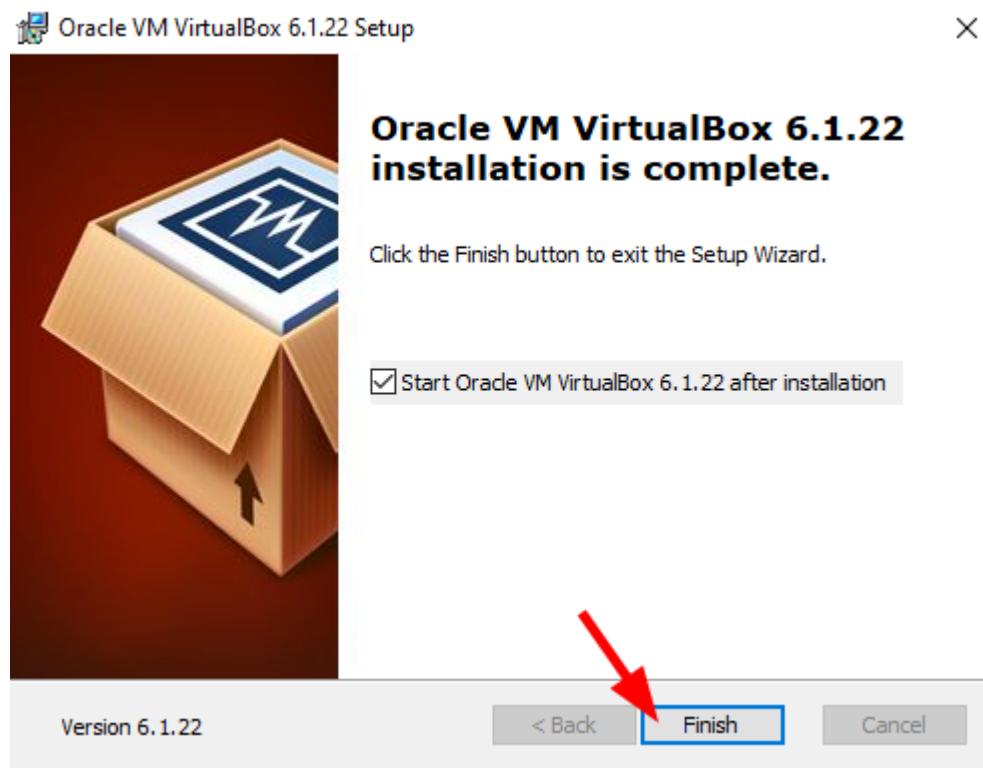
VirtualBox installation – User Account Control.

When prompted to allow the Oracle Corporation Universal Serial Bus controller, click the Install button



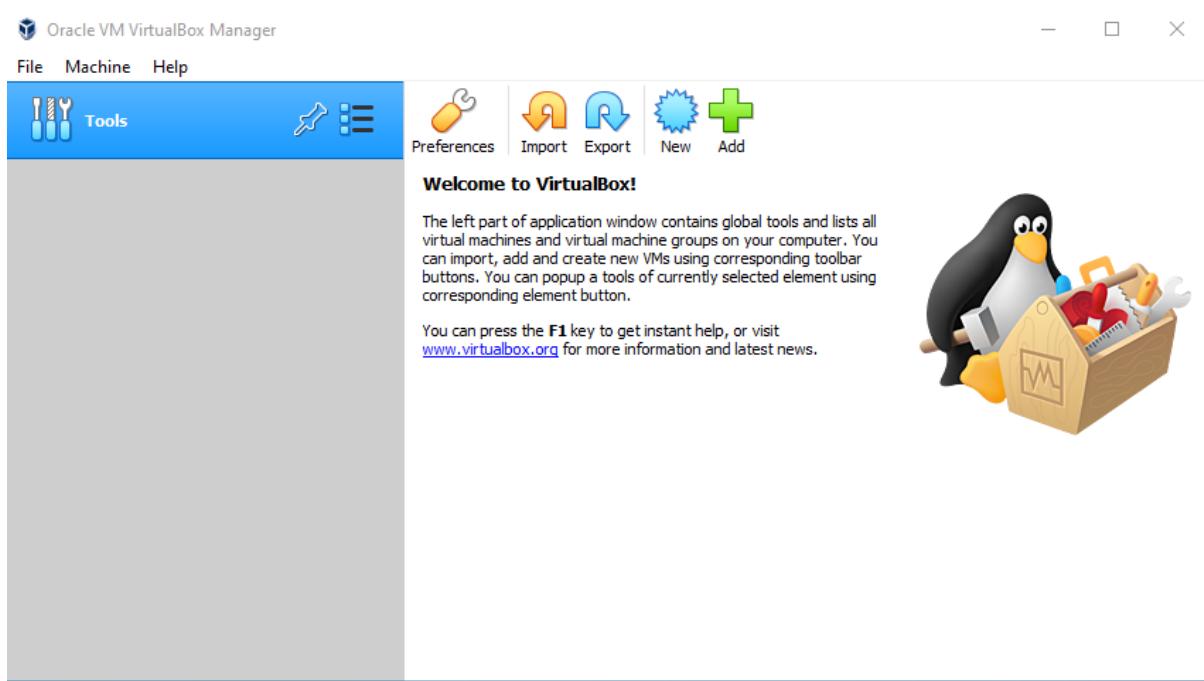
VirtualBox installation – Oracle virtual USB controller.

The VirtualBox installation is now completed. Click the Finish button to exit the installer



VirtualBox installation – Finish.

VirtualBox is now installed on your computer. You should see a window similar to the one in below.



## 2. Install Kali Linux in VirtualBox

Step 1: Download And Import Kali Linux VM In VirtualBox.

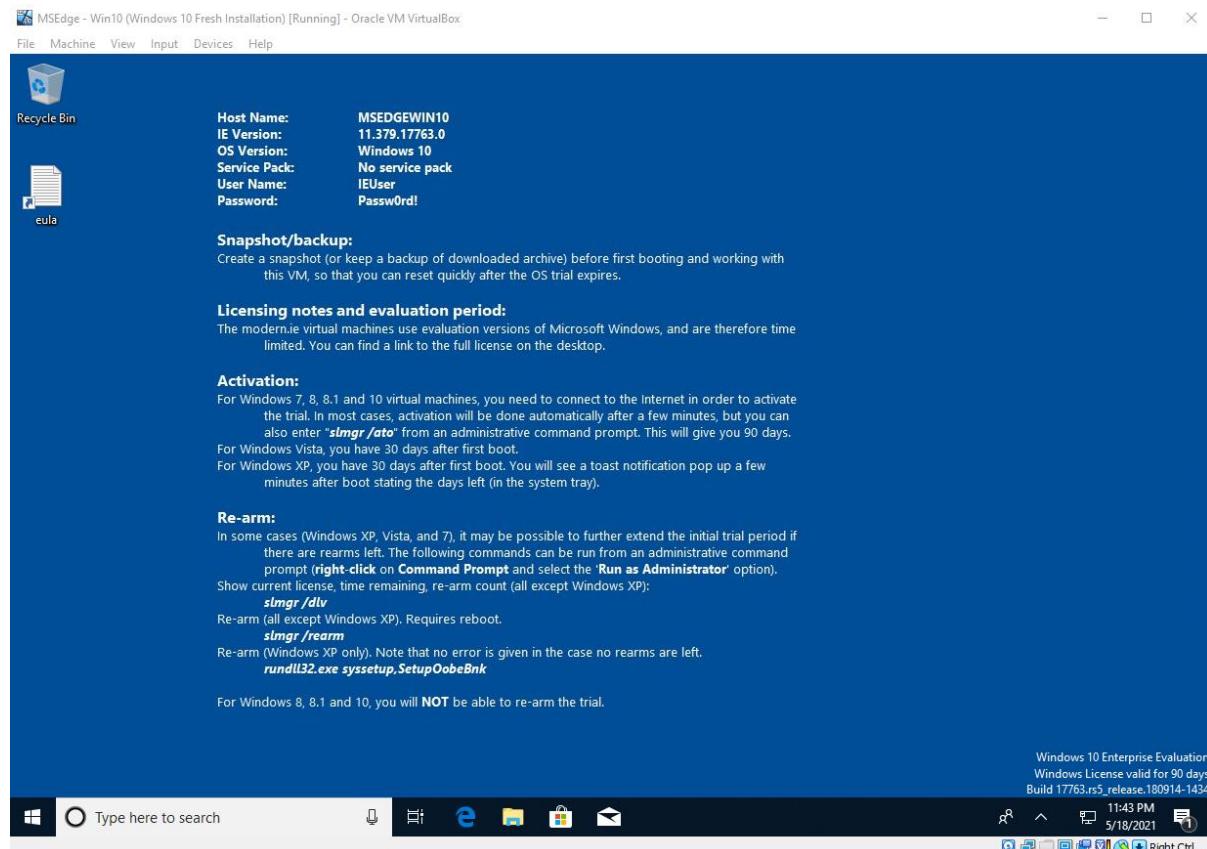
Step 2: Network Isolate Kali Linux VM

## 3. Install Windows 10 on VirtualBox

Step 1: Download Windows 10 VirtualBox VM

Step 2: Import Windows 10 VM in VirtualBox

Step 3: Boot into Windows 10 VM

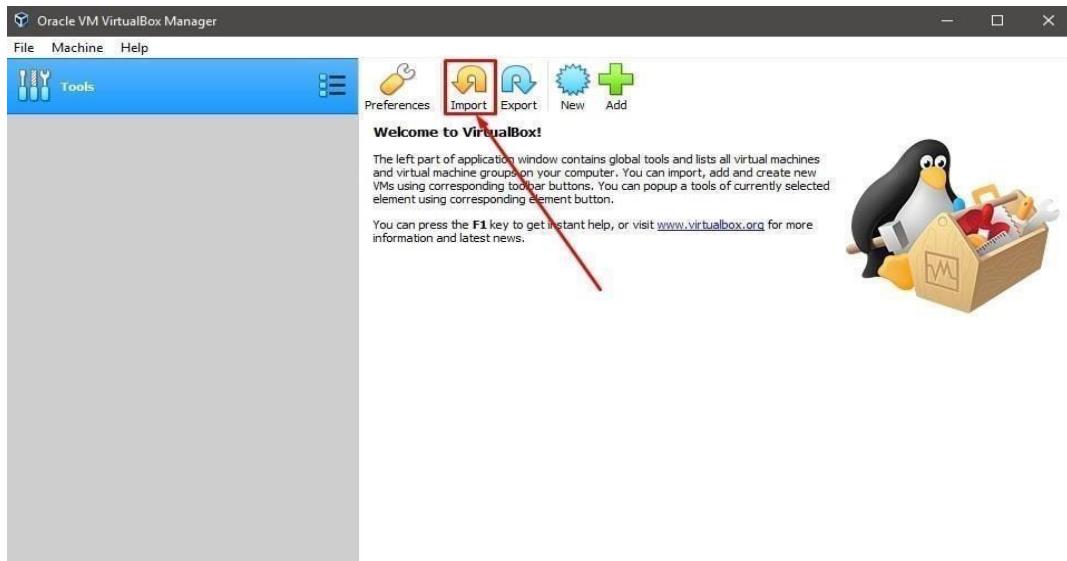


## Steps to install Kali Linux

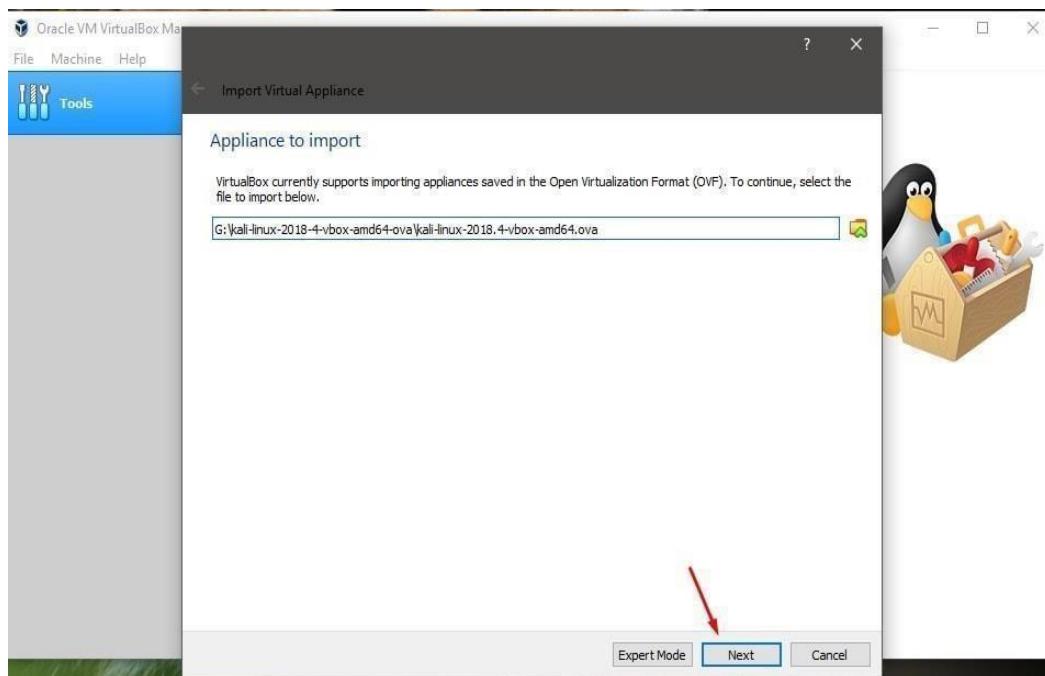
Step 1: Visit the Kali Linux official website and install kali linux OVA file.

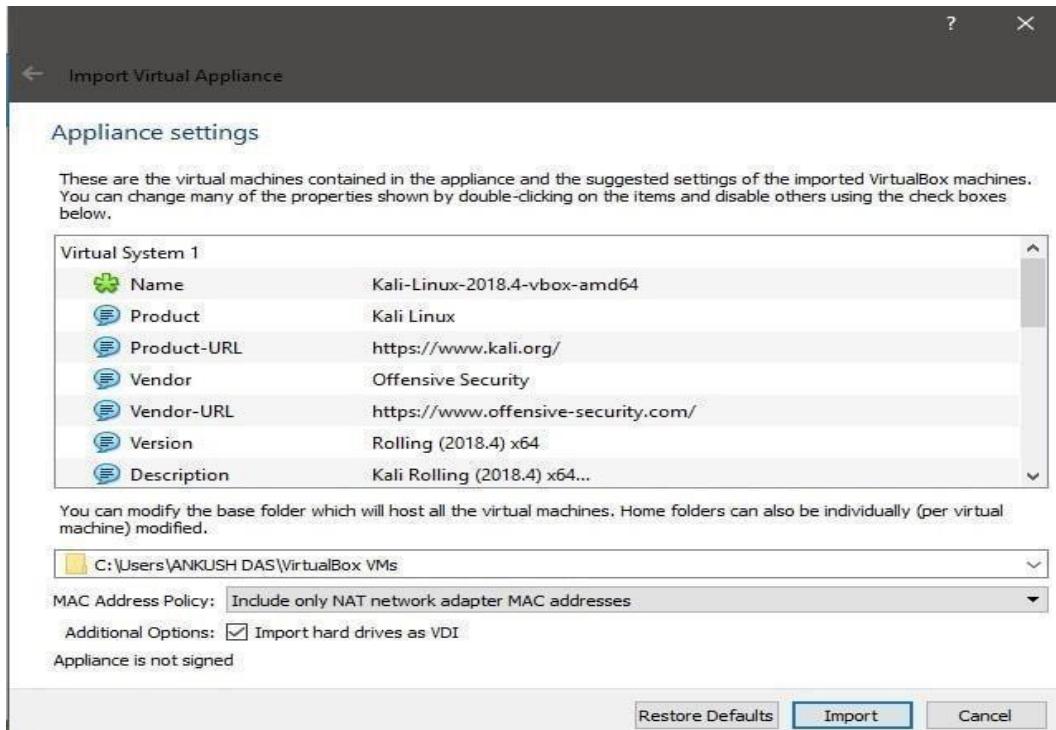
Image Name	Torrent	Size	Version	SHA256Sum
Kali Linux Vbox 32 Bit Ova	Torrent	3.6G	2018.4	2b28c5104f7936a57aed72dccb7e37c10923ca6e666cce5c9af14d9850a26e9
Kali Linux Vbox 64 Bit Ova	Torrent	3.6G	2018.4	88bc25f726cbbbe84a5a9375a91e4c675e18c016fdd2e0da8c38ee0744b3ae7e

Step 2: Open VirtualBox and click on import or File>Import Appliance.

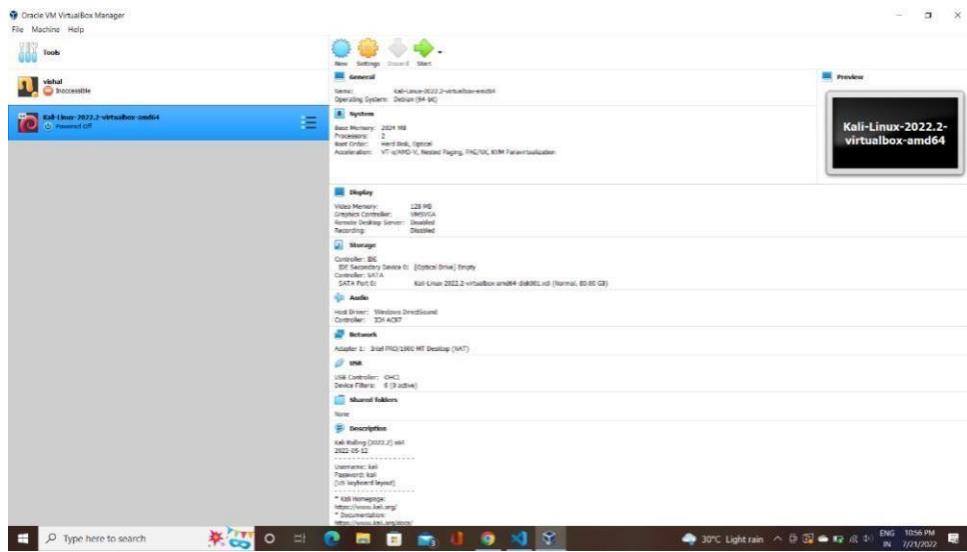


Step 3: Import the OVA file of kali Linux from disk file in the desired path.





Step 4: After getting install click on start button to get start with kali. The default username in Kali Linux used to be root and the default password was root. But since January 2020, Kali Linux is not using the root account. Now, the default account and password both are kali.



## **PRACTICAL -2**

**Aim : Information Gathering using NMAP framework and study about port scanning.**

### **Information Gathering**

An information-gathering process is the act of gathering information on a possible target. This might be done as part of penetration testing, network security monitoring, or other cybersecurity chores.

It is the first phase of ethical hacking that is to gather all the possible information about the target to find the weaknesses or vulnerabilities.

### **Nmap**

- Nmap is an open-source utility for network discovery.
- Network Mapper is a security auditing and network scanning independent tool developed by Gordon Lyon.
- It is used by network administrators to detect the devices currently running on the system and the port number by which the devices are connected.
- Many systems and network administrators are used for managing network inventory, service upgrade schedules, monitoring hosts and service uptime.
- At the top-level, Nmap is defined as a tool that can detect or diagnose services that are running on an Internet-connected system by a network administrator in their networked system used to identify potential security flaws. It is used to automate redundant tasks, such as monitoring the service.

### **Working of Nmap**

- Nmap is convenient during penetration testing of networked systems. Nmap provides the network details, and helps to determine the security flaws present in the system. Nmap is platform-independent and runs on popular operating systems such as Linux, Windows, and Mac.

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

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

## Features of Nmap

1. Host discovery: Nmap can identify live hosts on a network by sending a series of probes to different IP addresses and analyzing the responses.
2. Port scanning: Nmap can scan a range of TCP and UDP ports to identify open ports on a host, as well as the services running on those ports.
3. Operating system identification: Nmap can use various techniques to identify the operating system running on a target host, such as analyzing TCP/IP fingerprint data.
4. Service and version detection: Nmap can identify the services running on open ports, as well as the version information of those services.
5. Scriptable interaction: Nmap can be extended with custom scripts written in various languages, which can be used to automate certain tasks or customize the tool's behavior.
6. Output customization: Nmap provides various options for customizing the output of its scans, including different formats such as XML, grepable, and normal.
7. Performance tuning: Nmap provides various options for tuning the performance of its scans, such as setting the number of parallel probes and adjusting timeouts.
8. Stealth scanning: Nmap provides various options for stealth scanning, such as using decoy hosts and fragmenting packets, to avoid detection by intrusion detection systems (IDS) and firewalls.

## Commands

### 1) Ping scan -

Scans the list of devices up and running on a given subnet.

Here we ping scan mbitwebsite using the follwing command:

```
>>nmap -sP hostname
```

```
(jeritrix_2507@jeritrix)-[~]
$ nmap -sP www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 21:34 IST
Nmap scan report for www.flipkart.com (103.243.32.90)
Host is up (0.13s latency).
Other addresses for www.flipkart.com (not scanned): 64:ff9b::67f3:205a
Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds
```

## 2) Scan a single host –

The following syntax is used to scan to scan single host or ip address we can also give the ip of the website instead of host name

Use the asterisk (\*) to scan all of the subnets at once.

> nmap 192.164.1.\*

>> nmap hostname(or ip-address)

```
(jeritrix_2507@jeritrix)-[~]
$ nmap www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 22:53 IST
Nmap scan report for www.flipkart.com (163.53.76.86)
Host is up (0.025s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 4.68 seconds
```

## 3) Stealth Scan -

Stealth scanning is performed by sending an SYN packet and analyzing the response. If SYN/ACK is received, it means the port is open, and you can open a TCP connection. However, a stealth scan never completes the 3-way handshake, which makes it hard for the target to determine the scanning system(-sS, sW, sM, sA, sT). >>nmap -sS hostname

```
(root@kali)-[~]
# nmap -sS 202.129.241.235
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-28 09:42 EDT
Nmap scan report for icctw.ac.in (202.129.241.235)
Host is up (0.021s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 7.20 seconds
```

## 4) Version Scan –

This command is use to scan the version of the service running on the port.

>>nmap -sV hostname

```
(jeritrix_2507@jeritrix)-[~]
$ nmap -sV www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 22:57 IST
Nmap scan report for www.flipkart.com (163.53.76.86)
Host is up (0.026s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    open  tcpwrapped
443/tcp   open  ssl/https  nginx
1 service unrecognized despite returning data. If you know the service/version, please let us know!
SF-Port443-TCP:V=7.93%T=SSL%I=7%D=4/10%Time=64344704%P=x86_64-pc-linux-gnu
SF:<r>(GetRequest,E9,"HTTP/1.\.1\x20503\x20Service\x20Unavailable\r\nContent-S
SF:-length:\x20107\r\nCache-Control:\x20no-cache\r\nConnection:\x20close\r
SF:\nContent-Type:\x20text/html\r\n\r\n<html><body><h1>503\x20Service\x20U
SF:available</h1>\nNo\x20server\x20is\x20available\x20to\x20handle\x20thi
SF:s\x20request\.n</body></html>\n")%r(HTTPOptions,E9,"HTTP/1.\.1\x20503\x
SF:20Service\x20Unavailable\r\nContent-length:\x20107\r\nCache-Control:\x20
SF:0no-cache\r\nConnection:\x20close\r\nContent-Type:\x20text/html\r\n\r\n<
SF:<html><body><h1>503\x20Service\x20Unavailable</h1>\nNo\x20server\x20is\x
SF:x20available\x20to\x20handle\x20this\x20request\.n</body></html>\n")%r
SF:(FourOhFourRequest,E9,"HTTP/1.\.1\x20503\x20Service\x20Unavailable\r\nCo
SF:ntent-length:\x20107\r\nCache-Control:\x20no-cache\r\nConnection:\x20cl
SF:ose\r\nContent-Type:\x20text/html\r\n\r\n<html><body><h1>503\x20Service
SF:\x20Unavailable</h1>\nNo\x20server\x20is\x20available\x20to\x20handle\x
SF:20this\x20request\.n</body></html>\n");

Service detection performed. Please report any incorrect results at https://nmap.org/
Nmap done: 1 IP address (1 host up) scanned in 30.82 seconds
```

## 5) OS detection:

It enables OS detection of website. Now we need to run the actual commands to perform OS detection using NMAP, and at first, we will get the IP address of the host system, and then will perform a scan to get all active devices on the network.

>> nmap -O sitename

```
(root@kali)-[~]
# nmap -O mbit.edu.in
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-28 13:43 EDT
Nmap scan report for mbit.edu.in (202.129.241.235)
Host is up (0.012s latency).
rDNS record for 202.129.241.235: icctw.ac.in
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge|general purpose|switch
Running (JUST GUESSING): Oracle Virtualbox (99%), QEMU (96%), Bay Networks embedded (90%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu cpe:/h:baynetworks:bystack_450
Aggressive OS guesses: Oracle Virtualbox (99%), QEMU user mode network gateway (96%), Bay Networks BayStack
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.54 seconds
```

## 6) Port Scan:

We use this command to scan for the number of ports that are provided by the given site. A range of ports can be scanned by separating them with a hyphen.

```
> nmap -p 76–973 192.164.0.1
```

```
>>nmap –p sitename
```

```
└─(jeritrix_2507㉿jeritrix)-[~]
$ nmap -p 80 443 www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 23:00 IST
Nmap scan report for www.flipkart.com (163.53.76.86)
Host is up (0.028s latency).

PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 2 IP addresses (1 host up) scanned in 0.17 seconds
```

## 7) protocol scan -

This command helps scan the protocol used for the site which is important for hacking.

```
>>nmap –sO sitename
```

```
└─(root㉿kali)-[~]
# nmap -sO mbit.edu.in
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-28 14:10 EDT
Nmap scan report for mbit.edu.in (202.129.241.235)
Host is up (0.00095s latency).
rDNS record for 202.129.241.235: icctw.ac.in
Not shown: 255 open|filtered n/a protocols (no-response)
PROTOCOL STATE SERVICE
6          open  tcp

Nmap done: 1 IP address (1 host up) scanned in 7.21 seconds
```

## 8) ICMP Echo Request Ping (-PE) -

A ping command sends an ICMP(Internet Control Message Protocol) echo request to the target host. The target host responds with an echo reply. The ICMP echo request ping sends an ICMP echo request to the IP address of the destination machine. In the normal type of ICMP echo request, a combination of TCP and ACK pings is sent. Using

option -PE, the ICMP echo request can be specified as the nmap ping method without coupling TCP ACK ping.

>> nmap -PE www.flipkart.com

```
└─(jeritrix_2507@jeritrix)-[~]
$ nmap -PE www.flipkart.com
Warning: You are not root -- using TCP pingscan rather than ICMP
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 23:05 IST
Nmap scan report for www.flipkart.com (103.243.32.90)
Host is up (0.040s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 5.01 seconds
```

## 9) (-Pn) No Ping -

This option skips the host discovery stage altogether. Normally, Nmap uses this stage to determine active machines for heavier scanning and to gauge the speed of the network. By default, Nmap only performs heavy probing such as port scans, version detection, or OS detection against hosts that are found to be up. Disabling host discovery with -Pn causes Nmap to attempt the requested scanning functions against every target IP address specified.

>>nmap –Pn www.flipkart.com

```
└─(jeritrix_2507@jeritrix)-[~]
$ nmap -Pn 192.168.56.102
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 23:07 IST
Nmap scan report for 192.168.56.102 (192.168.56.102)
Host is up.
All 1000 scanned ports on 192.168.56.102 (192.168.56.102) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 201.41 seconds
```

## 10) (-PA) TCP ACK Ping -

Instead of using the default option of both an ICMP echo request and a TCP ACK, the –PA option sends a TCP ACK and forgoes any ICMP echo requests. This is a good alternative when the use of ICMP is not applicable because of packet filtering or firewalls.

>>nmap -PA [www.flipkart.com](http://www.flipkart.com)

```
[jeritrix_2507@jeritrix)-[~]
$ nmap -PA www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 23:15 IST
Nmap scan report for www.flipkart.com (163.53.76.86)
Host is up (0.030s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 7.42 seconds
```

## 11) UCP Scan (-sU) -

UDP services are widely deployed. Because UDP scanning is generally slower and more difficult than TCP, some security auditors ignore these ports. UDP scan is activated with the -sU option. It can be combined with a TCP scan type such as SYN scan (-sS) to check both protocols during the same run.

>> nmap -sU [www.flipkart.com](http://www.flipkart.com)

## 12) (-sA) TCP ACK scan -

This scan is different than the others discussed so far in that it never determines open (or even open|filtered) ports. It is used to map out firewall rulesets, determining whether they are stateful or not and which ports are filtered.

>>nmap -sA google.com

```
[root@kali)-[~]
# nmap -sA google.com
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-28 09:58 EDT
Nmap scan report for google.com (142.251.42.14)
Host is up (0.00088s latency).
Other addresses for google.com (not scanned): 2404:6800:4009:831::200e
rDNS record for 142.251.42.14: bom12s19-in-f14.1e100.net
All 1000 scanned ports on google.com (142.251.42.14) are in ignored states.
Not shown: 1000 unfiltered tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.85 seconds
```

## 13) (-sW) TCP Window scan -

Window scan is exactly the same as ACK scan except that it exploits an implementation detail of certain systems to differentiate open ports from closed ones, rather than always printing unfiltered when a RST is returned.

```
Nmap scan report for mbit.edu.in (202.129.241.235)
Host is up (0.00059s latency).
rDNS record for 202.129.241.235: icctw.ac.in
All 1000 scanned ports on mbit.edu.in (202.129.241.235) are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.54 seconds
```

#### 14) (-sZ) SCTP COOKIE ECHO scan -

SCTP COOKIE ECHO scan is a more advanced SCTP scan. It takes advantage of the fact that SCTP implementations should silently drop packets containing COOKIE ECHO chunks on open ports, but send an ABORT if the port is closed

>> nmap -sZ google.com

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-28 16:00 EDT
Stats: 0:00:04 elapsed; 0 hosts completed (1 up), 1 undergoing SCTP COOKIE-ECHO Scan
SCTP COOKIE-ECHO Scan Timing: About 59.62% done; ETC: 16:01 (0:00:03 remaining)
Nmap scan report for mbit.edu.in (202.129.241.235)
Host is up (0.0013s latency).
rDNS record for 202.129.241.235: icctw.ac.in
All 52 scanned ports on mbit.edu.in (202.129.241.235) are in ignored states.
Not shown: 52 open|filtered sctp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 9.11 seconds
```

#### 15) Script Summary

Gathers information from an IRC server. It uses STATS, LUSERS, and other queries to obtain this information.

>>nmap -sV -sC [www.flipkart.com](http://www.flipkart.com)

```
└─(jeritrix_2507@jeritrix)-[~]
$ nmap -sV -sC www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 23:25 IST
Nmap scan report for www.flipkart.com (103.243.32.90)
Host is up (0.038s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    open  tcpwrapped
443/tcp   open  tcpwrapped
| ssl-cert: Subject: commonName=www.flipkart.com
| Subject Alternative Name: DNS:www.flipkart.com, DNS:flipkart.com, DNS:bhaskar.store.flipkart.com
| .flipkart.com, DNS:indusind.store.flipkart.com, DNS:offers.store.flipkart.com, DNS:axis.store.flip
| Not valid before: 2023-03-30T09:27:42
|_Not valid after: 2023-07-04T07:50:53
|_ssl-date: TLS randomness does not represent time

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 31.12 seconds
```

16) OS detection:

Enables OS detection and services, script scanning and tracerouting.

>>nmap -www.flipkart.com

```
└──(jeritrix_2507㉿jeritrix)-[~]
└─$ nmap -A www.flipkart.com
Starting Nmap 7.93 ( https://nmap.org ) at 2023-04-10 23:28 IST
Nmap scan report for www.flipkart.com (103.243.32.90)
Host is up (0.040s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    open  tcpwrapped
443/tcp   open  tcpwrapped
| ssl-cert: Subject: commonName=www.flipkart.com
| Subject Alternative Name: DNS:www.flipkart.com, DNS:flipkart.com, DNS:bhaskar.store.flipkart.co
| .flipkart.com, DNS:indusind.store.flipkart.com, DNS:offers.store.flipkart.com, DNS:axis.store.fli
| Not valid before: 2023-03-30T09:27:42
|_.Not valid after: 2023-07-04T07:50:53
|_ssl-date: TLS randomness does not represent time
|_http-server-header: nginx

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 33.41 seconds
```

## **PRACTICAL -3**

**Aim : Understand packet capturing tool Wireshark or Ethercap and analysis of those packets.**

### **Packet Capturing**

- Packet capture is a vital tool used to keep networks operating safely and efficiently.
- In the wrong hands, it can also be used to steal sensitive data like usernames and passwords. In this post, we will dive into what a packet capture is, how it works, what kind of tools are used, and look at some sample use cases.
- Packet Capture refers to the action of capturing Internet Protocol (IP) packets for review or analysis.
- The term can also be used to describe the files that packet capture tools output, which are often saved in the .pcap format.
- Capturing packets is a common troubleshooting technique for network administrators, and is also used to examine network traffic for security threats.
- Following a data breach or other incident, packet captures provide vital forensic clues that aid investigations.
- From a threat actor's perspective, packet captures might be used to steal passwords and other sensitive data.
- Unlike active reconnaissance techniques like port scanning, capturing packets can be accomplished without leaving any trace behind for investigators.

### **Wireshark**

- Wireshark is a network protocol analyzer, or an application that captures packets from a network connection, such as from your computer to your home office or the internet.
- Packet is the name given to a discrete unit of data in a typical Ethernet network.
- Wireshark is a packet sniffer and analysis tool. It captures network traffic on the local network and stores that data for offline analysis.
- Wireshark captures network traffic from Ethernet, Bluetooth, Wireless (IEEE.802.11), Token Ring, Frame Relay connections, and more. (A "packet" is a single message from any network protocol).

Wireshark is the most often-used packet sniffer in the world. Like any other packet sniffer, Wireshark does three things:

1. **Packet Capture:** Wireshark listens to a network connection in real time and then grabs entire streams of traffic – quite possibly tens of thousands of packets at a time.
2. **Filtering:** Wireshark is capable of slicing and dicing all of this random live data using filters. By applying a filter, you can obtain just the information you need to see.
3. **Visualization:** Wireshark, like any good packet sniffer, allows you to dive right into the very middle of a network packet. It also allows you to visualize entire conversations and network streams.

## Working of Wireshark

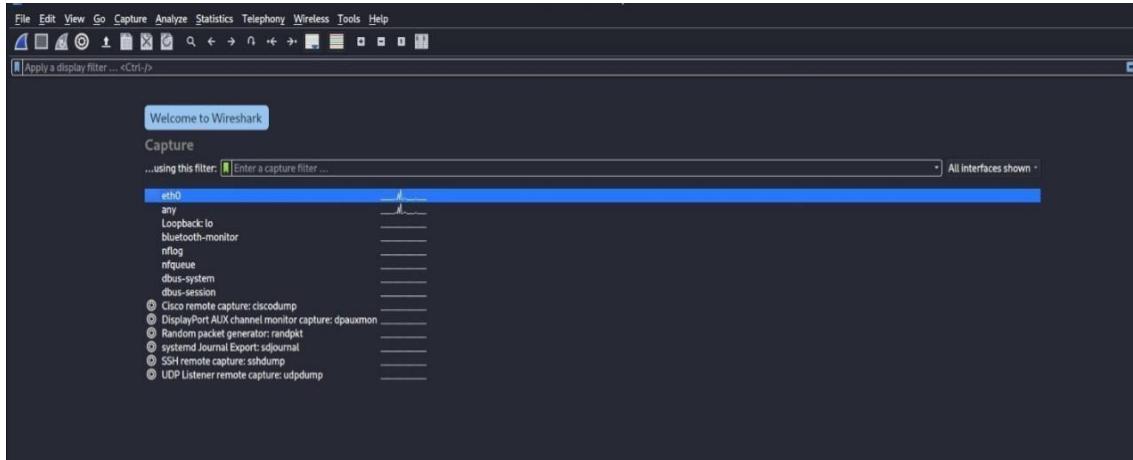
The packets in the Wireshark are highlighted with blue, black, and green color. These colors help users to identify the types of traffic. It is also called as packet colorization. The kinds of coloring rules in the Wireshark are temporary rules and permanent rules.

- The temporary rules are there until the program is in active mode or until we quit the program.
- The permanent color rules are available until the Wireshark is in use or the next time you run the Wireshark. The steps to apply color filters will be discussed later in this topic.

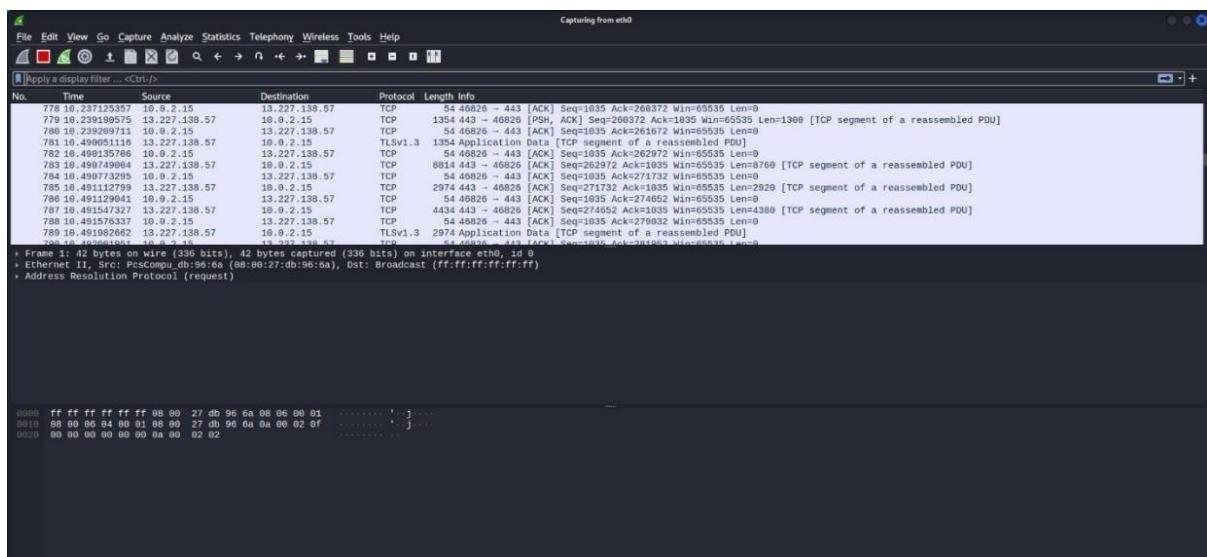
It works on 3-way Handshake rule,

- When you are capturing your data, analyze the problem, you will get the three-way handshake.
- It contains good options like the TCP options.
- From this, you can determine the shift time and figure out if you have captured packets on the client-side or the server-side.

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



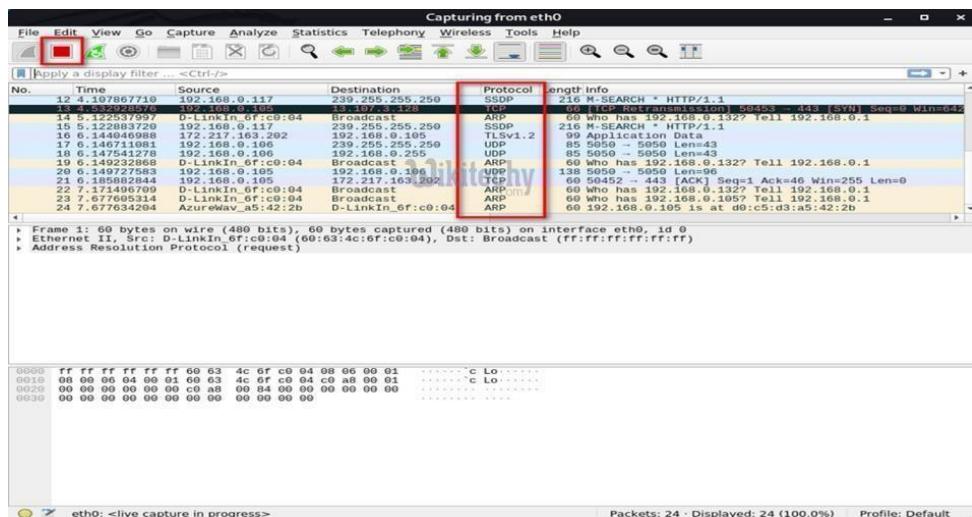
- Choose eth0, which is used for monitoring network such as protocols, sessions , packsand then select the shark tail symbol or double tap the eth0.
- Now Capturing will start which shows sender and receiver IP address, it means trace router root, protocols and response time in milli seconds.



- The above arrow shows the packet content written in hexadecimal or the ASCII format. And the information above the packet content, are the details of the packet header.
- It will continue listening to all the data packets, and you will get much data. If you want to see a particular data, then you can click on the red button. The traffic will be stationary, and you can note the parameters like time, source,

destination, the protocol being used, length, and the Info. To view in-depth detail, you can click on that particular address; a lot of the information will be displayed below that.

- Now Capturing will start which shows sender and receiver IP address, it means trace router root, protocols, and response time in milli seconds.



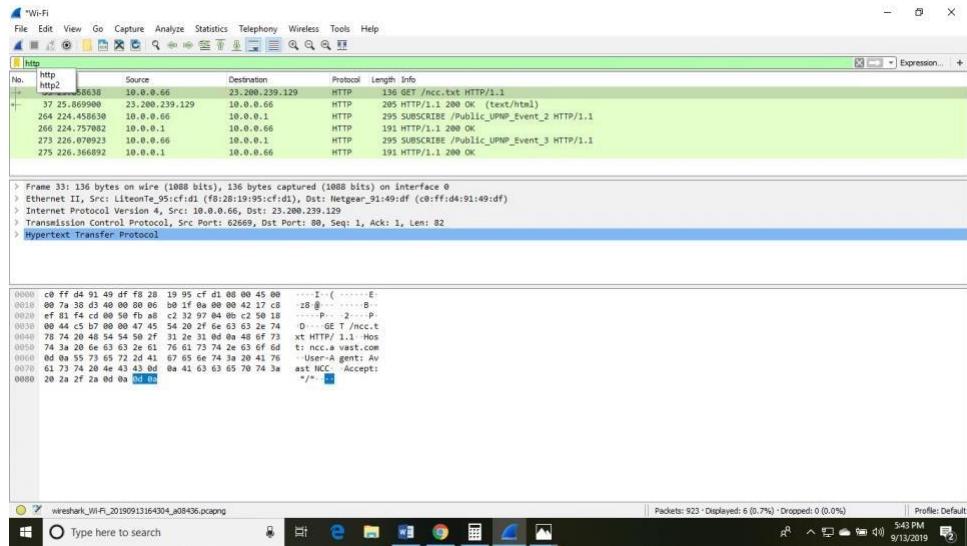
Wireshark shows you three different panes for inspecting packet data.

The Packet List, the top pane, is a list of all the packets in the capture. When you click on a packet, the other two panes change to show you the details about the selected packet.

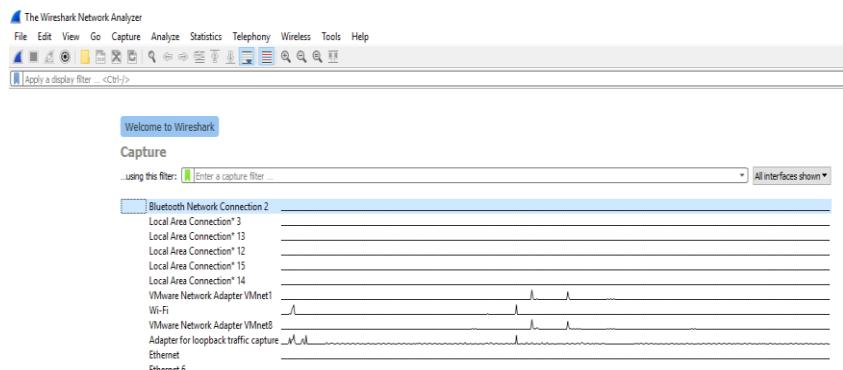
You can also tell if the packet is part of a conversation. Here are some details about each column in the top pane:

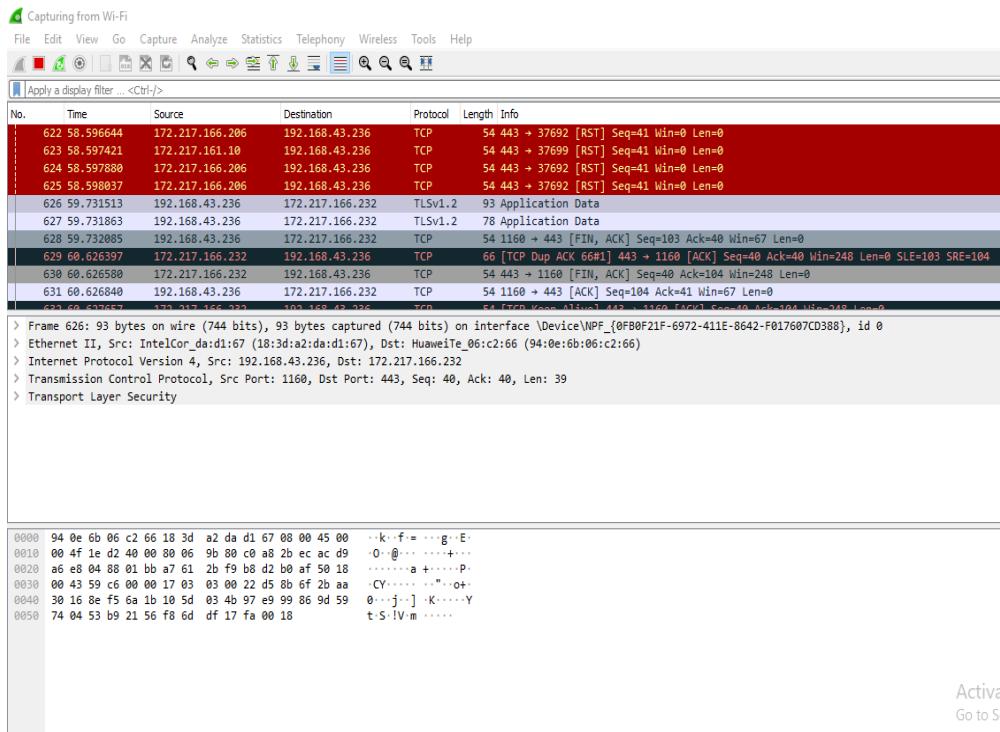
- No.: This is the number order of the packet that got captured. The bracket indicates that this packet is part of a conversation.
- Time: This column shows you how long after you started the capture that this packet got captured. You can change this value in the Settings menu if you need something different displayed.
- Source: This is the address of the system that sent the packet.
- Destination: This is the address of the destination of that packet.
- Protocol: This is the type of packet, for example, TCP, DNS, DHCPv6, or ARP.
- Length: This column shows you the length of the packet in bytes.
- Info: This column shows you more information about the packet contents, and will vary depending on what kind of packet it is.

- There is a filter block below the menu bar, from where a large amount of data can be filtered. For example, if we apply a filter for HTTP, only the interfaces with the HTTP will be listed



- For example if we open the adit localhost website of IP address 10.0.0.1 of port number 8090 and if we do the login with user name and password than it is captured in wireshark as it captures the packet so when we open the wireshark and filter it with http we can see a login http session so we right click on follow the stream than it can show the username and password which I entered in adit website. Now click the Html FormURL Encoded which is for user credentials are stored in URL encoded tag, then http web page login details are displayed.





## **PRACTICAL -4**

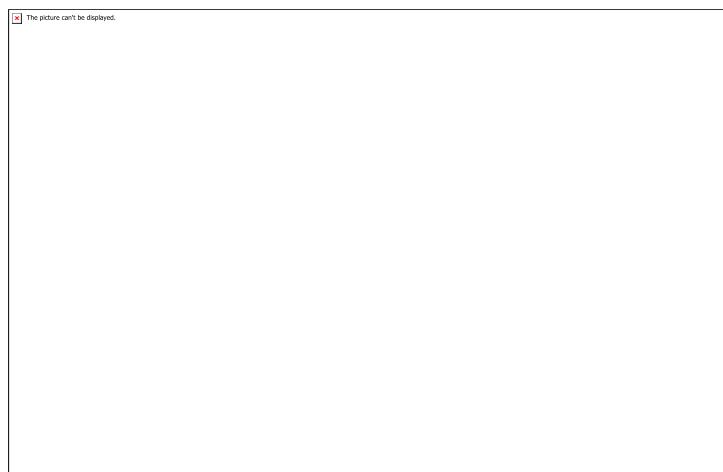
**Aim : Using open port information perform MITM(Man In The Middle) attack using arpspoof, urlsnarf, dsniff, dnsspoof.**

**1. Interruption**

**2. Interception**

### **Man in the Middle Attack (MITM)**

- A man in the middle (MITM) attack is a general term for when a perpetrator positions himself in a conversation between a user and an application—either to eavesdrop or to impersonate one of the parties, making it appear as if a normal exchange of information is underway.
- The goal of an attack is to steal personal information, such as login credentials, account details and credit card numbers.
- Targets are typically the users of financial applications, SaaS businesses, e-commerce sites and other websites where logging in is required.



- These attacks are carried out through a two-step process known as data interception and decryption.
- Data interception entails an attacker intercepting a data transfer between a client and a server.
- The attacker tricks the client and the server into believing that they are exchanging information with each other, while the attacker intercepts the data,

creates a connection to the real site and acts as a proxy to read and insert false information into the communication.

## 1] Working of Interruption in MITM

- It is the first step in the MITM attack.
- 1. When two parties communicate over a network, their communication is typically encrypted to ensure that only the intended recipient can read the data.
- 2. However, if an attacker is able to intercept the communication, they can decrypt the data and read it themselves.
- 3. In addition to reading the data, the attacker can also manipulate it before forwarding it on to the intended recipient. This manipulation can involve inserting malicious code or altering the data to achieve a specific outcome.
- 4. To the recipient, the communication appears to be normal, but in reality, it has been tampered with by the attacker.
- 5. The attacker can also inject their own data into the communication, posing as one of the legitimate parties. This can allow the attacker to gain access to sensitive information, such as login credentials or financial data.
- 6. The goal of the attacker is to remain undetected and to manipulate the communication without either party realizing that anything is wrong.
- 7. To protect against interruption-based MitM attacks, it's important to use secure communication protocols, such as HTTPS or SSL/TLS, and to avoid using unsecured public Wi-Fi networks. Additionally, using digital certificates and verifying the authenticity of SSL/TLS connections can help prevent MitM attacks.

In an interruption attack, a network service is made degraded or unavailable for legitimate use. They are the attacks against the availability of the network.

Examples of Interruption attacks :

- Overloading a server host so that it cannot respond.
- Cutting a communication line.
- Blocking access to a service by overloading an intermediate network or network device.
- Redirecting requests to invalid destinations.
- Theft or destruction of software or hardware involved.

## 2] Working of Interception

1. In a MitM attack, the attacker positions themselves between the two parties communicating on a network, intercepting the data being sent between them.
2. The attacker can intercept data through various means, such as Wi-Fi spoofing, DNS spoofing, and ARP spoofing.
3. Once the attacker has intercepted the communication, they can view the data being transmitted, including any sensitive information that the parties are exchanging.
4. The attacker can also manipulate the data or inject their own data into the communication, posing as one of the legitimate parties.
5. The goal of the attacker is to remain undetected and to manipulate the communication without either party realizing that anything is wrong.
6. Interception-based MitM attacks are a serious threat, especially in situations where sensitive information is being transmitted, such as online banking or e-commerce transactions.
7. To protect against interception-based MitM attacks, it's important to use secure communication protocols, such as HTTPS or SSL/TLS, and to avoid using unsecured public Wi-Fi networks. Additionally, using digital certificates and verifying the authenticity of SSL/TLS connections can help prevent MitM attacks.

An interception is where an unauthorized individual gains access to confidential or private information. Interception attacks are attacks against network the confidentiality objective of the CIA Triad.

Examples of Interception attacks:

1. Eavesdropping on communication.
2. Wiretapping telecommunications networks.
3. Illicit copying of files or programs.
4. Obtaining copies of messages for later replay.
5. Packet sniffing and key logging to capture data from a computer system or network.

Mitigate the attack :

**Using Encryption** - SSL, VPN, 3DES, BPI+ are deployed to encrypts the flow of information from source to destination so that if someone can snoop in on the flow of traffic, all the person will see is ciphered text.

**Traffic Padding** - It is a function that produces cipher text output continuously, even in the absence of plain text. A continuous random data stream is generated. When plaintext is available, it is encrypted and transmitted. When input plaintext is not present, the random data are encrypted and transmitted. This makes it impossible for an attacker to distinguish between tree data flow and noise and therefore impossible to deduce the amount of traffic.

## ARP Spoof

- The actual ARP poisoning attack is redirecting the flow of packets and making it flow through our device.
- Here, we use a tool called arpspoof, which is part of the suite called **dsniff**. This suite contains several programs that can be used to launch MITM attacks.
- Here, we see how to use arpspoof tool to carry out ARP poisoning, which redirects the flow of packets through our device.
- Windows is the target device, and we are going to the ARP table.
- So, we will run **arp -a** on the Windows machine to see the ARP table.
- In the following screenshot, we can see that the IP address for the access point is 10.0.0.1, and we can see its MAC address is c0-ff-d4-91-49-df. It is stored in the ARP table:

Interface: 10.0.0.62 --- 0x7		
Internet Address	Physical Address	Type
10.0.0.1	c0-ff-d4-91-49-df	dynamic
10.0.0.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

- We are connected now to the target network.
- We are going to use a tool arpspoof -i to choose our internet card which is wlan0.
- Then we are going to put the IP address of the target Window device which is 10.0.0.62.
- Then we are going to put the IP address for the access point, which is 10.0.0.1.
- We will tell the access point that the client IP address has our MAC address, so basically, we're going to tell the access point that we are the target client:

```
root@kali:~# arpspoof -i wlan0 -t 10.0.0.62 10.0.0.1
10:f0:5:87:19:32 b0:fc:36:6b:11:39 0806 42: arp reply 10.0.0.1 is-at 10:f0:5:87:19:32
10:f0:5:87:19:32 b0:fc:36:6b:11:39 0806 42: arp reply 10.0.0.1 is-at 10:f0:5:87:19:32
10:f0:5:87:19:32 b0:fc:36:6b:11:39 0806 42: arp reply 10.0.0.1 is-at 10:f0:5:87:19:32
10:f0:5:87:19:32 b0:fc:36:6b:11:39 0806 42: arp reply 10.0.0.1 is-at 10:f0:5:87:19:32
```

- After this, we're going to run arpspoof again, and instead of telling the access point that we are the target client, we are going to tell the client that we are the access point, so we're just going to flip the IPs:

```
root@kali:~# arpspoof -i wlan0 -t 10.0.0.1 10.0.0.62
10:f0:5:87:19:32 c0:ff:d4:91:49:df 0806 42: arp reply 10.0.0.62 is-at 10:f0:5:87:19:32
10:f0:5:87:19:32 c0:ff:d4:91:49:df 0806 42: arp reply 10.0.0.62 is-at 10:f0:5:87:19:32
10:f0:5:87:19:32 c0:ff:d4:91:49:df 0806 42: arp reply 10.0.0.62 is-at 10:f0:5:87:19:32
10:f0:5:87:19:32 c0:ff:d4:91:49:df 0806 42: arp reply 10.0.0.62 is-at 10:f0:5:87:19:32

C:\Users\jtp>arp -a

Interface: 10.0.0.62 --- 0x7
    Internet Address          Physical Address      Type
    10.0.0.1                  10-f0-05-87-19-32  dynamic
    10.0.0.11                 10-f0-05-87-19-32  dynamic
    10.0.0.255                ff-ff-ff-ff-ff-ff  static
    224.0.0.22                 01-00-5e-00-00-16  static
    224.0.0.251                01-00-5e-00-00-fb  static
    224.0.0.252                01-00-5e-00-00-fc  static
    239.255.255.250            01-00-5e-7f-ff-fa  static
    255.255.255.255            ff-ff-ff-ff-ff-ff  static
```

- Now, we are going to enable the IP forwarding.
- We do that so that when the packets flow through our device, they do not get dropped so that each packet that goes through our device gets actually forwarded to its destination.
- So, when we get a packet from the client, it goes to the router, and when a packet comes from the router, it should go to the client without being dropped in our device.
- So, we are going to enable it using this command:

```
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
```

- The window device now thinks that the attacker device is the access point, and whenever the window device tries to communicate with the access point, it is going to send all these requests to the attacker device.
- This will place our attacker device in the middle of the connection, and we will be able to read all the packets, modify them, or drop them.

## URLsnarf

- "URL snarf" is a computer network tool used to intercept and log URLs (Uniform Resource Locators) that are transmitted over a network.
- The tool is typically used by network administrators or security professionals to monitor network traffic and detect potential security threats.

- URL snarf works by capturing the data packets that are transmitted over the network and then filtering out any URLs contained within those packets.
- The tool then logs those URLs, providing the network administrator with a record of the web pages that were accessed by network users.
- While URL snarf can be a useful tool for monitoring network activity, it can also be used for malicious purposes.
- For example, an attacker could use URL snarf to intercept login credentials or other sensitive information that is transmitted over the network.
- It is important to note that using URL snarf without permission from the network owner is illegal and unethical.
- It is always best to obtain permission before using any network monitoring tools, and to use them only for legitimate purposes.

## Dsniff

- "Dsniff" is a computer network tool that is used to intercept and analyze network traffic in real-time.
- It is commonly used by network administrators and security professionals to monitor network activity, detect potential security threats, and analyze network performance.
- Dsniff can be used to capture a wide range of network traffic, including emails, passwords, web pages, and other network data.
- The tool uses a variety of techniques to intercept this traffic, including ARP spoofing, DNS spoofing, and packet sniffing.
- Once it has intercepted network traffic, dsniff can be used to analyze and decode the captured data, allowing network administrators to identify potential security threats or performance issues.
- The tool can also be used to generate reports on network activity, making it easier for administrators to understand and manage their network infrastructure.
- While dsniff can be a useful tool for network monitoring and analysis, it can also be used for malicious purposes.
- For example, an attacker could use dsniff to intercept sensitive information, such as login credentials or credit card numbers, that is transmitted over the network.
- As with any network monitoring tool, it's important to use dsniff responsibly and with permission from the network owner. Using dsniff or any other network monitoring tool without permission is illegal and unethical.

## Dnsspoof

- "DNSpoof" is a computer network tool that is used to intercept and modify DNS (Domain Name System) requests and responses.
- The tool is commonly used by network administrators and security professionals to monitor and control network traffic, detect potential security threats, and enforce network policies.
- DNSpoof works by intercepting DNS requests and responses and modifying them to redirect traffic to a different IP address.
- This can be useful for a variety of purposes, such as blocking access to malicious websites, redirecting traffic to a different server for load balancing, or enforcing network policies.
- For example, a network administrator could use DNSpoof to block access to a known malicious website by intercepting DNS requests for that website and redirecting them to a local web server that displays a warning message instead of the actual website.
- While DNSpoof can be a useful tool for network administration and security, it can also be used for malicious purposes. For example, an attacker could use DNSpoof to redirect users to a fake website that mimics a legitimate website in order to steal login credentials or other sensitive information.
- It is important to note that using DNSpoof without permission from the network owner is illegal and unethical.
- It is always best to obtain permission before using any network monitoring or control tools, and to use them only for legitimate purposes.

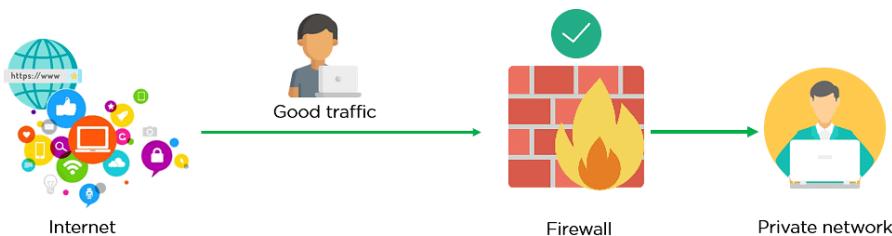
## **PRACTICAL -5**

**Aim : Understand the concept of firewall and configure the Statefull Packet Inspection(SPI) firewall IPTABLES.**

### **Firewall**

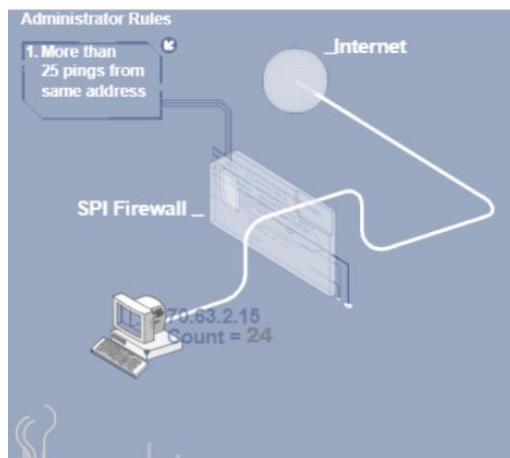
- A firewall can be defined as a special type of network security device or a software program that monitors and filters incoming and outgoing network traffic based on a defined set of security rules.
- It acts as a barrier between internal private networks and external sources (such as the public Internet).
- The primary purpose of a firewall is to allow non-threatening traffic and prevent malicious or unwanted data traffic for protecting the computer from viruses and attacks.
- A firewall is a cybersecurity tool that filters network traffic and helps users block malicious software from accessing the Internet in infected computers.

### **Working of Firewall**



- A firewall system analyzes network traffic based on pre-defined rules. It then filters the traffic and prevents any such traffic coming from unreliable or suspicious sources.
- It only allows incoming traffic that is configured to accept.
- Typically, firewalls intercept network traffic at a computer's entry point, known as a port. Firewalls perform this task by allowing or blocking specific data packets (units of communication transferred over a digital network) based on pre-defined security rules.
- Incoming traffic is allowed only through trusted IP addresses, or sources.

## Statefull packet Inspection



- Stateful Packet Inspection (SPI) Firewalls, also known as dynamic packet filtering, improves static packet filtering by using a "state table" to keep track of legitimate Internet service requests.
- SPI firewalls work by only allowing data through that comes from known and active connections. Any other data is rejected.
- SPI firewalls record attributes including the requesting (egress) client's IP and port addresses, handshake (SYN-ACK-ACK) statuses, and egress routes. This group of attributes is collectively known as the state of the connection.
- This firewall matches ingress traffic against these outgoing requests to ensure that each response is expected.
- Stateful packet inspectors typically have the same features as static packet filters, but they are able to view more of the network packet to determine whether to allow or block traffic.
- Again, ingress packets that do not meet the criteria established by the SPI are discarded.
- Stateful packet inspectors can increase the performance of some networks because they allow fewer packets to pass.

**Iptables Basics** :iptables is made up of some basic structures known as tables, chains, and targets Let us look at each of these.

## Tables

Tables are a iptables construct that defines categories of functionality such as packet filtering or NAT.

There are four tables: FILTER, NAT, MANGLE and RAW. Filter is the default table if none other is specified. NAT is used to rewrite the source and/or destination of packets.

MANGLE is used for packet alteration such as modifying the TCP header. RAW is used for configuring exemptions from connection tracking.

## Chains

Each table has its own built-in chains and the user can define their own chains. Chains are lists of rules within a table. For our purposes here, the most important chains are

INPUT, OUTPUT and FORWARD.

### INPUT

This chain is for packets destined for the local system

### OUTPUT

This chain is for packets leaving the local system

### FORWARD

This chain is for packets being routed through the local system

### MATCH

A MATCH is where a packet meets the condition established by the rule. iptables then processes the packet according to the action in the rule.

### TARGETS

iptables supports a set of targets that trigger an action when the packet meets the condition of a rule. The most important of these are ACCEPT (allow the packet to pass), DROP (drop the packet), LOG, REJECT (drop the packet and send back an error) and RETURN.

## Steps to Configure

### Step #1: Installing Iptables

Iptables comes installed on nearly every Linux and \*nix system, but if for some reason your system doesn't have iptables tables installed, you can download it from the repository. **kali > sudo apt install iptables**

```
kali:kali:~$ sudo apt install iptables
[sudo] password for kali:
Reading package lists... Done
Building dependency tree...
Reading state information... Done
The following packages were automatically installed and are no longer required:
cabextract forensic-artifacts gnustep-base-common gnustep-common libarmadillo0 libbfio1 libcfitsio8 libdap25 libgdal27 libgnutls-dane0 libisl22
libmspack0 libpython3.8 libpython3.8-dev libunbound8 libvhdi1 libvmdk1 libxml-dom-perl libxml-perl libxml-regexp-perl libyara3 python3-acora
python3-arrow python3-artifacts python3-capstone python3-expiringdict python3-flask-restless python3-intervaltree python3-isodate python3-mimemparse
python3-mimerender python3-parsedatetime python3-psutil python3-pyaff4 python3-pyelftools python3-rdflib python3-sparqlwrapper python3-tsk
python3.8-dev
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
libip4tc2 libip6tc2 libxtables12
```

### Step #2: Configuring the Default Policy

Before we begin configuring our iptables, we must first decide what will be our default policy. In other words, what should the firewall do to packets that do not match any rule.

To see the default policy on your policy chains, simply enter; **kali**

> sudo iptables -L

```
kali:kali:~$ sudo iptables -L
[sudo] password for kali:
Chain INPUT (policy ACCEPT)
target     prot opt source          destination

Chain FORWARD (policy ACCEPT)
target     prot opt source          destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
```

### Step #3: Create Some rules

Next, let's create some rules. Let's assume that you want to block any packets coming from IP address 192.168.1.102. To create this rule, we simply do the following;

**-A** this appends this rule to the chain

**INPUT** looks to match on packets coming to the local system

**-s** sets the source address of the packets  
**-j** sets the target in this case **DROP**

- We can do the same for the entire sub-network by using CIDR notation or 192.168.1.0/24

```
kali㉿kali:~$ sudo iptables -A INPUT -s 192.168.1.0/24 -j DROP
```

- If we want to **DROP** packets destined for a particular port, we can use **-p** option followed by the protocol (tcp) and the **--dport** (destination port) followed by the port(ssh).

```
kali㉿kali:~$ sudo iptables -A INPUT -p tcp --dport ssh -j DROP
```

- If we wanted to accept connections to the website [www.amazon.com](http://www.amazon.com), we could build a rule that **ACCEPTs** outgoing connection (OUTPUT) over the TCP protocol (-p tcp) to [amazon.com](http://amazon.com) (-d amazon.com) **kali > sudo iptables -A OUTPUT -p tcp -d amazon.com -j ACCEPT**

```
kali㉿kali:~$ sudo iptables -A OUTPUT -p tcp -d amazon.com -j ACCEPT
```

- It's important to note that **iptables** will do a DNS lookup only at the time of the creation of the rule. If the IP address changes, the rule will become ineffective. For this reason, it is preferable to use the IP address of the domain.

If we wanted to block access to any other websites, we could create the following two rules;

```
kali > sudo iptables -A OUTPUT -p tcp --dport 80 -j DROP kali
```

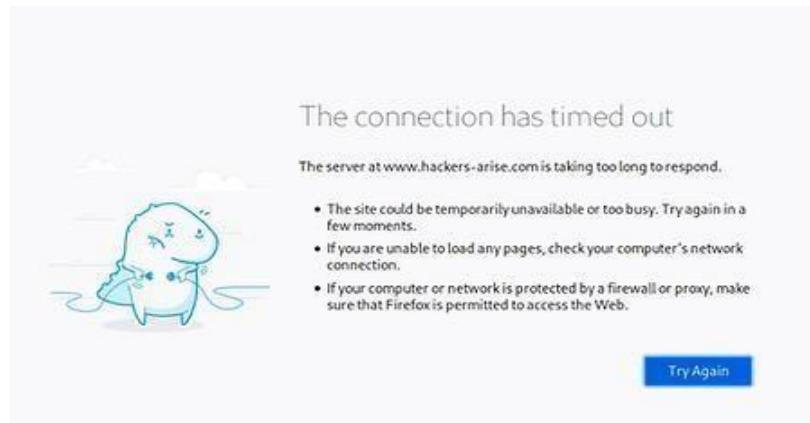
```
> sudo iptables -A OUTPUT -p tcp --dport 443 -j DROP
```

```
kali㉿kali:~$ sudo iptables -A OUTPUT -p tcp --dport 80 -j DROP
kali㉿kali:~$ sudo iptables -A OUTPUT -p tcp --dport 443 -j DROP
```

- The order of these rules is critical. **iptables** will search the rules until it finds a match. This means that if the last 2 rules, dropping port 80 and 443, were placed before the domain rule, the user would never be able to reach [amazon.com](http://amazon.com) as the drop rules

would match before reaching the domain rule.

- So when the local system attempts to connect to [amazon.com](http://amazon.com), they are blocked and the browser times out as seen below.



- Finally, we can view our table by using the -L or list option

```
kali㉿kali:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP      all  --  192.168.1.102          anywhere
DROP      all  --  192.168.1.0/24         anywhere
DROP      tcp  --  anywhere             anywhere           tcp dpt:ssh
ACCEPT    tcp  --  192.168.1.102          anywhere           tcp dpt:ssh state NEW,ESTABLISHED

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

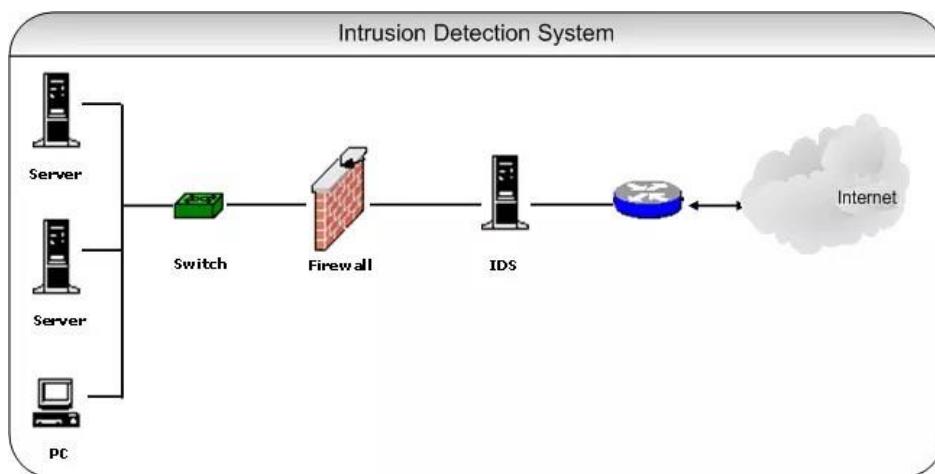
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT    tcp  --  anywhere            176.32.103.205
ACCEPT    tcp  --  anywhere            54.239.28.85
ACCEPT    tcp  --  anywhere            s3-console-us-standard.console.aws.amazon.com
DROP      tcp  --  anywhere            anywhere           tcp dpt:http
DROP      tcp  --  anywhere            anywhere           tcp dpt:https
kali㉿kali:~$
```

- To delete a table and start over, we can flush (-F) the table

## PRACTICAL -6

**Aim : BASIC configuration of Intrusion Detection System: Snort.**

### **Intrusion Detection System**



- A system called an intrusion detection system (IDS) observes network traffic for malicious transactions and sends immediate alerts when it is observed.
- It is software that checks a network or system for malicious activities or policy violations.
- Each illegal activity or violation is often recorded either centrally using a SIEM system or notified to an administration. IDS monitors a network or system for malicious activity and protects a computer network from unauthorized access from users, including perhaps insiders.
- The intrusion detector learning task is to build a predictive model (i.e., a classifier) capable of distinguishing between 'bad connections' (intrusion/attacks) and 'good (normal) connections.'

### **Working of IDS**

- An IDS (Intrusion Detection System) monitors the traffic on a computer network to detect any suspicious activity.
- It analyzes the data flowing through the network to look for patterns and signs of abnormal behavior.
- The IDS compares the network activity to a set of predefined rules and patterns to identify any activity that might indicate an attack or intrusion.
- If the IDS detects something that matches one of these rules or patterns, it sends an alert to the system administrator.

- The system administrator can then investigate the alert and take action to prevent any damage or further intrusion.

## Snort



- SNORT is a network-based intrusion detection system which is written in C programming language.
- It was developed in 1998 by Martin Roesch. Now it is developed by Cisco. It is free open-source software.
- It can also be used as a packet sniffer to monitor the system in real time.
- The network admin can use it to watch all the incoming packets and find the ones which are dangerous to the system.
- It is based on library packet capture tool.
- The rules are easy to create and implement and it can be deployed in any kind of operating system and any kind of network environment.
- The main reason of the popularity of this IDS over others is that it is a free-to-use software and open source because of which any user can be able to use it as the way he wants.

## Features of Snort

1. Real-time traffic monitor
2. Packet logging
3. Analysis of protocol
4. Content matching
5. OS fingerprinting
6. Can be installed in any network environment.

7. Creates logs
8. Open Source
9. Rules are easy to implement

## Different modes in SNORT

1. Sniffer Mode – To print TCP/IP header use command ./snort -v  
To print IP address along with header use command ./snort -vd
2. Packet Logging – To store packet in disk you need to give path where you want to store the logs.  
  
For this command is ./snort -dev -l ./SnortLogs.
3. Activate network intrusion detection mode – To start this mode use this command ./snort -dev -l ./SnortLogs -h 192.127.1.0/24 -c snort.conf

## Rules in SNORT

There are 3 types of rules in SNORT, those are

1. Alert Rules: This uses the alert technique to produce notifications.
2. Logging Rules: It logs each individual alert as soon as it is generated.
3. Pass Rules: If the packet is deemed malicious, it is ignored and dropped.

## Usage of SNORT

1. Packet Sniffing: The way traffic is being transmitted can be thoroughly examined by gathering the individual packets that travel to and from devices on the network.
2. Generates Alerts: It generates warnings based on the configuration file's rules when it discovers unusual or malicious activity, the possibility of a vulnerability being exploited, or a network threat that compromises the organization's security policy.
3. Debug Traffic: After the traffic has been logged, any malicious packets and configuration problems are checked.

## Configuration of IDS in SNORT

- Like all Linux commands and applications, Snort also has the help of the command line, which can be invoked by using the following code:

kali > sudo snort --help

- I have highlighted a few switches from the Help section of Snort:

-c gives us the location of the Snort rules and tells it to use its rules. They are the signature against which the new packets are verified.

-d tells Snort to show the application layer of data.

-e displays the Data Link layer of information, which contains the MAC address of the system.

-i allows the user to designate the interface we want to use. By default, Snort uses eth0.

-k allows the users to define how they want to store the details of the data capture performed by Snort.

-v is like in most of the programs — verbose, providing all the information.

```
GNU nano 5.4          /etc/snort/snort.conf *
# Setup the network addresses you are protecting
#
# Note to Debian users: this value is overriden when starting
# up the Snort daemon through the init.d script by the
# value of DEBIAN_SNORT_HOME_NET s defined in the
# /etc/snort/snort.debian.conf configuration file
#
ipvar HOME_NET 192.168.0.0/16

# Set up the external network addresses. Leave as "any" in most cases
ipvar EXTERNAL_NET any

^G Help      ^O Write Out ^W Where Is ^K Cut      ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste    ^J Justify
```

## Starting Snort

- Let us get to know some of the basic switches of Snort, by first running it. What makes Snort fabulous is that it can be run as a sniffer, packet logger, or even as a NIDS. In this article, we will look at Snort as a packet sniffer and NIDS.
- To run Snort in packet dump mode, use the following command:  
kali > sudo snort -vde
- The output we get is pretty self-explanatory (Figures 2). For using Snort as a NIDS, we need to instruct Snort to include the configuration file and rules.
- Generally, we can find the conf file at /etc/snort/snort.conf and that file will point to Snort rules. We need to give the -c switch and then the location.

kali > sudo snort -vde -c /etc/snort/snort.conf

We can also customise the rules to suit our needs.

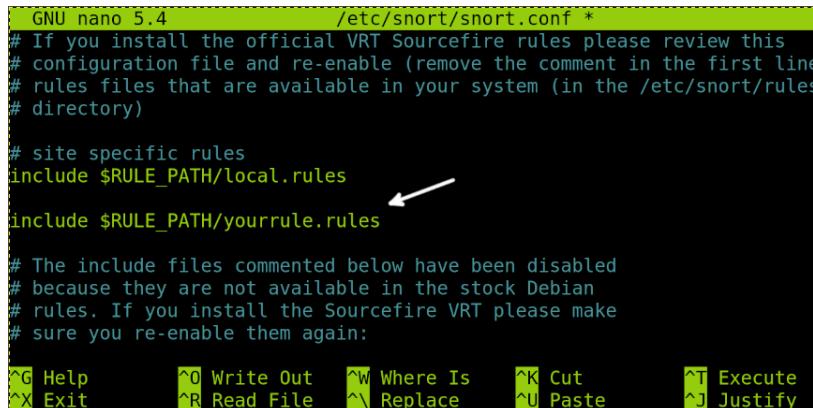
```
#  
# $Id: backdoor.rules,v 1.44.2.6.2.3 2005/05/31 17:13:02 mwatchinski Exp $  
#-----  
# BACKDOOR RULES  
#-----  
  
alert tcp $EXTERNAL_NET 27374 -> $HOME_NET any (msg:"BACKDOOR subseven 22"; flow:to_server,established; content:"|0D 0A|[RPL]002|0D 0A|"; reference:arachnids,485; reference:url,www.hackfix.org/subseven/; classtype:misc-activity; sid:103; rev:7;)  
alert tcp $HOME_NET 16959 -> $EXTERNAL_NET any (msg:"BACKDOOR subseven DEFCON8 2.1 access"; flow:from_server,established; content:"PWD"; classtype:trojan-activity; sid:107; rev:6;)  
  
alert tcp $HOME_NET 12345:12346 -> $EXTERNAL_NET any (msg:"BACKDOOR netbus active"; flow:from_server,established; content:"NetBus"; reference:arachnids,401; classtype:misc-activity; sid:109; rev:5;)  
alert tcp $EXTERNAL_NET any -> $HOME_NET 12345:12346 (msg:"BACKDOOR netbus getinfo"; flow:to_server,established; content:"GetInfo|0D|"; reference:arachnids,403; classtype:misc-activity; sid:110; rev:4;)  
:  
:
```

## Snort — rules and configuration

- Like all general Linux applications, Snort is configured via a conf file, which can be opened as a simple text file. Edit this text file, restart the application and we have a new working configuration.
- Before going any further, let's take a brief look into the syntax of Snort rules.
- Snort rules must be contained in a single line or we can use the multi-line character \. For example:

log tcp !x.x.x/xx OR log tcp !x.x.x/xx any -> xxx \ (msg: "some command")

- All rules should contain a rule header (which identifies the actions) and rule options (which identify the rule's alert messages).
- The rules must describe situations like a violation of the security policy of the company, or correctly detect the exploitable vulnerabilities.
- There are three kinds of rules in Snort:
- Alert rules: This generates alerts using the alert method.
- Log rules: Upon generation of any alert, it logs that specific alert.
- Pass rules: Ignores the packet if deemed malicious and drops it.
- Now we can move on to the configuration file, which can be opened using the following command:
- kali > mousepad /etc/snort/snort.conf
- In the Snort configuration file, we may see nine sections, as shown in Figure 4.
- For the most basic configuration, we will address only Sections 1, 6 and 7 as seen in Figure 5, 6, 7.



```
GNU nano 5.4          /etc/snort/snort.conf *
# If you install the official VRT Sourcefire rules please review this
# configuration file and re-enable (remove the comment in the first line)
# rules files that are available in your system (in the /etc/snort/rules
# directory)

# site specific rules
include $RULE_PATH/local.rules ←

include $RULE_PATH/yourrule.rules

# The include files commented below have been disabled
# because they are not available in the stock Debian
# rules. If you install the Sourcefire VRT please make
# sure you re-enable them again:

^G Help      ^O Write Out   ^W Where Is   ^K Cut       ^T Execute
^X Exit      ^R Read File   ^\ Replace    ^U Paste    ^J Justify
```

### Setting variables

- In the screenshot in Figure 5, we can see the highlighted line ‘ipvar HOME\_NET’. This variable denotes the network is protected. ‘HOME\_NET’ is the variable name to which the IP address is assigned. This can be a single IP address, a list of IP addresses, or a subnet in CIDR notation, or even can be left as any.

### Checking the output

- Using Ctrl+F, we then move on to output plugins (Figure 6). By default, Snort sends the output in log format. But if we want, we can comment out that line (unified2) and uncomment the bottom line (log\_tcpdump), enabling the output in tcpdump format, which is saved in the /var/log/snort directory.

### Disable rules

- Depending on your enterprise, we may need to change the rules that Snort relies upon, and customise them in Section 7, as shown in Figure 7.
- To not let Snort use a given set, simply comment out the include part.
- After making any change, simply save the file and test the configuration using the -T switch.

kali > sudo snort -T -c /etc/snort/snort.conf

```
linuxhint@LinuxHint:~$ sudo snort -d -l /var/log/snort/ -h 192.168.0.1/16 -A console -c /etc/snort/snort.conf
Running in IDS mode

      --- Initializing Snort ---
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
PortVar 'HTTP_PORTS' defined : [ 80:81 311 383 591 593 901 1220 1414 1741
1830 2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145
7510 7777 7779 8000 8008 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181
8243 8280 8300 8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 3444
3:34444 41080 50002 55555 1
-----  
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: appid Version 1.1 <Build 5>
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Commencing packet processing (pid=55415)
05/23-21:44:01.544344  [**] [1:100006927:1] SSH incoming [**] [Priority: 0]
{TCP} 192.168.0.101:48928 -> 192.168.0.103:22
```

## PRACTICAL -7

**Aim : Network vulnerability assessment using OpenVAS/Necuss Framework.**

### **Network Vulnerability Assessment**

- A vulnerability assessment is a process that helps review and analyze endpoint and device networks for security issues.
- The assessment may detect network flaws and holes in the network that could leave an opportunity for hackers to exploit.
- A network vulnerability should also be performed on an ongoing basis as new threats arise and hackers find additional ways to break into systems.



## STEPS TO CONFIGURE AND RUN OPEN VAS IN KALI LINUX

### 1) Installing Openvas on Kali Linux

- To install Openvas and its dependencies on our Kali Linux system run the following command:

```
sudo apt update
```

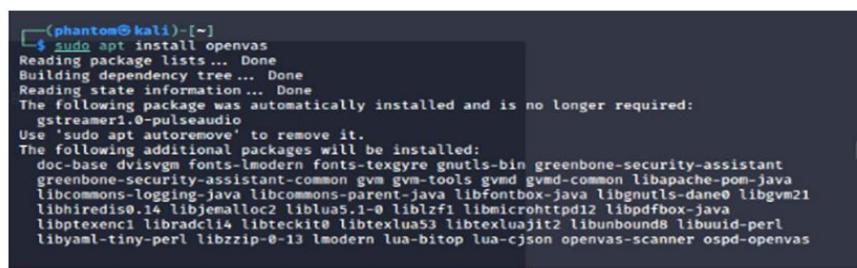
```
sudo apt upgrade -y
```

```
sudo apt dist-upgrade -y
```

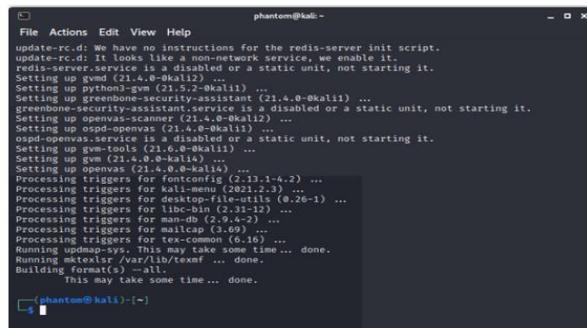


```
phantom@kali:~  
File Actions Edit View Help  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
└─(phantom㉿kali)-[~]  
$ sudo apt dist-upgrade  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done  
The following package was automatically installed and is no longer required:  
  gstreamer1.0-pulseaudio  
Use 'sudo apt autoremove' to remove it.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

```
sudo apt install openvas
```



```
└─(phantom㉿kali)-[~]  
$ sudo apt install openvas  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following package was automatically installed and is no longer required:  
  gstreamer1.0-pulseaudio  
Use 'sudo apt autoremove' to remove it.  
The following additional packages will be installed:  
  doc-base dvisvgm fonts-lmodern fonts-texgyre gnults-bin greenbone-security-assistant  
  greenbone-security-assistant-common gvm gvm-tools gvmd gvmd-common libapache-pom-java  
  libcommons-logging-java libcommons-parent-java libfontbox-java libgnutls-dane0 libgvm21  
  libhiredis0.14 libjemalloc2 liblbus5.1-0 liblzfl libmicrohttpd12 libpdfbox-java  
  libptexenc1 libradcli4 libteckit libtexlua53 libtexluajitz libunbound2 libuuid-perl  
  libyaml-tiny-perl libzzip0-13 lmodern lua-bitop lua-cjson openvas-scanner ospd-openvas
```



```
phantom@kali:~  
File Actions Edit View Help  
update-rc.d: We have no instructions for the redis-server init script.  
update-rc.d: It looks like a non-network service, we enable it.  
redis-server is not currently installed or a static unit, not starting it.  
Setting up gwm (21.4.0-0kali12) ...  
Setting up python3-gvm (21.5.2-0kali1) ...  
Setting up greenbone-security-assistant (21.4.0-0kali1) ...  
greenbone-security-assistant is not currently installed or a static unit, not starting it.  
Setting up openvas-scanner (21.4.0-0kali12) ...  
Setting up ospd-openvas (21.4.0-0kali1) ...  
ospd-openvas.service is not currently installed or a static unit, not starting it.  
Setting up gdm (21.0.0-0kali11) ...  
Setting up gdm (21.4.0-0kali16) ...  
Setting up openvas (21.4.0-0kali14) ...  
Processing triggers for kali-menu (2021.2.1-2) ...  
Processing triggers for desktop-file-utils (0.26-1) ...  
Processing triggers for libc-bin (2.31-12) ...  
Processing triggers for libfontbox1 (2.9.4-2) ...  
Processing triggers for mailcap (3.69) ...  
Processing triggers for tex-common (6.16) ...  
Running update-sysv ... may take some time ... done.  
Running update-etc ... done.  
Running update-etc /var/lib/cxmf ... done.  
Building format(s) --all.  
This may take some time ... done.  
└─(phantom㉿kali)-[~]
```

- The next step is to run the installer, which will configure OpenVAS and download various network vulnerability tests (NVT) or signatures. Due to many NVTs (50.000+), the setting process may take some time and consume a lot of data.
- In the test setup we used for this tutorial, the complete setup process took 10 minutes, which is not bad.
- Run the following command to start the setup process:  
gvm-setup

```
[>] Creating database
CREATE ROLE
GRANT ROLE
CREATE EXTENSION
CREATE EXTENSION
[>] Migrating database
[>] Checking for admin user
[*] Creating user admin for gvm
[*] Please note the generated admin password
[*] User created with password 'c273c26d-28d3-485b-9865-5c96e30acf6d'.
[*] Define Feed Import Owner
[>] Updating OpenVAS feeds
[*] Updating: NVT
Greenbone community feed server - http://feed.community.greenbone.net/
This service is hosted by Greenbone Networks - http://www.greenbone.net/
All transactions are logged.

If you have any questions, please use the Greenbone community portal.
See https://community.greenbone.net for details.

By using this service you agree to our terms and conditions.

Only one sync per time, otherwise the source ip will be temporarily blocked.
```

```
phantom@kali: ~
File Actions Edit View Help
dfn-cert-2019.xml
 3,549,005 100% 367.22kB/s  0:00:09 (xfr#22, to-chk=6/29)
dfn-cert-2020.xml
 3,659,131 100% 363.89kB/s  0:00:09 (xfr#23, to-chk=5/29)
dfn-cert-2021.xml
 1,749,636 100% 374.37kB/s  0:00:04 (xfr#24, to-chk=4/29)
shasums
 1,419 100%  3.99kB/s  0:00:00 (xfr#25, to-chk=3/29)
sha256sums
 2,019 100%  5.68kB/s  0:00:00 (xfr#26, to-chk=2/29)
sha256sums.asc
 819 100%  1.78kB/s  0:00:00 (xfr#27, to-chk=1/29)
timestamp
 13 100%  0.03kB/s  0:00:00 (xfr#28, to-chk=0/29)

sent 711 bytes received 76,459,880 bytes 403,485.97 bytes/sec
total size is 76,439,315 speedup is 1.00
[*] Checking Default scanner
06b69003-5fc2-4037-a479-93b440211c73  OpenVAS /var/run/ospd/ospd.sock  0  OpenVAS Default

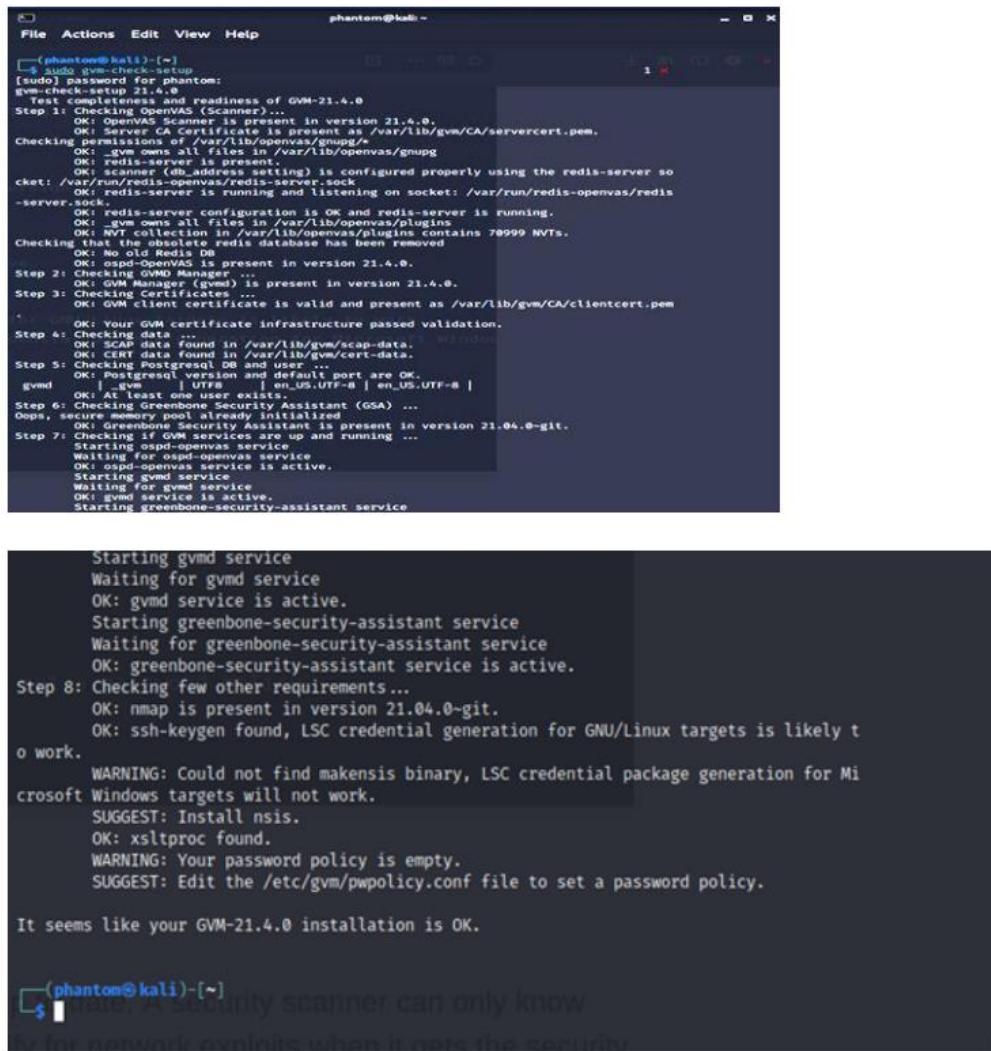
[+] Done
[*] Please note the password for the admin user
[*] User created with password 'c273c26d-28d3-485b-9865-5c96e30acf6d'.
```

- After the configuration process is complete, all the necessary OpenVAS processes will start and the web interface will open automatically.
- The web interface is running locally on port 9392 and can be accessed through <https://localhost:9392>.
- OpenVAS will also set up an admin account and automatically generate a password for this account which is displayed in the last section of the setup output.

## 2) Verify the Installation

- You can verify your installation with.

gvm-check-setup



```
phantom@kali:~$ sudo -u gvm gvm-check-setup 21.4.0
[sudo] password for phantom:
gvm-check-setup 21.4.0
  Test completeness and readiness of GVM-21.4.0
Step 1: Checking OpenVAS ...
  OK: OpenVAS Scanner is present in version 21.4.0.
  OK: Server CA Certificate is present at /var/lib/gvm/CA/servercert.pem.
Checking permissions on /var/lib/openvas/gnupg/
  OK: All files in /var/lib/openvas/gnupg
  OK: redis-server is present.
  OK: scanner (db_address setting) is configured properly using the redis-server socket: /var/run/redis-openvas/redis-server.sock.
  OK: redis-server is running and listening on socket: /var/run/redis-openvas/redis-server.sock.
  OK: redis-server configuration is ok and redis-server is running.
  OK: redis client library /var/lib/openvas/plugins
  OK: NVT collection in /var/lib/openvas/plugins contains 70999 NVTs.
Checking that the obsolete redis database has been removed
  OK: No old Redis database.
  OK: ospd-OpenVAS is present in version 21.4.0.
Step 2: Checking GVM Manager ...
  OK: GVM Manager (gvmd) is present in version 21.4.0.
Step 3: Checking certificates...
  OK: GVM client certificate is valid and present as /var/lib/gvm/CA/clientcert.pem
.
  OK: Your GVM certificate infrastructure passed validation.
Step 4: Checking data ...
  OK: SCAP data found in /var/lib/gvm/scap-data.
  OK: CERT data found in /var/lib/gvm/cert-data.
Step 5: Checking PostgreSQL DB and ports ...
  OK: PostgreSQL version and default port are OK.
  gvard | _gvm | UTF8 | en_US.UTF-8 | en_US.UTF-8 |
  OK: At least one user exists.
Step 6: Checking Greenbone Security Assistant (GSA) ...
  Ooops, secure memory pool already initialized
  OK: Greenbone Security Assistant is present in version 21.04.0-git.
Step 7: Checking installed services are up and running ...
  Starting ospd-openvas service
  Waiting for ospd-openvas service
  OK: ospd-openvas service is active.
  Starting gvard service
  Waiting for gvard service
  OK: gvard service is active.
  Starting gvmdb service
  Starting greenbone-security-assistant service
.
Starting gvmdb service
Waiting for gvmdb service
OK: gvmdb service is active.
Starting greenbone-security-assistant service
Waiting for greenbone-security-assistant service
OK: greenbone-security-assistant service is active.
Step 8: Checking few other requirements ...
  OK: nmap is present in version 21.04.0-git.
  OK: ssh-keygen found, LSC credential generation for GNU/Linux targets is likely to work.
  WARNING: Could not find makensis binary, LSC credential package generation for Microsoft Windows targets will not work.
  SUGGEST: Install nsis.
  OK: xsltproc found.
  WARNING: Your password policy is empty.
  SUGGEST: Edit the /etc/gvm/pwpolicy.conf file to set a password policy.

It seems like your GVM-21.4.0 installation is OK.

phantom@kali:~$
```

## 3) Password Resetting

- You can change the admin password sing the following commands:  
`gvmd --user=admin --new-password=passwd;`
- The next step is to accept the self-signed certificate warning and use the automatically generated admin credentials to login on to the web interface:

The image shows two screenshots of the Greenbone Security Assistant web interface. The top screenshot is a login page with a green header 'Greenbone Security Assistant' and a sub-header 'Version 7.0.2'. It features a cartoon green dragon logo on the right and a login form with fields for 'Username' (admin) and 'Password' (represented by a series of blue dots). The bottom screenshot shows a dashboard with a green header 'Greenbone Security Assistant' and a sub-header 'Dashboard'. The dashboard includes several widgets: 'Tasks by Severity Class (Total: 0)', 'Tasks by status (Total: 0)', 'CVTs by creation time (Total: 104997)', 'Hosts topology' (with a note 'No hosts with topology selected'), and 'NVTs by Severity Class (Total: 51123)' with a pie chart showing distribution across High, Medium, Low, and Log categories.

`sudo gvm-start`

A terminal session on a Kali Linux system (phantom㉿kali) executing the command `sudo gvm-start`. The terminal prompts for a password for the user phantom. It then displays a warning message about port 9392 being used. The output shows the process ID (861) and name (gvm) for the gsad daemon, which is listening on port 9392. Finally, the command `ps aux | grep gvm` is run to verify the process is running.

```
(phantom㉿kali)-[~] $ sudo gvm-start
[sudo] password for phantom:
[-] Something is already using port: 9392/tcp
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
gsad 861 _gvm 10u IPv4 17527      0t0  TCP localhost:9392 (LISTEN)

UID        PID  PPID  C STIME TTY      STAT   TIME CMD
_gvm     861      1  0 14:02 ?
          861      1  0 14:02 ?      Sl    0:00 /usr/sbin/gsad --listen=127.0.0.1 --port=9392

(phantom㉿kali)-[~]
```

```
phantom@kali: ~
File Actions Edit View Help
(phantom㉿kali)-[~]
$ sudo gvm-stop
[sudo] password for phantom:
[>] Stopping OpenVAS services
● greenbone-security-assistant.service - Greenbone Security Assistant (gsad)
    Loaded: loaded (/lib/systemd/system/greenbone-security-assistant.service; enabled; vendor preset
: disabled)
    Active: inactive (dead) since Sat 2021-06-26 14:23:35 EDT; 505ms ago
      Docs: man:gsad(8)
         https://www.greenbone.net
     Process: 834 ExecStart=/usr/sbin/gsad --listen=127.0.0.1 --port=9392 (code=exited, status=0/SUCCE
55)
      Main PID: 836 (code=killed, signal=TERM)
        CPU: 17ms

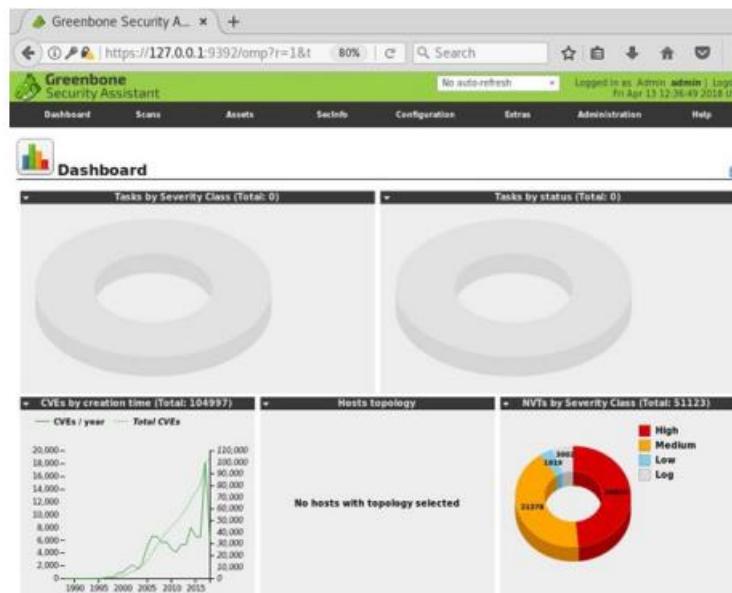
Jun 26 14:22:22 kali systemd[1]: Starting Greenbone Security Assistant (gsad)...
Jun 26 14:22:22 kali gsad[834]: Oops, secure memory pool already initialized
Jun 26 14:22:22 kali systemd[1]: Started Greenbone Security Assistant (gsad).
Jun 26 14:23:35 kali systemd[1]: Stopping Greenbone Security Assistant (gsad)...
Jun 26 14:23:35 kali systemd[1]: greenbone-security-assistant.service: Succeeded.
Jun 26 14:23:35 kali systemd[1]: Stopped Greenbone Security Assistant (gsad).

● gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)
    Loaded: loaded (/lib/systemd/system/gvmd.service; enabled; vendor preset: disabled)
    Active: inactive (dead) since Sat 2021-06-26 14:23:35 EDT; 547ms ago
      Docs: man:gvmd(8)
     Process: 812 ExecStart=/usr/sbin/gvmd --osp-vt-update=/run/ospd/ospd.sock (code=exited, status=0/
SUCCESS)
      Main PID: 813 (code=killed, signal=TERM)
        CPU: 600ms

Jun 26 14:22:21 kali systemd[1]: Starting Greenbone Vulnerability Manager daemon (gvmd)...
```

- \*Note: To create a new user :  
sudo runuser -u \_gvm -- gvmd --create-user=admin2 --new-password=12345
- To change the password of the existing user:  
sudo runuser -u \_gvm -- gvmd --user=admin --new-password=new\_password

#### 4) Configuration for New Target

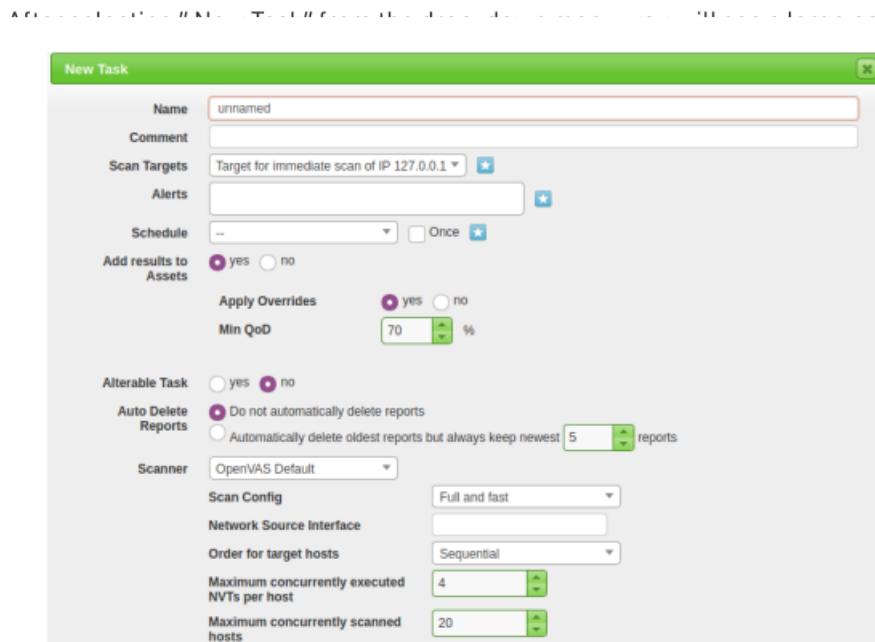
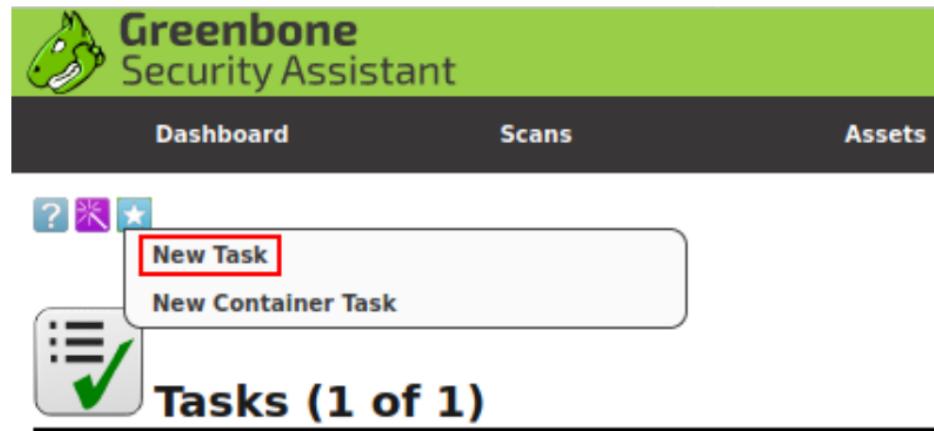


#### 5) Scheduling the scan Process

- Expand the car to scan and> start the task of creating a scan task for the managed computer.

- Creating a Task

To create a custom task, navigate to the star icon in the upper right corner of the taskbar and select New task.



- For this task, we'll be specializing only in the Name, Scan Targets, and Scanner Type, and Scan Config. In later tasks, we will be focusing on the opposite choices for additional advanced configuration and implementation/automation.

1. Name: permits North American country to line the name the scan are going to be referred to as inside OpenVAS
2. Scan Targets: The targets to scan, can embrace Hosts, Ports, and Credentials. To make a brand new target you may follow another pop-up, this can be lined later during this task.

3. Scanner: The scanner to use by default will use the OpenVAS design but you'll be able to set this to any scanner of your selecting within the settings menu.
4. Scan Config: OpenVAS has seven totally different scan sorts you can choose from and can be used supported however you're aggressive or what info you wish to gather from your scan.

## 6) Scoping a new target

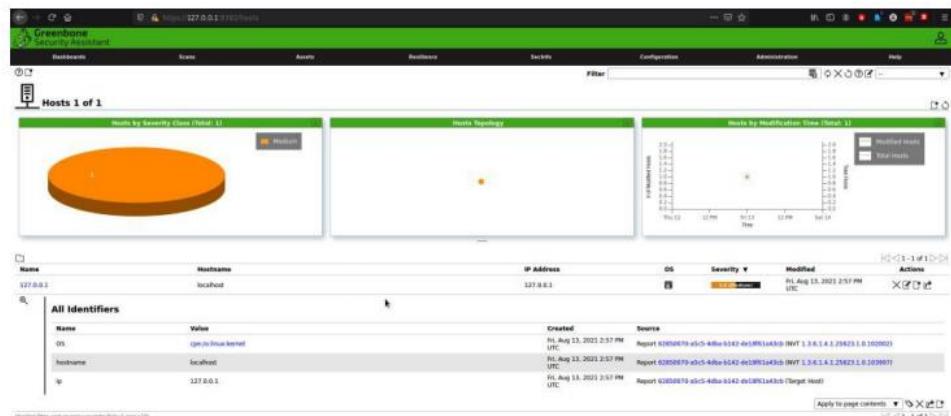
- To scope a new target, navigate to the star icon next to Scan Targets.

The screenshot shows the 'New Target' dialog box. It includes fields for Name (set to 'unnamed'), Comment, Hosts (set to '172.17.0.1'), Exclude Hosts, Reverse Lookup Only, Reverse Lookup Unify, Port List (set to 'All IANA assigned TCP 20...'), Alive Test (set to 'Scan Config Default'), and Credentials for authenticated checks (SSH on port 22). A 'Create' button is at the bottom right.

The screenshot shows the 'New Target' dialog box again, but with different values. The Name field is set to 'DVWA'. The Hosts section has 'Manual' selected and '10.10.147.246' entered. The 'From file' option is also present. A 'Create' button is at the bottom right.

- To run the task, navigate to the run icon within the operation.



## 7) Additional Features

Allow adding common parameters to OpenVAS:

The screenshot shows the Greenbone Security Assistant web interface at https://127.0.0.1:9392/trashcan. The top navigation bar includes links for Dashboards, Scans, Assets, Findings, Scripts, Configuration, Administration, and Help. A trash can icon is visible in the top left. The main content area is titled 'Trashcan' and contains a table with two columns: 'Type' and 'Items'. The types listed are Alerts, Audits, Credentials, Events, Groups, Notes, Overrides, Permissions, Policies, Port Lists, Report Formats, Roles, Scan Configs, Scenarios, Schedules, Tags, Targets, Tasks, and Tickets. Each type has a count of 0 items.

## Administration

As the name suggests, you can manage passwords, users, etc.:

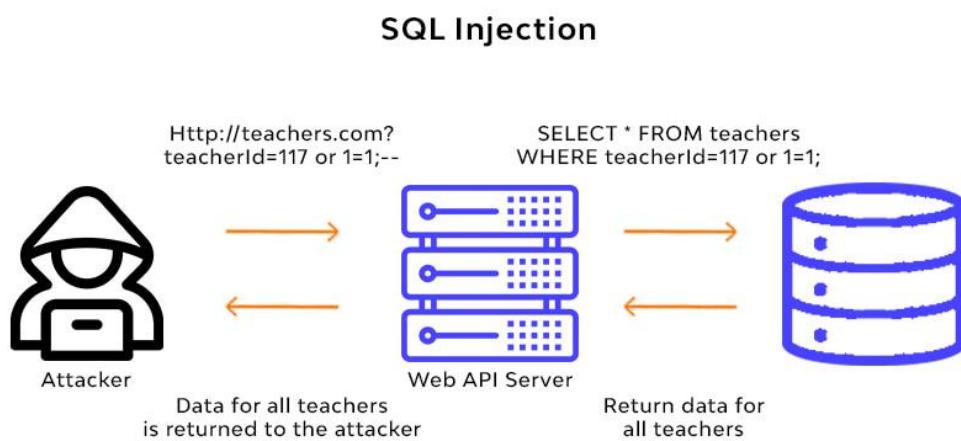
The screenshot shows the Greenbone Security Assistant web interface at https://127.0.0.1:9392/users. The top navigation bar includes links for Dashboards, Scans, Assets, Findings, Scripts, Configuration, Administration, and Help. A user icon is visible in the top left. The main content area is titled 'Users 1 of 1' and displays a table with columns: Name, Roles, Groups, Host Access, Authentication Type, and Actions. There is one user entry: 'admin2' with 'Admin' role, 'Allow all' host access, 'Local' authentication type, and standard edit and delete icons. The bottom right shows a page footer with '1 - 1 of 1' and a 'Apply to page contents' button.

## PRACTICAL -8

**Aim : Demonstrate automated SQL injection with SqLMap Or OWASP Juice Shop.**

### **SQL Injection**

- SQL injection is a technique used to extract user data by injecting web page inputs as statements through SQL commands. Basically, malicious users can use these instructions to manipulate the application's web server.
1. SQL injection is a code injection technique that can compromise your database.
  2. SQL injection is one of the most common web hacking techniques.
  3. SQL injection is the injection of malicious code into SQL statements via web page input.



### **Types of SQL injection**

SQL injections typically fall under three categories: In-band SQLi (Classic), Inferential SQLi (Blind) and Out-of-band SQLi. You can classify SQL injections types based on the methods they use to access backend data and their damage potential.

## 1) In-band SQLi

The attacker uses the same channel of communication to launch their attacks and to gather their results. In-band SQLi's simplicity and efficiency make it one of the most common types of SQLi attack. There are two sub-variations of this method:

- **Error-based SQLi**—the attacker performs actions that cause the database to produce error messages. The attacker can potentially use the data provided by these error messages to gather information about the structure of the database.
- **Union-based SQLi**—this technique takes advantage of the UNIONSQL operator, which fuses multiple select statements generated by the database to get a single HTTP response. This response may contain data that can be leveraged by the attacker.

## 2) Inferential (Blind) SQLi

The attacker sends data payloads to the server and observes the response and behavior of the server to learn more about its structure. This method is called blind SQLi because the data is not transferred from the website database to the attacker, thus the attacker cannot see information about the attack in-band.

Blind SQL injections rely on the response and behavioral patterns of the server so they are typically slower to execute but may be just as harmful. Blind SQL injections can be classified as follows:

- **Boolean**—the attacker sends a SQL query to the database prompting the application to return a result. The result will vary depending on whether the query is true or false. Based on the result, the information within the HTTP response will modify or stay unchanged. The attacker can then work out if the message generated a true or false result.
- **Time-based**—attacker sends a SQL query to the database, which makes the database wait (for a period in seconds) before it can react. The attacker can see from the time the database takes to respond, whether a query is true or false. Based on the result, an HTTP response will be generated instantly or after a waiting period. The attacker can thus work out if the message they used returned true or false, without

relying on data from the database.

### 3) Out-of-band SQLi

The attacker can only carry out this form of attack when certain features are enabled on the database server used by the web application. This form of attack is primarily used as an alternative to the in-band and inferential SQLite techniques.

Out-of-band SQLi is performed when the attacker can't use the same channel to launch the attack and gather information, or when a server is too slow or unstable for these actions to be performed. These techniques count on the capacity of the server to create DNS or HTTP requests to transfer data to an attacker.

### Demonstration using OWASP Juice Shop.

- 1) Open the official OWASP juice shop website with URL: <https://owasp.org/www-project-juice-shop/>

The screenshot shows the OWASP Juice Shop homepage. At the top, there's a navigation bar with links to 'PROJECTS', 'CHAPTERS', 'EVENTS', and 'ABOUT'. Below the navigation is a search bar, a 'Store' button, and a 'Donate' button. The main content area features a logo of a yellow juice carton with 'JS' on it. Below the logo is a navigation menu with tabs for 'Main', 'Overview', 'News', 'Challenges', 'Learning', 'CTF', 'Ecosystem', and 'Supporters'. A GitHub badge indicates the project is a 'flagship project' with a release of v14.5.1, 7.9k stars, and a 'Follow' button. Another GitHub badge shows it's an 'openSSF best practices gold' project. A 'Contributor Covenant v2.0 adopted' badge is also present. The main text on the page describes the Juice Shop as a modern and sophisticated insecure web application used for security trainings, awareness demos, CTFs, and as a guinea pig for security tools. It mentions vulnerabilities from the OWASP Top Ten and real-world applications. To the right, there's a sidebar with sections for 'Project Information' (Flagship Project, Classification, Tool), 'Sources' (GitHub, CTF Extension, Crowdin I18N), 'Documentation' (Online Demo, Introduction Slides, Companion Guide), and 'Community' (Chat, Subreddit).

- 2) Find the link of “Online Demo” in the Documentation section:

#### Description

Juice Shop is written in Node.js, Express and Angular. It was the first application written entirely in JavaScript listed in the [OWASP VWA Directory](#).

The application contains a vast number of hacking challenges of varying difficulty where the user is supposed to exploit the underlying vulnerabilities. The hacking progress is tracked on a score board. Finding this score board is actually one of the (easy) challenges!

Apart from the hacker and awareness training use case, pentesting proxies or security scanners can use Juice Shop as a “guinea pig”-application to check how well their tools cope with JavaScript-heavy application frontends and REST APIs.

*Translating “dump” or “useless outfit” into German yields “Saftladen” which can be reverse-translated word by word into “juice shop”. Hence the project name. That the initials “JS” match with those of “JavaScript” was purely coincidental!*

#### Docker Image

#### Sources

[GitHub](#)  
[CTF Extension \(GitHub\)](#)  
[Crowdin I18N](#)

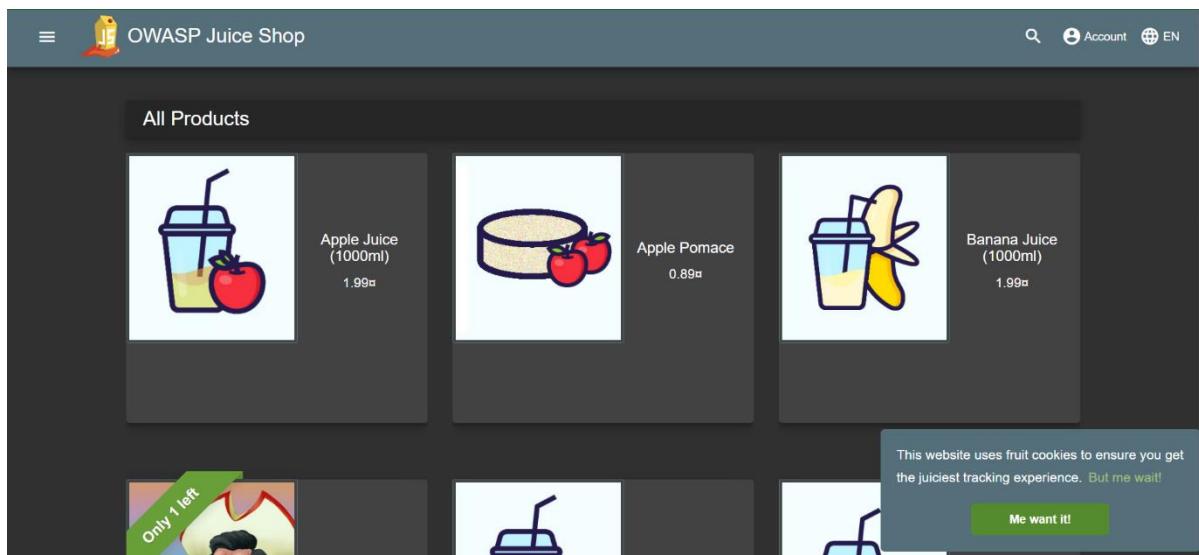
#### Documentation

[Online Demo](#)  
[Introduction Slides](#)  
[Companion Guide \(LeanPub/Online\)](#)

#### Community

[Chat \(Gitter/Matrix\)](#)  
[Subreddit](#)

3) The following window will be displayed:

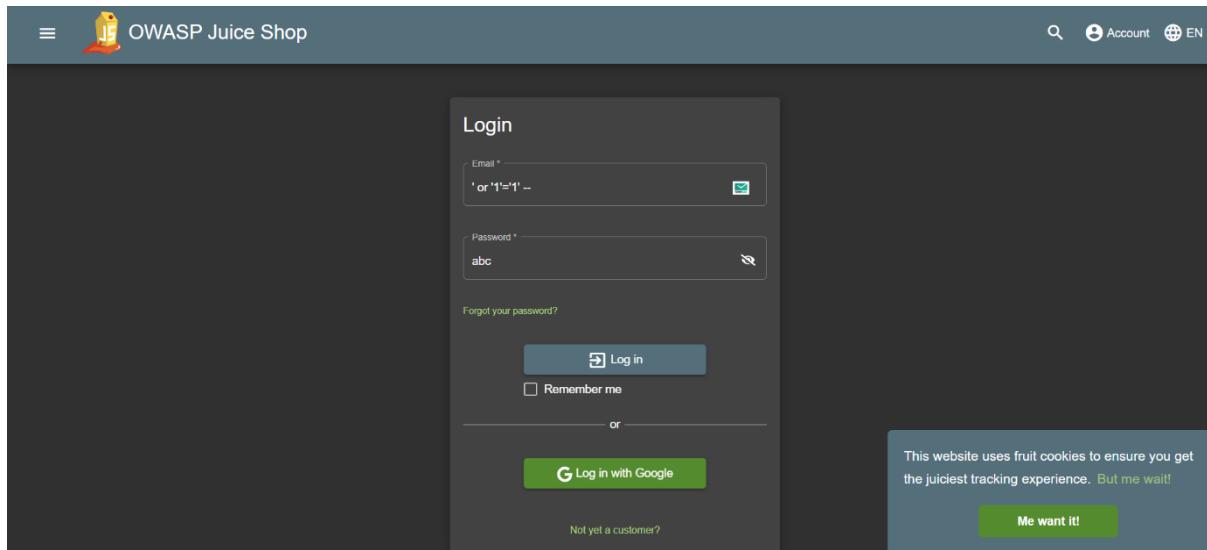


4) Click on “Account” and click on Login:

In the “Email” section, write the following SQL Query from inside the brackets:

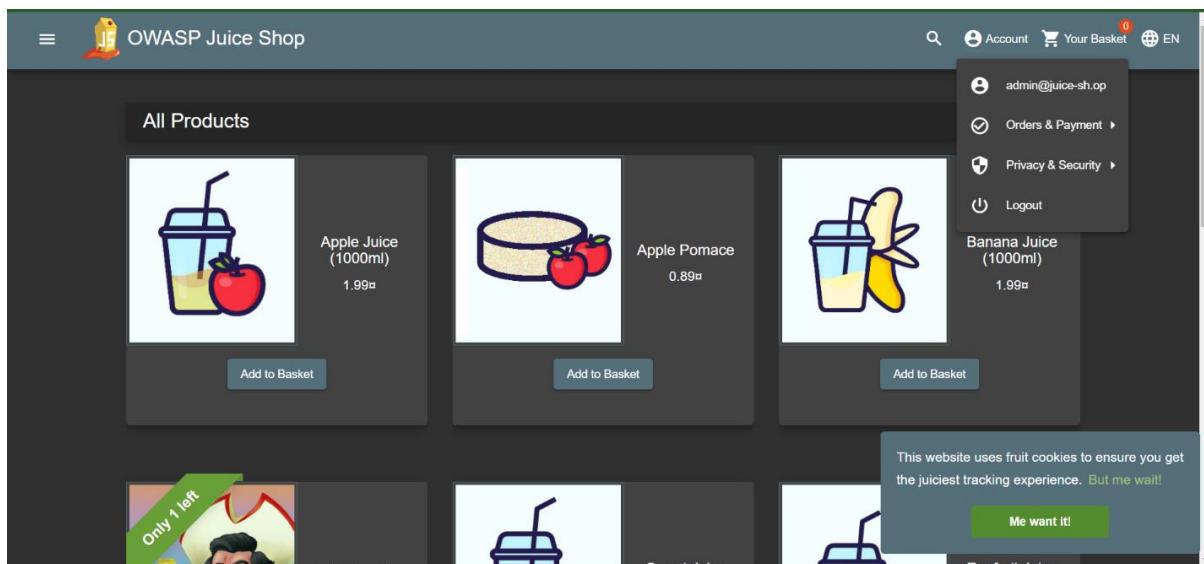
(‘ or ‘1’=’1’ -- )

We can write anything in the password section.



And click Log in

- 5) Now, you should be logged in to the web application without knowing the username or the password, that too with ADMIN privileges.



Now, after being logged in, we can perform any task as we are having administrative rights to the web application.

## **PRACTICAL -9**

**Aim : Demonstrate Application Injection using Zed Attack Proxy.**

### **Zed Attack Proxy**

- Zed Attack Proxy is an open-source security software written in Java programming language and released in 2010. It is used to scan web applications and find vulnerabilities in it.
- It was started as a small project by the Open Web Application Security Project (OWASP) and now it is the most active project maintained by thousands of individuals around the globe.
- It is available for Linux, Windows, and mac in 29 languages. It can also be used as a proxy server like a burp suite to manipulate the request including the HTTPS request.
- Daemon mode is also present in it which can later be controlled by REST API.

### **Features of Zed Attack Proxy -OWASP**

- Passive Scanner
- Automated Scanner
- Proxy Server
- Port Identification
- Directory Searching
- Brute Force Attack
- Web Crawler
- Fuzzer

### **Use of Zed Attack Proxy**

-Zed Attack Proxy is used to detect vulnerabilities present on any web server and try to remove them. Here is some big vulnerability that could be present in the web server:

- SQL injection
- Cross-site scripting (XSS)
- Broken access control
- Security miss-configuration
- Broken authentication

- Sensitive data exposure
- Cross-site request forgery (CSRF)
- Using components with known vulnerabilities.

## Demonstration/Practical

**Step1:** Go to the Foxy Proxy

website: <https://addons.mozilla.org/enUS/firefox/addon/foxyproxy-standard/>.

**Step 2:** Click “Continue to download.”

**Step 3:** Click “add to Firefox.”

**Step 4:** Firefox should prompt you to install Foxy Proxy. Firefox will need to be restarted at this point.

The steps for Automatic SQL Injection in Web Applications using ZED Attack Proxy are as follow:

1. Launch Jx Browser by clicking on the highlighted icon.



Jx Browser's GUI is as given:

The screenshot shows the ZAP Browser interface. At the top, it says "Welcome to the ZAP Browser (based on JxBrowser)". Below that, there is a message about proxy configuration. The main window has tabs for "Sites" and "Analyses". Under "Analyses", there is a tree view showing a context named "Http://testfire.net" with sub-items "POST doLogin.jsp?submit=login&uid". Underneath this, there are "images", "GETlogin.jsp", and "GETstyle.css". To the right of the tree view, there is a "Header Tab" showing a POST request to "http://testfire.net/doLogin HTTP/1.1" with various headers like "Proxy-Connection: keep-alive", "Content-Length: 35", and "Cache-Control: max-age=0". Below the header tab is a "Body Tab" containing the URL "uid=Test1passwTest1&submit=Login". At the bottom of the interface, there is a table titled "History" showing network traffic details:

ID	Req. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Body	Highest Alert	Note	Tags
1	5/11/19 3:31:15 PM	GET	http://testfire.net/	200	OK	972 ms	9,369 bytes	Medium		Form, SetCookie, Com.
3	5/11/19 3:31:16 PM	GET	http://testfire.net/style.css	200	OK	660 ms	1,251 bytes	Low		
12	5/11/19 3:31:21 PM	GET	http://testfire.net/login.jsp	200	OK	680 ms	8,519 bytes	Medium		Form, Password, Slop..
13	5/11/19 3:31:34 PM	POST	http://testfire.net/doLogin	302	Found	682 ms	0 bytes	Medium		
14	5/11/19 3:31:35 PM	GET	http://testfire.net/login.jsp	200	OK	338 ms	8,622 bytes	Medium		Form, Password, Slop..

3. Observe network traffic, which, accessed via Jx Browser, captured in the ZAP tool.

The screenshot shows the OWASP ZAP interface. The 'Request' tab is selected. In the 'Header' section, there is a POST request to `http://testfire.net/dologin` with the following headers:

```

POST /http://testfire.net/dologin HTTP/1.1
Proxy-Connection: keep-alive
Content-Length: 35
Cache-Control: max-age=0
Origin: http://testfire.net
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: http://testfire.net/login.jsp
Accept-Language: en-gb
Cookie: JSESSIONID=04D439EBC12512311027C482CE1871B
Host: testfire.net

```

The 'Body' section contains the payload: `uid=Test1;password=Test1&Submit=Login`. Below the request, the 'History' tab shows the following log entries:

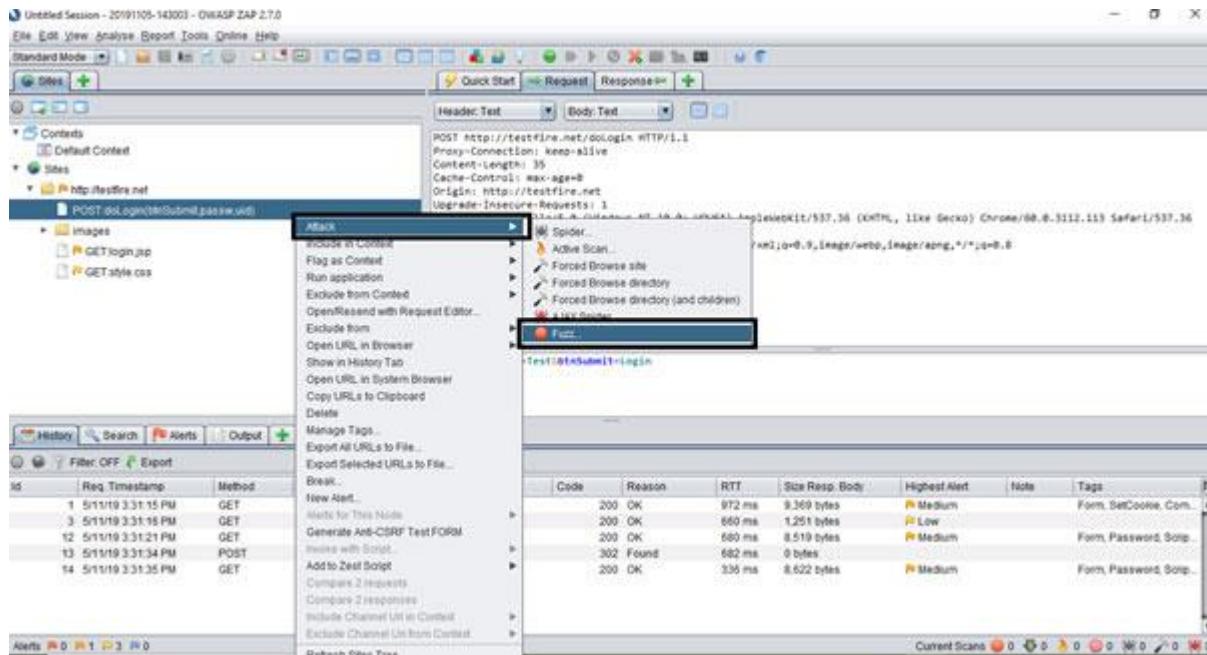
ID	Req. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Body	Highest Alert	Note	Tags
1	5/11/19 3:31:15 PM	GET	<code>http://testfire.net/</code>	200	OK	972 ms	9,369 bytes	Medium		Form, SetCookie, Com...
3	5/11/19 3:31:16 PM	GET	<code>http://testfire.net/style.css</code>	200	OK	660 ms	1,251 bytes	Low		
12	5/11/19 3:31:21 PM	GET	<code>http://testfire.net/login.jsp</code>	200	OK	680 ms	8,519 bytes	Medium		Form, Password, Scrip...
13	5/11/19 3:31:34 PM	POST	<code>http://testfire.net/dologin</code>	302	Found	682 ms	0 bytes	Medium		
14	5/11/19 3:31:35 PM	GET	<code>http://testfire.net/login.jsp</code>	200	OK	336 ms	8,622 bytes	Medium		Form, Password, Scrip...

#### 4. Now find out the Post method inside the login API.

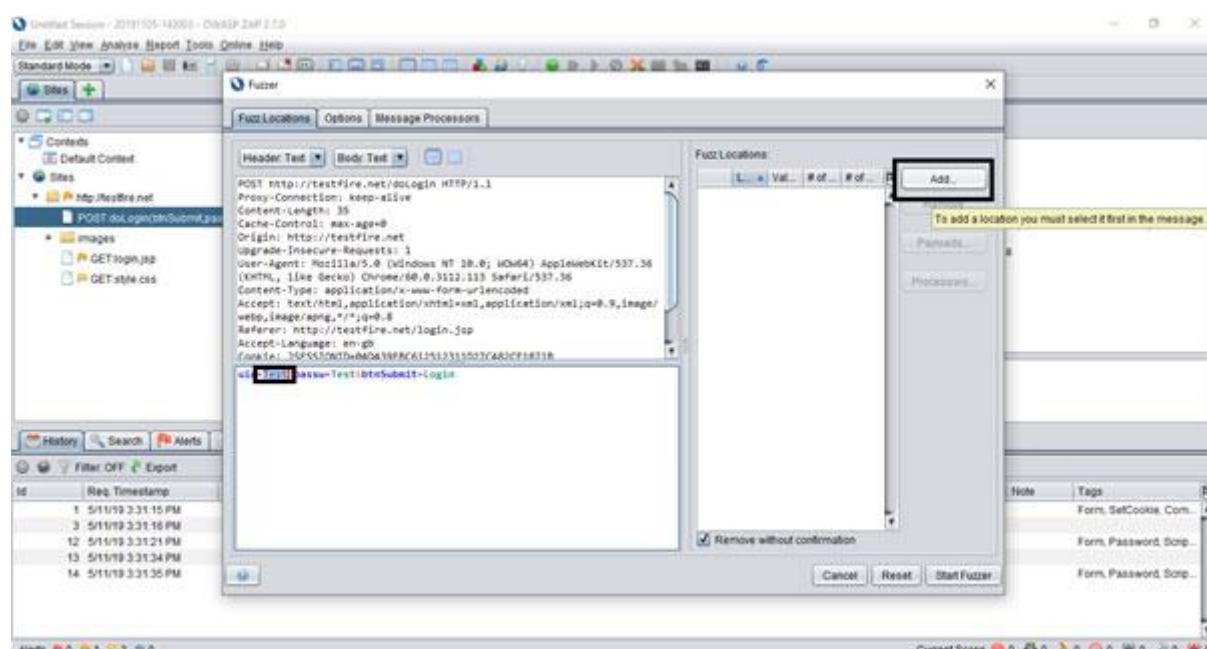
This screenshot is identical to the one above, showing the same POST request details and history log. The difference is that the 'Request' tab is highlighted in blue at the top of the interface.

#### 5. Now we will use 'Fuzz' functionality from ZAP, which is provided in the Attack section.

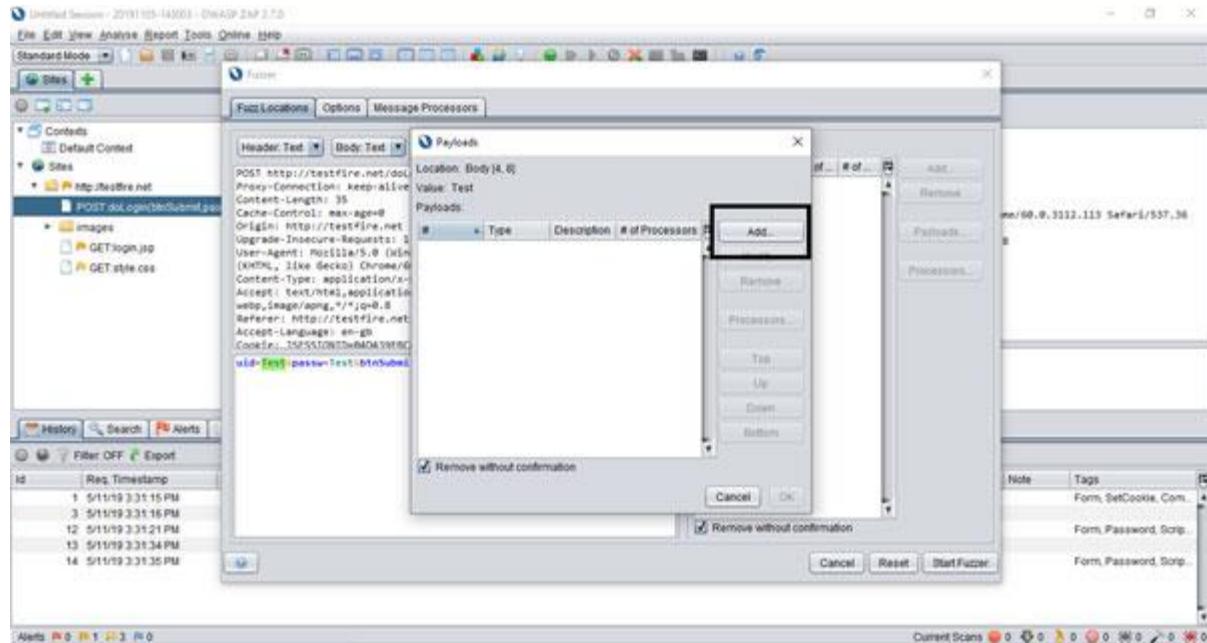
We need to right click on login API call and need to select Fuzz option as below



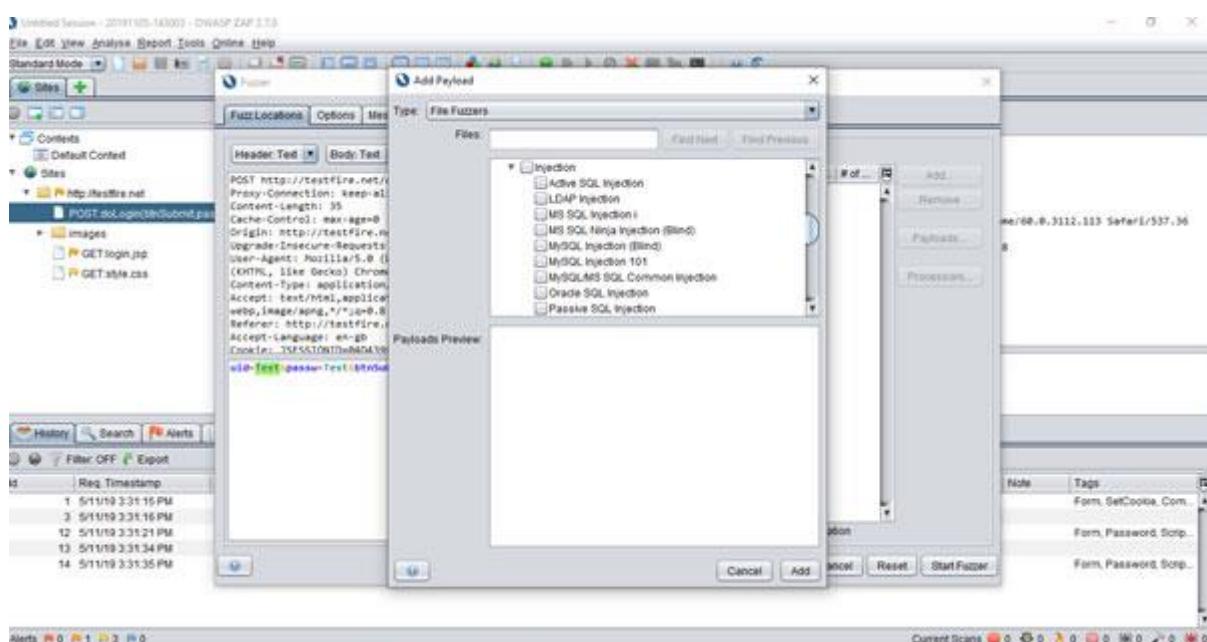
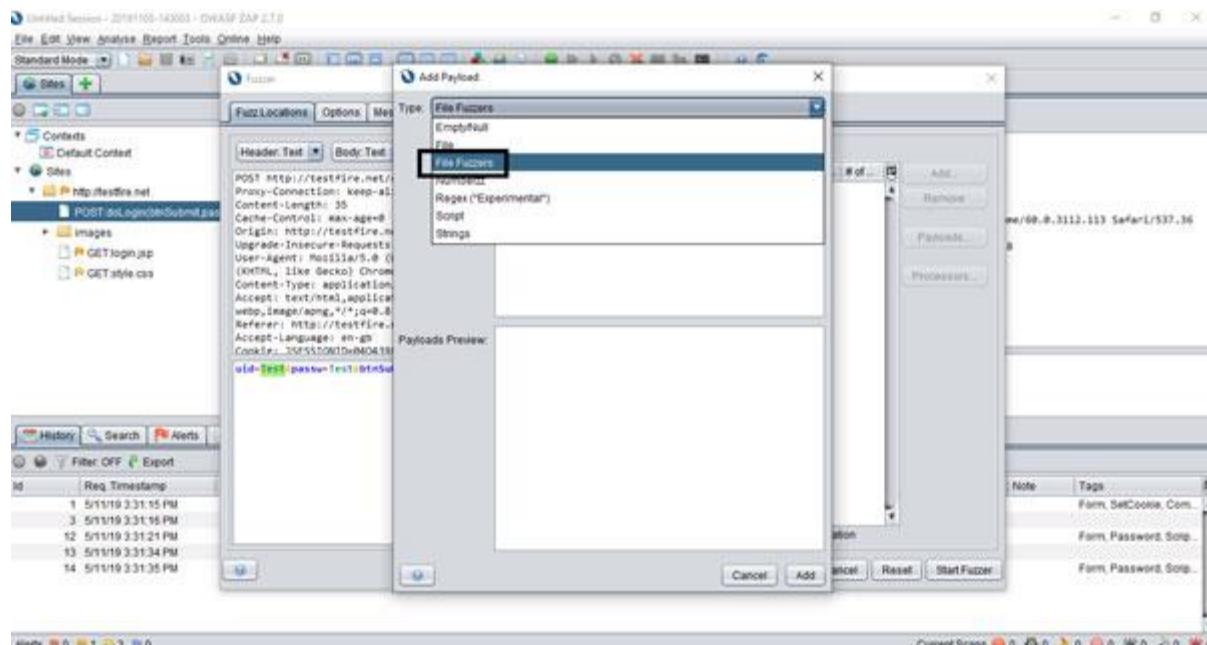
## 6. Select uid and click on 'Add'.



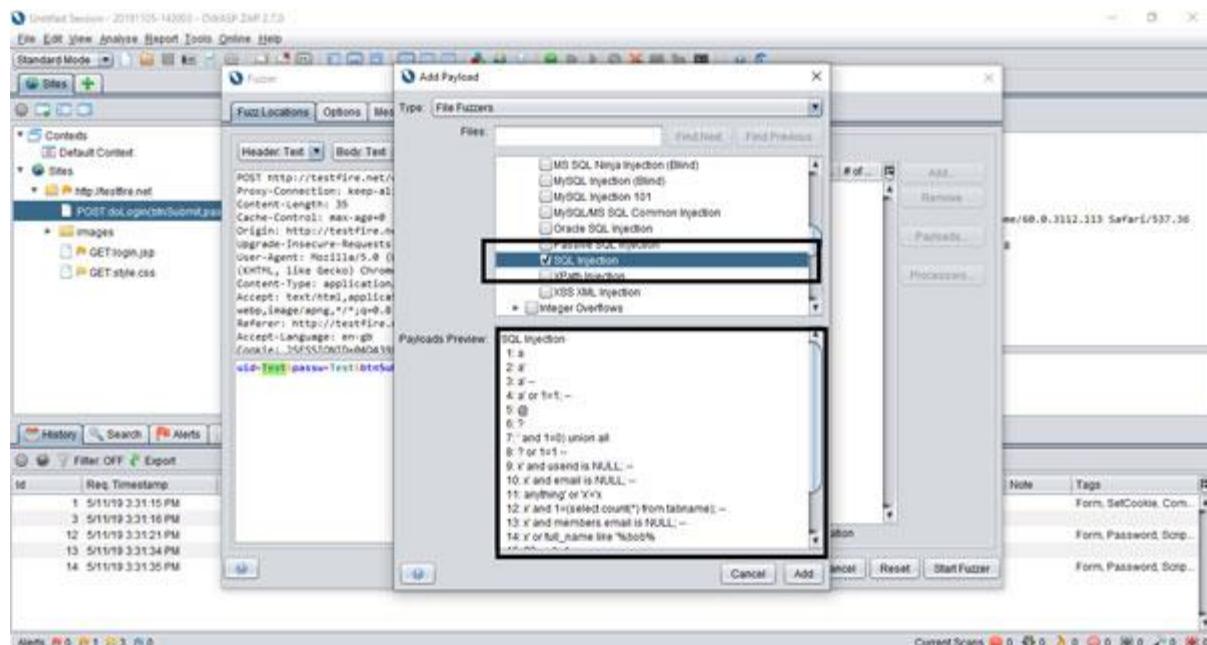
7. Now Payload window opens, click on 'Add'.



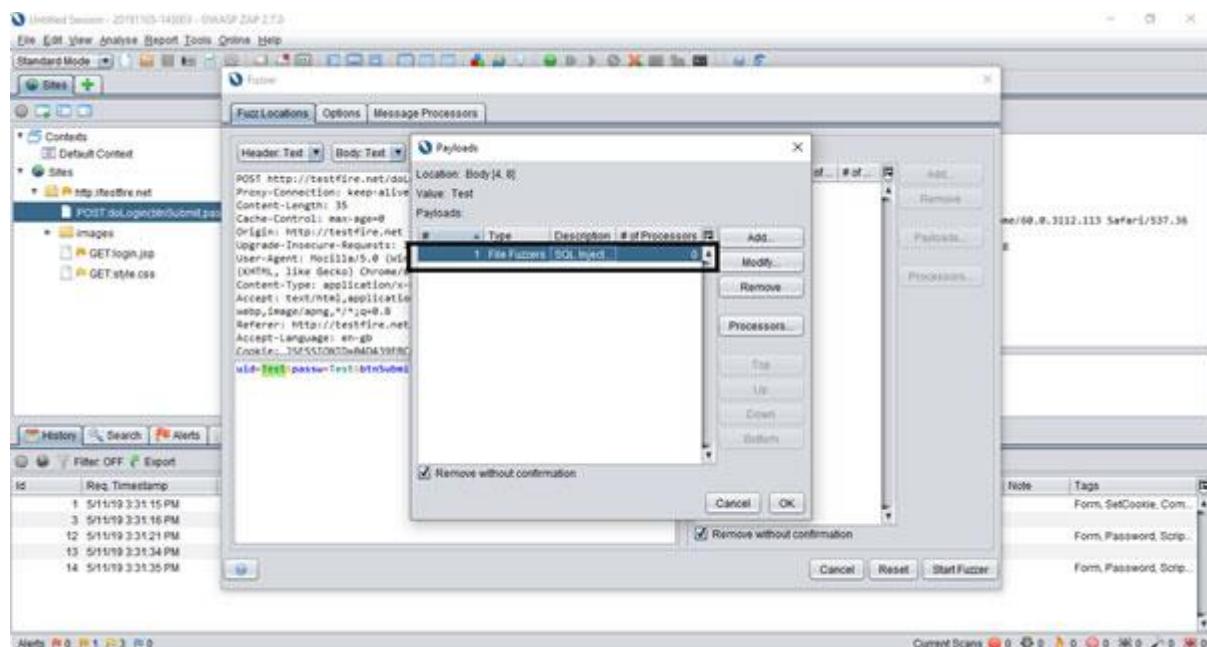
8. Select 'File Fizzers' from Type dropdown and expand the 3rd Then expand 'Injections'.



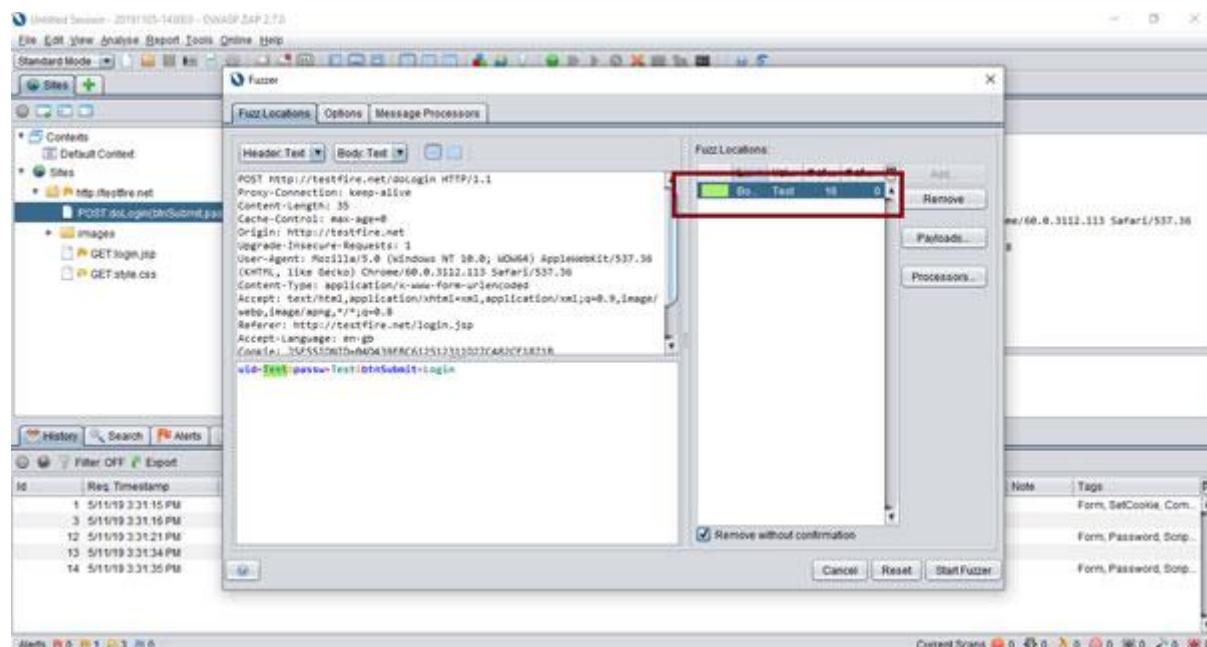
9. Scroll down the pane and select **SQL injection** as a payload for uid filed. Selected payloads must be seen in the Payloads Preview pane.



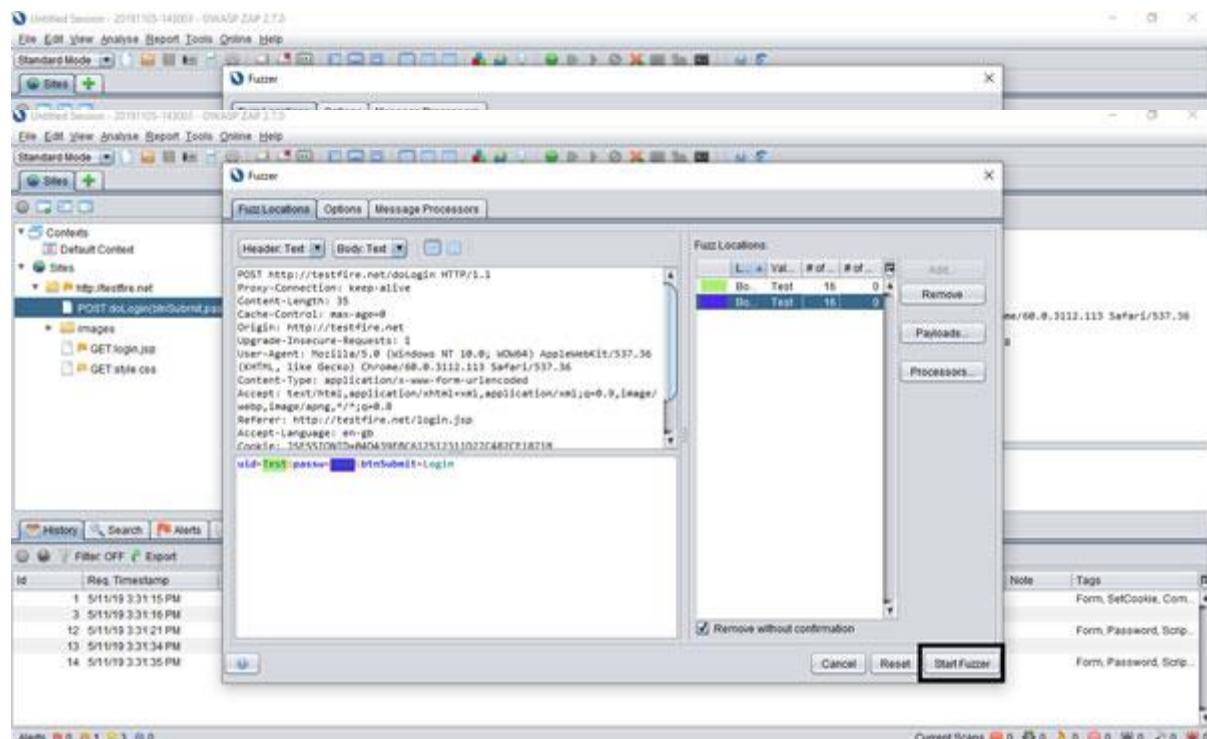
10. Click on the 'Add' button and observe that payload gets added for the uid field.



11. Click on Ok. Payload is now successfully added for the uid field.



12. Similarly, add payload for the 'passw' field.



13. Click on 'Start Fuzzer' and wait until it reaches 100%.

The screenshot shows the OWASP ZAP 2.7.0 interface. The top menu bar includes File, Edit, View, Analyse, Report, Tools, Online, Help, Standard Mode, and a toolbar with various icons. The main window has tabs for Header.Text, Body.Text, and a Request/Response view. The Request tab shows a POST request to `http://testfire.net/doLogin` with the following headers:

```
POST /doLogin HTTP/1.1
Host: testfire.net
Connection: keep-alive
Content-Length: 33
Cache-Control: max-age=0
Origin: http://testfire.net
Upgrade-Insecure-Request: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: http://testfire.net/login.jsp
Accept-Language: en-US
Cookie: JSESSIONID=00D4394BCA12512331D27CA82CE1871B
Host: testfire.net
```

The Body.Text tab contains the payload: `uid=Test!pass=Text!btnSubmit+login`. Below the tabs is a navigation bar with History, Search, Alerts, Output, Fuzzer, and a status message "Current fuzzers: 0". The main content area displays a table of fuzzing results:

Task ID	Message Type	Code	Reason	RTT	Size Resp. Header	Size Resp. Body	Highest Alert	Status	Payloads
0	Original	302	Found	682 ms	199 bytes	0 bytes			a; a
1	Fuzzed	302	Found	656 ms	264 bytes	0 bytes			a; a'
2	Fuzzed	302	Found	650 ms	264 bytes	0 bytes			a; a'--
3	Fuzzed	302	Found	642 ms	264 bytes	0 bytes			a; a' or !=; --
4	Fuzzed	302	Found	661 ms	264 bytes	0 bytes			a; @
5	Fuzzed	302	Found	639 ms	264 bytes	0 bytes			a; ?
6	Fuzzed	302	Found	292 ms	264 bytes	0 bytes			a; ' or !=0 union...
7	Fuzzed	302	Found	292 ms	264 bytes	0 bytes			a; ? or !=1 --
8	Fuzzed	302	Found	305 ms	264 bytes	0 bytes			a; X and userid is
9	Fuzzed	302	Found	309 ms	264 bytes	0 bytes			

Finally, check the status code. If the code returns 302, i.e., redirected to next page, it means that the application is vulnerable to an SQL injection. We can utilize the same SQL queries in 'uid' and 'passw' fields and login into an application without knowing the actual password.

## **PRACTICAL -10**

**Aim : Perform web application testing using DVWA .**

- 1. Perform Manual SQL injection**
- 2. XSS using DVWA**

**DVWA: Damn Vulnerable Web Application:**

- Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable.
- Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.
- The aim of DVWA is to practice some of the most common web vulnerabilities, with various levels of difficulty, with a simple straightforward interface.
- Note, there are both documented and undocumented vulnerabilities with this software. This is intentional. You are encouraged to try and discover as many issues as possible.

**DVWA Attacks:**

- Brute Force
- Command Injection
- File Upload
- Insecure CAPTCHA
- SQL Injection / SQL Injection (Blind)
- Weak Session IDs

## Demonstration

- In order to exploit SQL injection vulnerabilities, we need to figure out how the query is built in order to inject our parameter in a situation that the query will remain true.
- For example, in the DVWA we can see a text field where it asks for user ID. If we enter the number 1 and we click on the submit button we will notice that it will return the first name and the surname of the user with ID=1.

## Vulnerability: SQL Injection

User ID:

Submit

ID: 1  
First name: admin  
Surname: admin

- The next step will be to try to identify what kind of database is running on the back-end in order to construct the queries accordingly and to extract the information that we want.
- This is very important because If we don't know the database that exists behind we will not be able to exploit successfully the SQL injection vulnerability.
- Most of the times the web application technology (Java, ASP.NET, PHP etc.) will give us an idea of the database that the application is using. For example, ASP.NET applications often use Microsoft SQL Server, PHP applications likely to use MySQL and Java probably Oracle or MySQL.
- Additionally, we can assume the database type from the web server and operating system of the target.
- For example, if the web server is running Apache and PHP and it is a Linux host then the database has more possibilities to be MySQL.
- If it is an IIS then it is probably Microsoft SQL Server.
- Of course, we cannot rely on these information, this is just for giving us an indication in order to speed the database fingerprint process.
- We can very easily identify the database type especially if we are in a non-blind situation.
- The basic idea is to make the database to respond in a way that it will produce an error message that it will contain the database type and version. For example, this can be achieved by a single quote because it will force the

database to consider any characters that are following the quote as a string and not as SQL code and it will cause a syntax error.

- So now if we add a single quote on the vulnerable parameter id=' this will make the database to generate an error message which as we can see from the image below it contains the database type which is MySQL server.



## Vulnerability: SQL Injection

User ID:

' union select 1,@@version#

ID: ' union select 1,@@version#  
First name: 1  
Surname: 5.0.51a-3ubuntu5

## **PRACTICAL -11**

**Aim : Perform brute force attack using John the RIPPER.**

### **Brute Force Attack**

- A brute force attack can manifest itself in many ways, but primarily consists in an attacker configuring predetermined values, making requests to a server using those values, and then analyzing the response.
- For the sake of efficiency, an attacker may use a dictionary attack (with or without mutations) or a traditional brute-force attack (with given classes of characters e.g.: alphanumeric, special, case (in)sensitive).
- Considering a given method, number of tries, efficiency of the system which conducts the attack, and estimated efficiency of the system which is attacked the attacker can calculate approximately how long it will take to submit all chosen predetermined values.
- For Example, the matching string that you are using for cracking passwords should include uppercase alphabets, special characters and numbers like ABC32@\$
- The user gets a password on the successful match, but this effective process is slow. For example, a 10-character password including upper and lower letters along with numbers and special characters will take over 10 years to be guessed by a computer.
- The penetration testers, ethical hackers, security experts and other Cyber Security professionals use this tool to find weak algorithms and then make them strong so that they can't be hacked.
- Security professionals build their confidential files with a strong hash algorithm to prevent external unauthorized access.
- Hackers used it to crack multiple accounts and simply crack their credentials.
- Security experts use it to strengthen their encryption.
- It can also be used for hacking shells and passwords.
- SHA-crypt hashes.
- It provides a mangling feature which is a pre-processor in JTR that optimizes the word list to make the password cracking process faster.

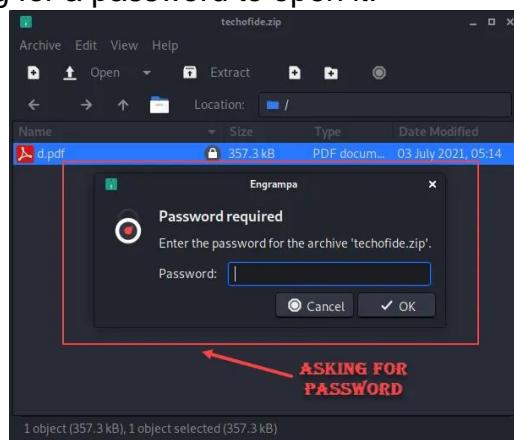
## John the RIPPER

- John the Ripper is the name of the password cracker tool that is developed by Openwall. As the name, it is used to crack password hashes by using its most popular inbuilt program, rules and codes that are also an individual password cracker itself in a single package.
- It automatically detects types of password hashes; you can also customize this tool according to your wish.
- It can be used to crack password-protected compressed files like Zip, Rar, Doc, pdf etc.
- The main objective of John the Ripper is to crack the password.
- There are many ways that can be supported but it is mainly known for Dictionary attacks. However, you can also run other types of attacks like Brute force attack, Rainbow Table etc.

## Practical/Demonstration

### 1. Cracking a ZIP file:

- a) Now you can see that we have a zip file techofide.zip which is password protected and asking for a password to open it.



- b) Now as we know JTR use hash to crack password, so we first need to generate a hash of our zip file. The below command will generate a hash of our techofide.zip file and store that generated hash value into a hash.txt file

***sudo zip2john techofide.zip > hash.txt***

```
(kanav@Techofide)-[~/Desktop]
$ sudo zip2john techofide.zip > hash.txt
[sudo] password for kanav:
ver 2.0 efh 9901 techofide.zip/d.pdf PKZIP Encr: cmplen=340694, decmplen=357334, crc=CA455687
```

c) Let us break it with our tool, so now we have a hash of our zip file that we will use to crack the password. In the below command we use the format option to specify the zip file and then the hash.txt file where we store our hash value.

***sudo john --format=zip hash.txt***

```
(kanav@Techofide)-[~/Desktop]
$ sudo john --format=zip hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 256/256 AVX2 8x])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 3 candidates buffered for the current salt, minimum 16 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 16 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 6 candidates buffered for the current salt, minimum 16 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
54321          (techofide.zip/d.pdf)
1g 0:00:00:01 DONE 2/3 (2021-07-07 12:26) 0.9615g/s 34925p/s 34925c/s 34925C/s 123456 .. Peter
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

In the above picture, you can see our command complete the session and returns with the correct password 54321.

## 2. Cracking MD5 Password:

- i) In this example I am generating a hash by using md5 hash generator to show you how to crack MD5 formatted files password. In the below image you can see I have generated the hash of the 12345 strings. You can copy the MD5 hash to perform the same practical.
- ii) In the below picture you can see the file sha1.txt. I have used the cat command to show you the data of the sha1.txt file, You can see the MD5 hash value 8772cc.

```
(kanav@Techofide)-[~/Desktop]
$ cat sha1.txt
827ccb0eea8a706c4c34a16891f84e7b
```

## MD5 Hash Generator

Use this generator to create an MD5 hash of a string:

12345

Generate →

Your String

12345

MD5 Hash

827ccb0eea8a706c4c34a16891f84e7b

Copy

SHA1 Hash

8cb2237d0679ca88db6464eac60da96345513964

Copy

- iii) Now let us crack the MD5 Hash, In the below command we have specified format along with the hash file.

**john sha1.txt --format=RAW-MD5**

```
(kanav@Techofide)-[~/Desktop]
$ john sha1.txt --format=RAW-MD5 ← COMMAND
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
12345 (?) ← Password
1g 0:00:00:00 DONE 2/3 (2021-07-07 16:17) 3.225g/s 1238p/s 1238c/s 1238C/s 123456 ..larry
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed
```

In the above screenshot, you can see the output that cracks the hash and returns the 12345 password.

## Conclusion: -

Now we know what is John the Ripper, how to use John the Ripper, How John the Ripper password cracker works, how passwords can be cracked and also a tutorial on its real-life important uses, but this not get over yet there are lots of other things that can be done by JTR.7

However, remember that if the password is long, it will also take long time to crack.