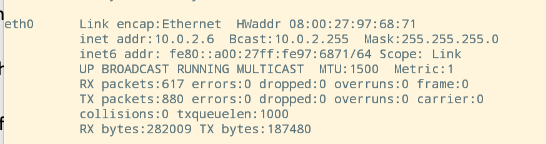
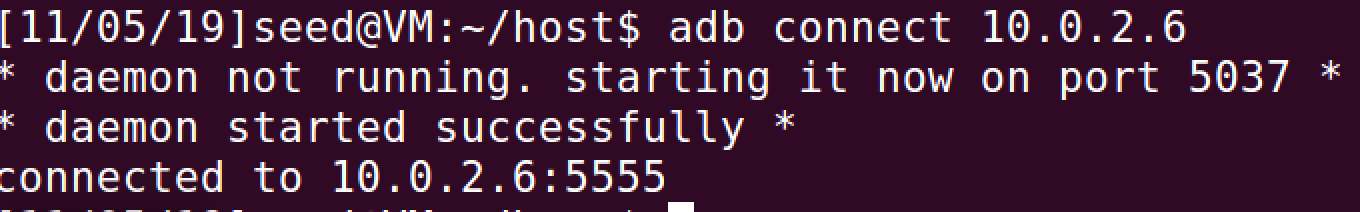
Android Repackaging Attack Lab

**Task1: Obtain an Android App (APK file) and Install It**

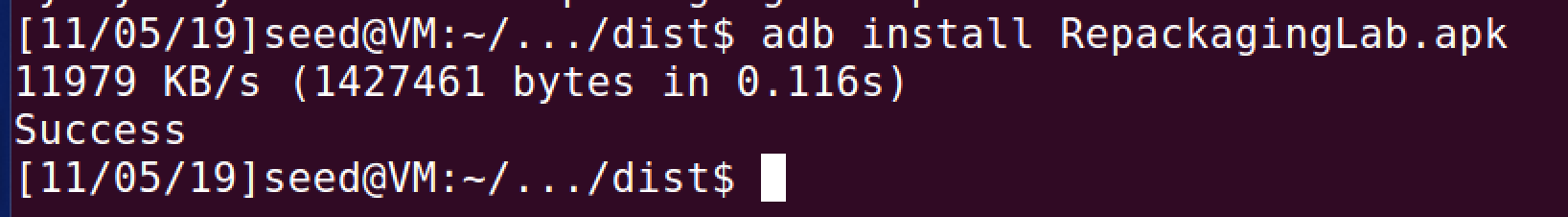
Check the IP address of the android VM.



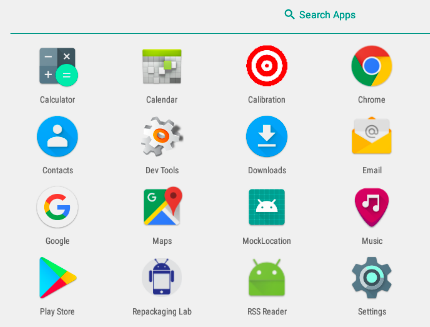
Connect with android VM with command adb connection IP address



Install the apk file with command adb install file.apk

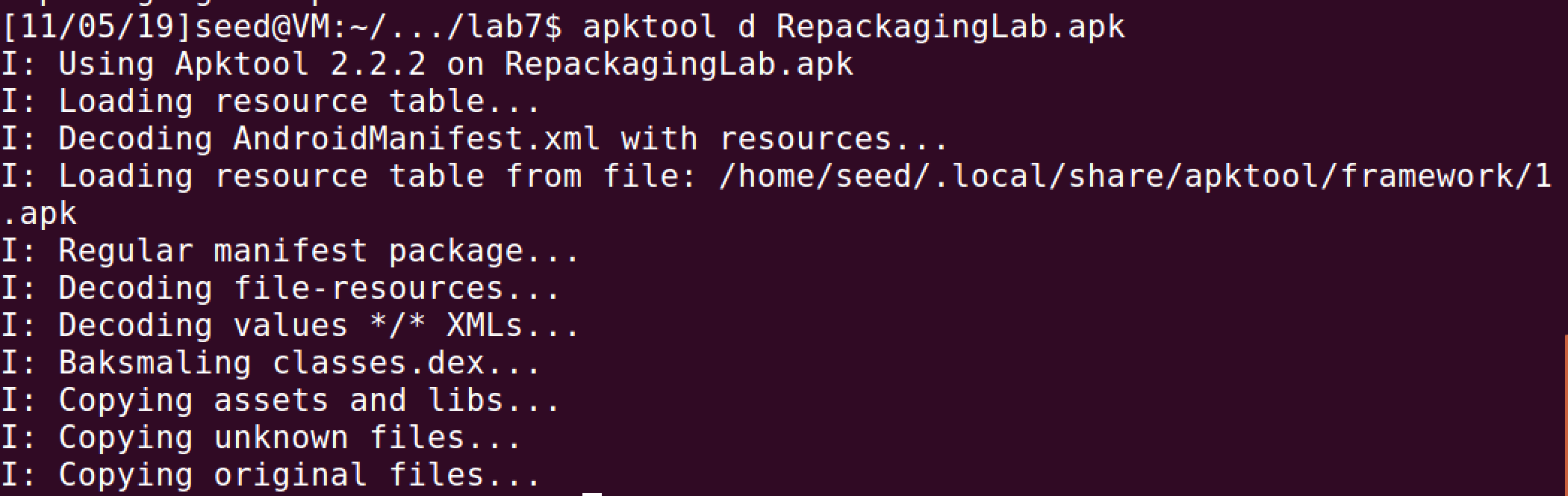


And we can see that we have installed successfully.



**Task2: Disassemble Android App**

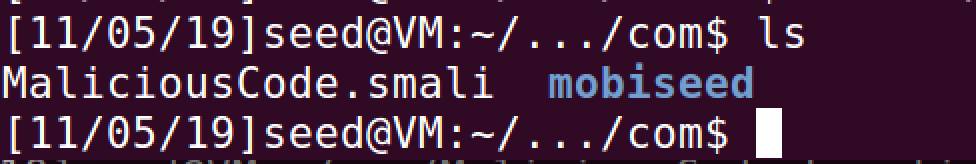
Disassemble the APP: apktool d file.apk

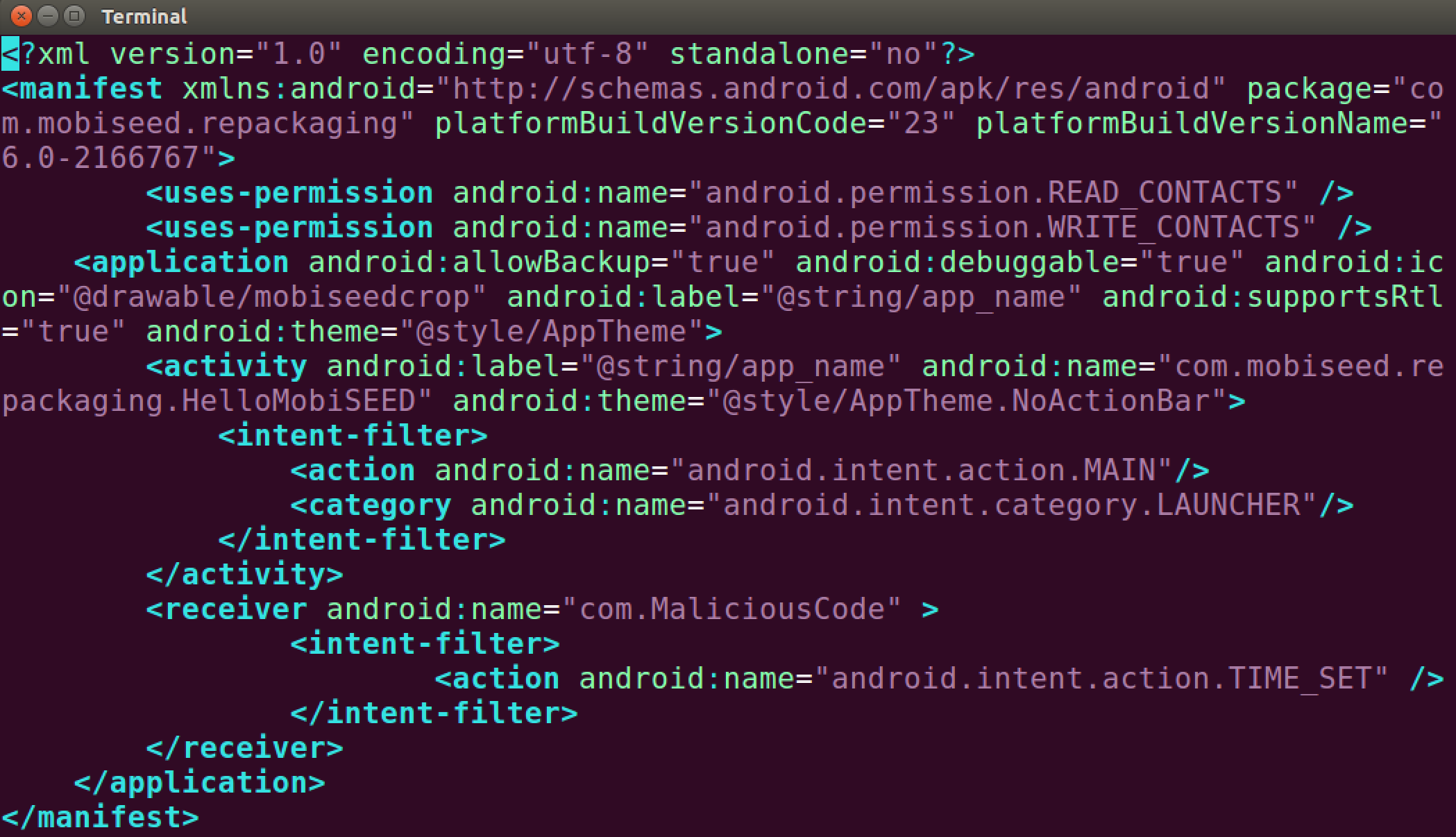


Observation: We disassemble the apk file using the apktool with d.

**Task3: Inject Malicious Code**

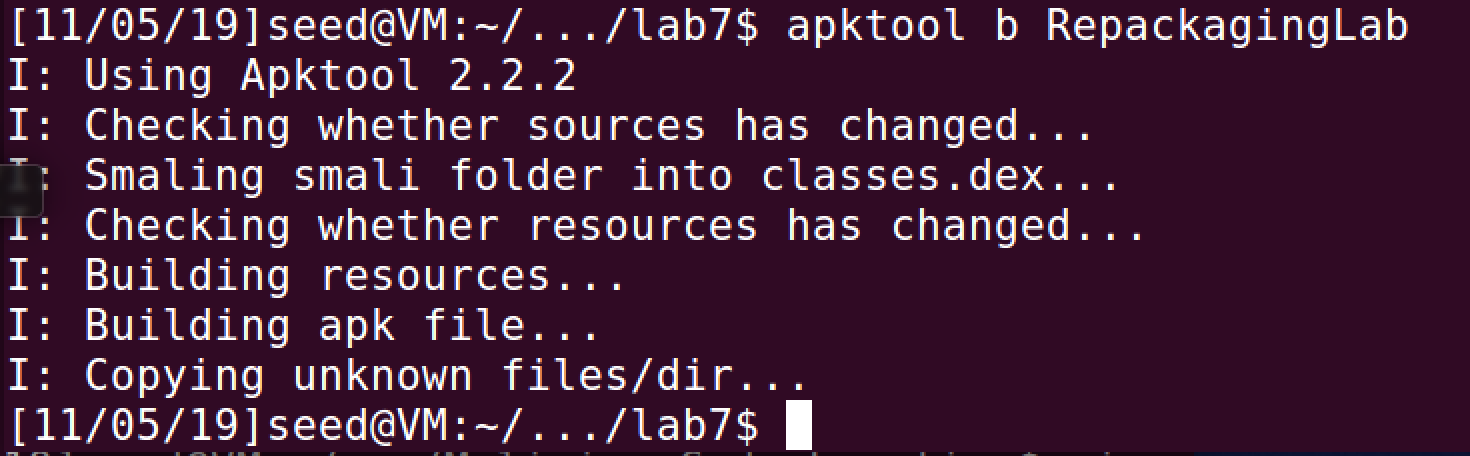
First: We download the smali code and place it directly into the com folder of the disassembled apk file.



Second: we modify the AndroidManifest.xml file by giving it sufficient permissions for our attack to work.

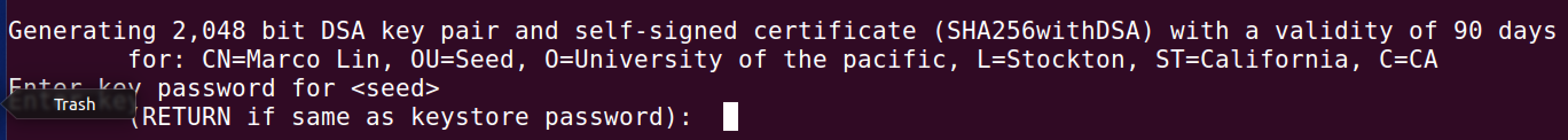
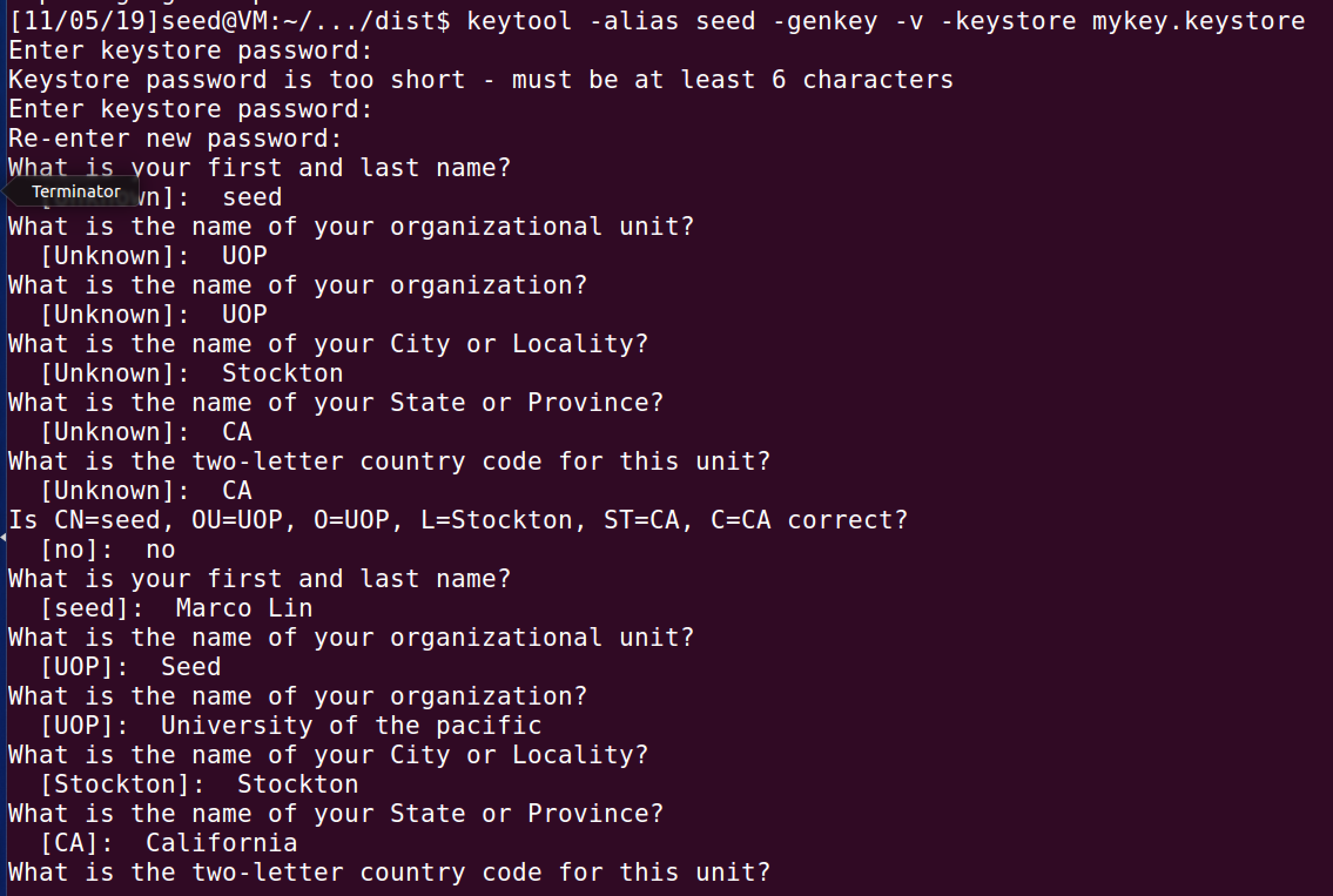
**Task4: Repack Android App with Malicious Code**

Step1: We repack our Android app by using the apktool with b option and in the folder

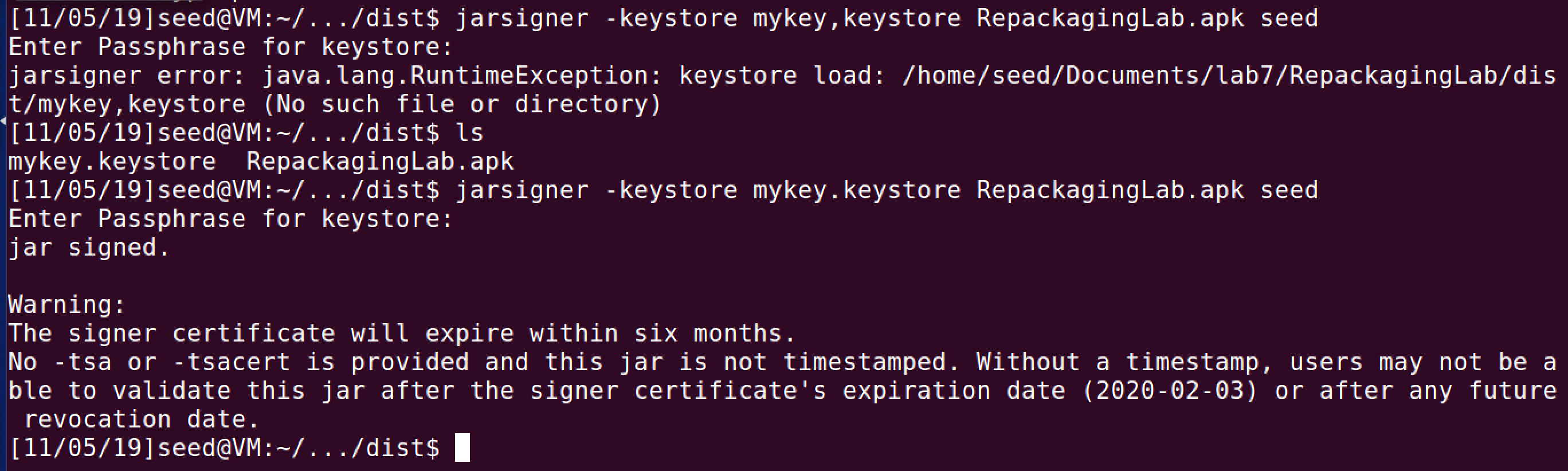
which contains the necessary code for the apk file.

Step2:

a. We generate the public and private key and digital certificate using the above commands as shown in the screenshots.

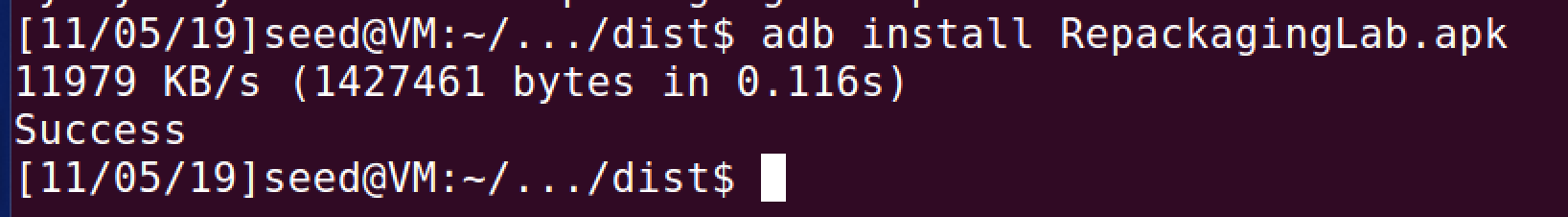


1. Using Jarsigner to sign the APK file using the key generated in the previous step. The command jarsigner prompts the user to enter the password, which is needed for accessing the keystore. It then use the key (identified by the alias name) to sign the APK file.

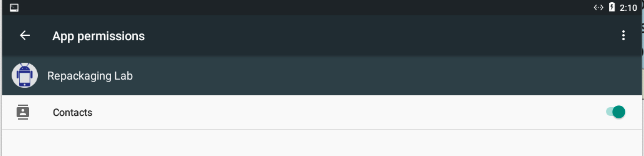


**Task5: Install the Repackaged App and Trigger the Malicious Code**

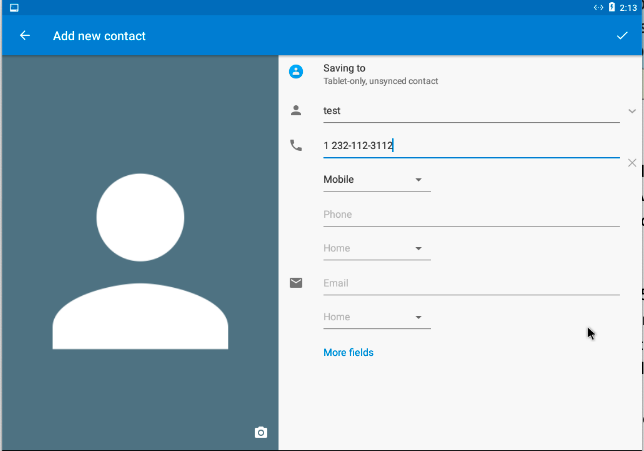
1. Delete the apk from android VM first and Install the apk again.



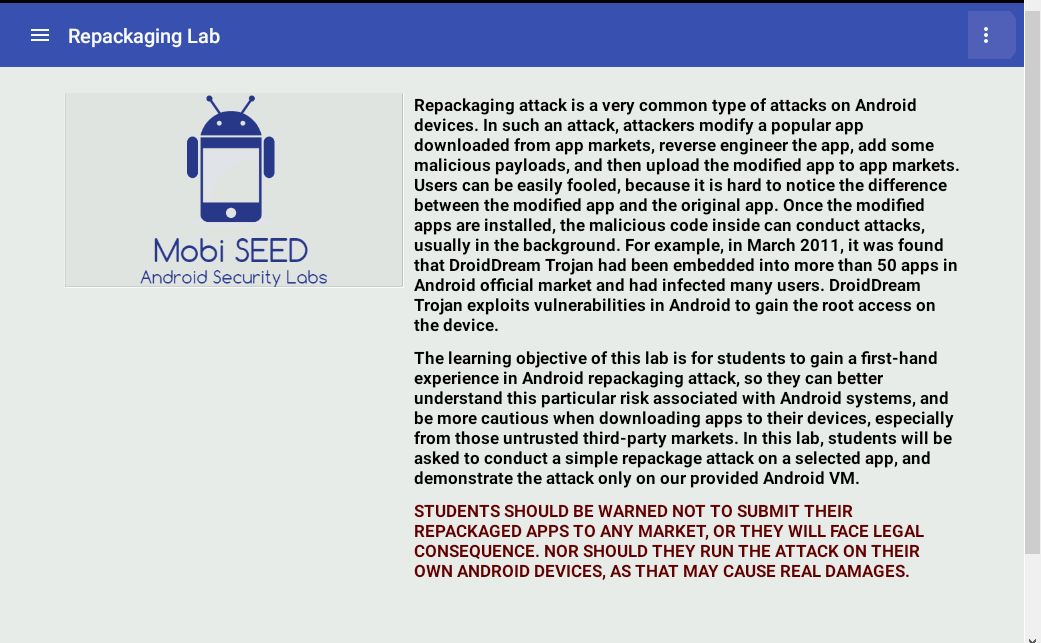
1. On android VM, give the application permission to access contacts.

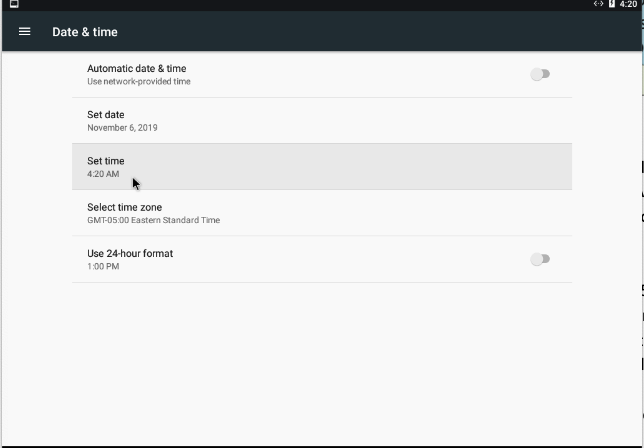


1. Add some new contact

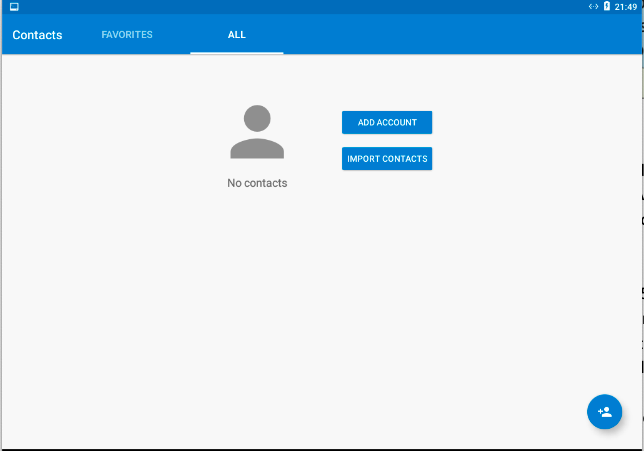


1. To demonstrate whether the attack works, we just need to run the application once, add a few contacts in the Contacts app and change the time on the android VM.



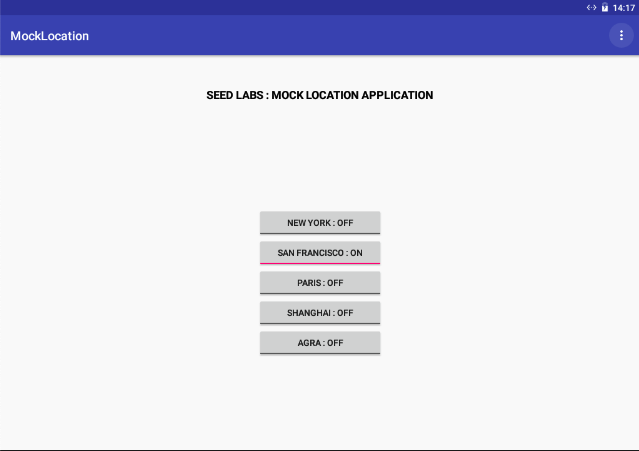


Result: all contacts have deleted by malicious code.



**Task6: Using Repackaging Attack to Track Victim’s Location**

Step1: Setting up mock locations



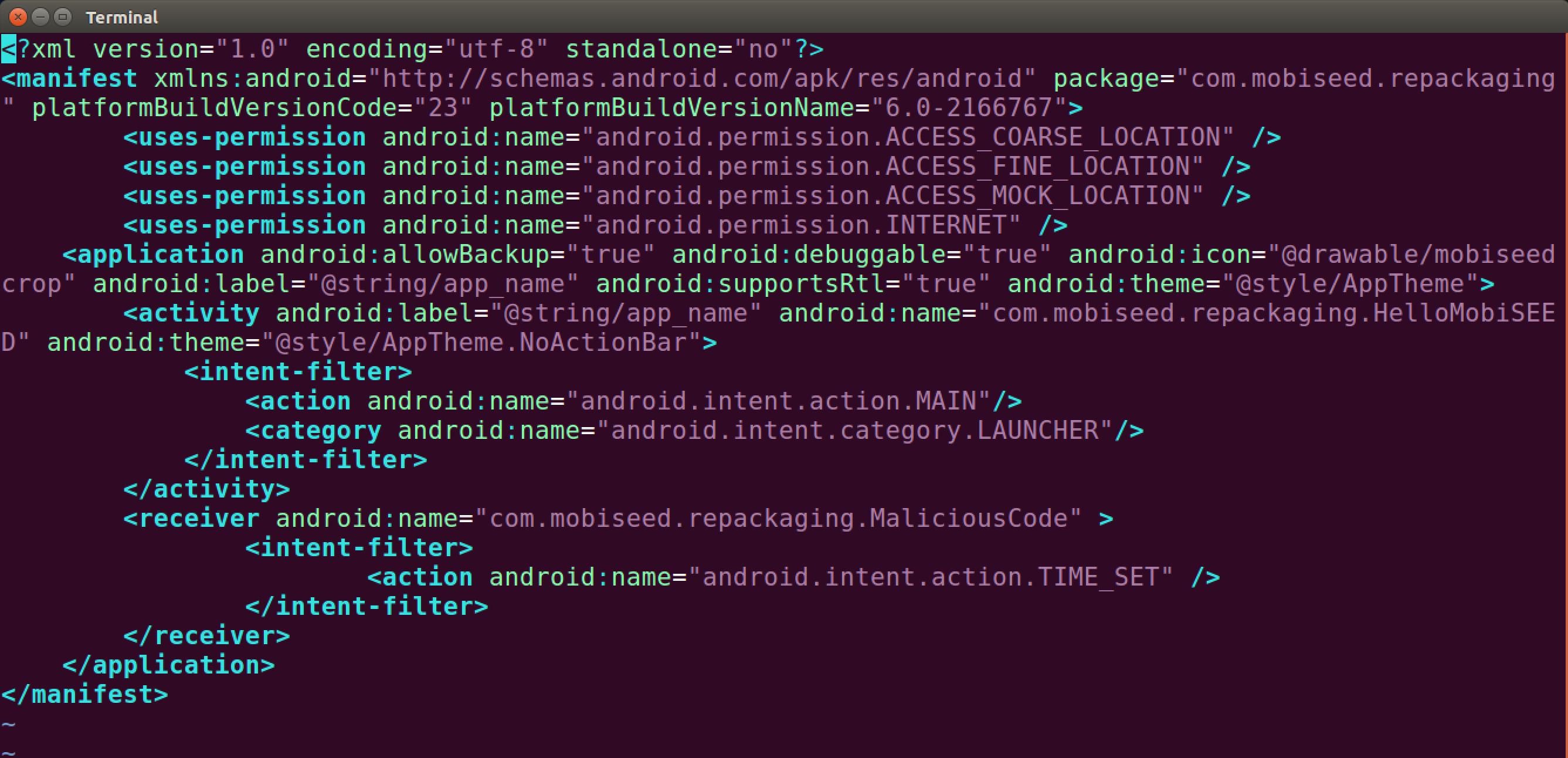
Step2: Configuring DNS

The malicious code in the repackaged app will send out the victim’s coordinates to the attacker’s server at www.repackagingattacklab.com. We are going to use the SEED Ubuntu VM to host this server. Therefore, we need to map the hostname to the Ubuntu VM’s IP address. The easiest way to set this up is to add a line to the /system/etc/hosts file on the Android VM.

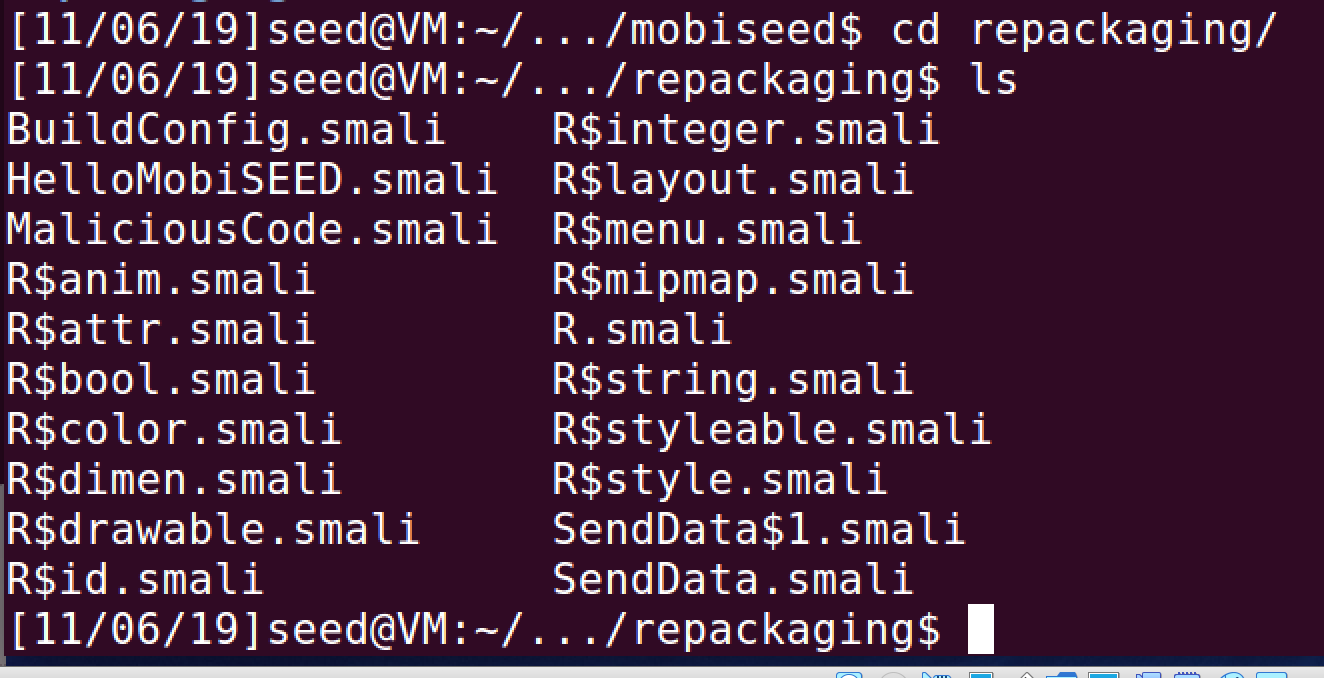


Step3: Repackaging and Installing the victim APP

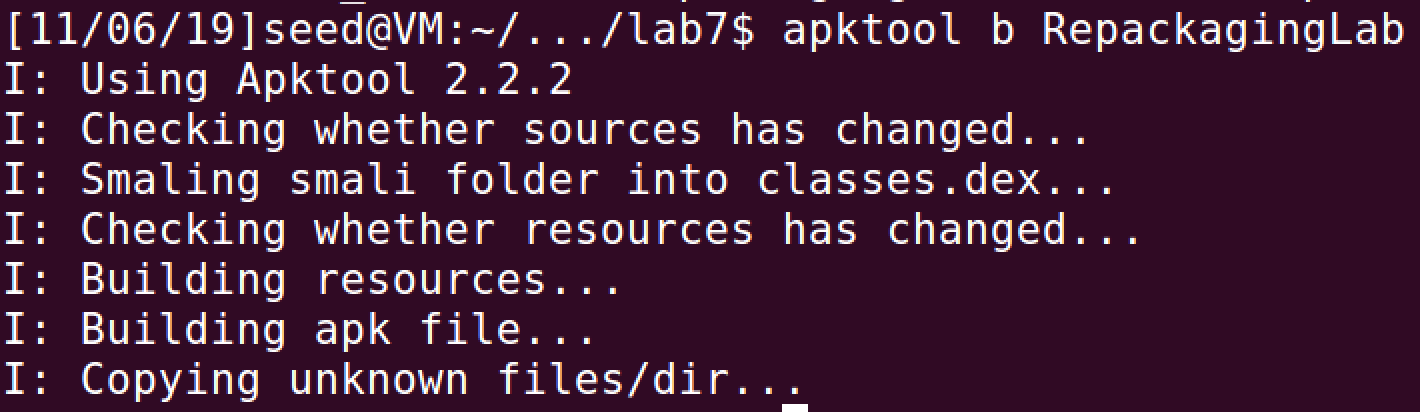
1. Modify the AndroidManifest.xml to give different permissions which are related to location and internet access



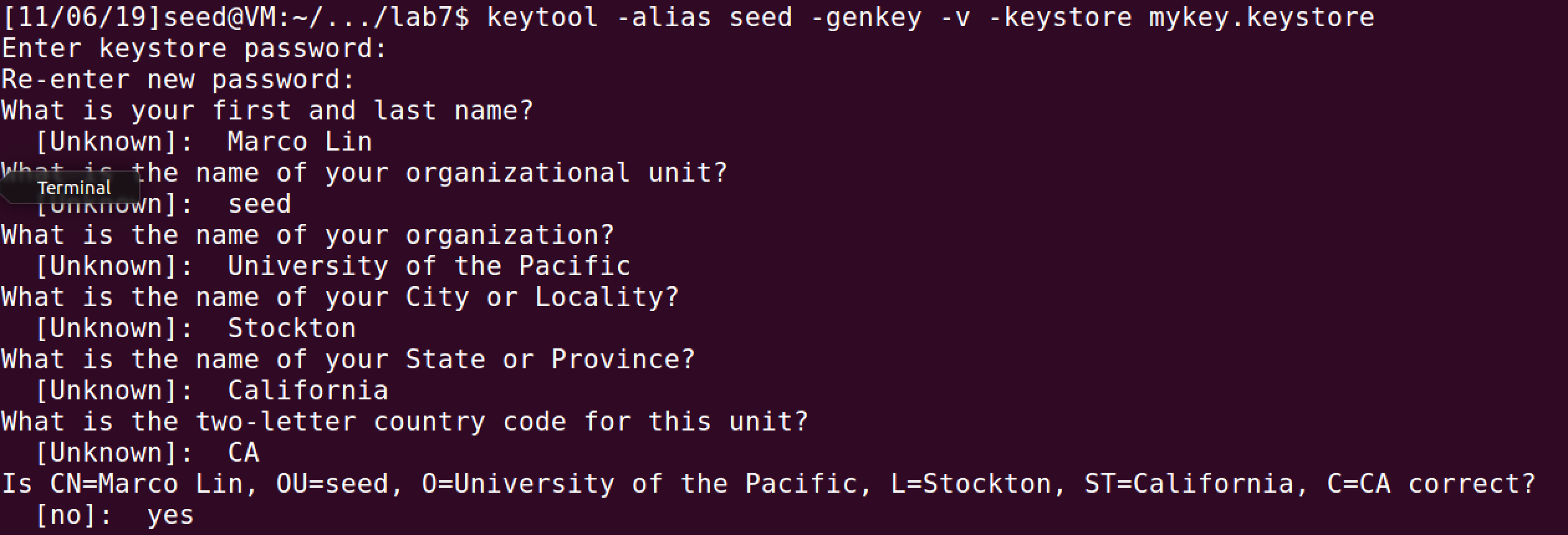
1. Place three files in the smali/com/mobiseed/repackaging folder of the unpacked application.

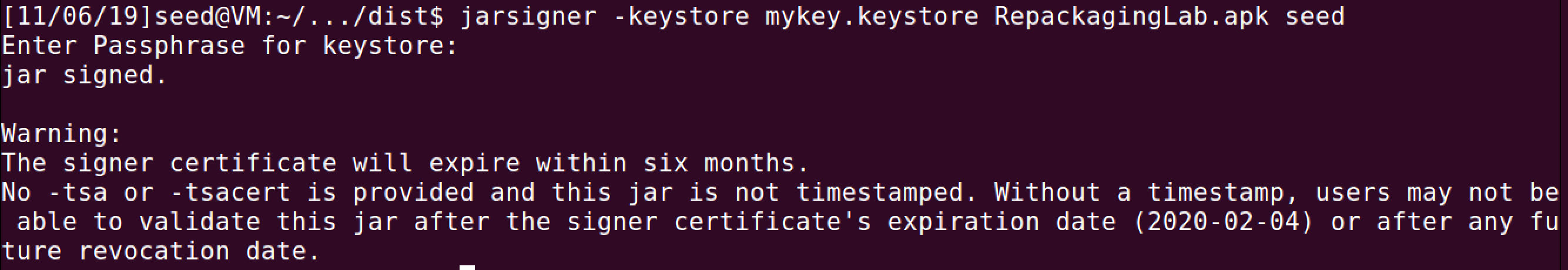


1. Repackaging the application

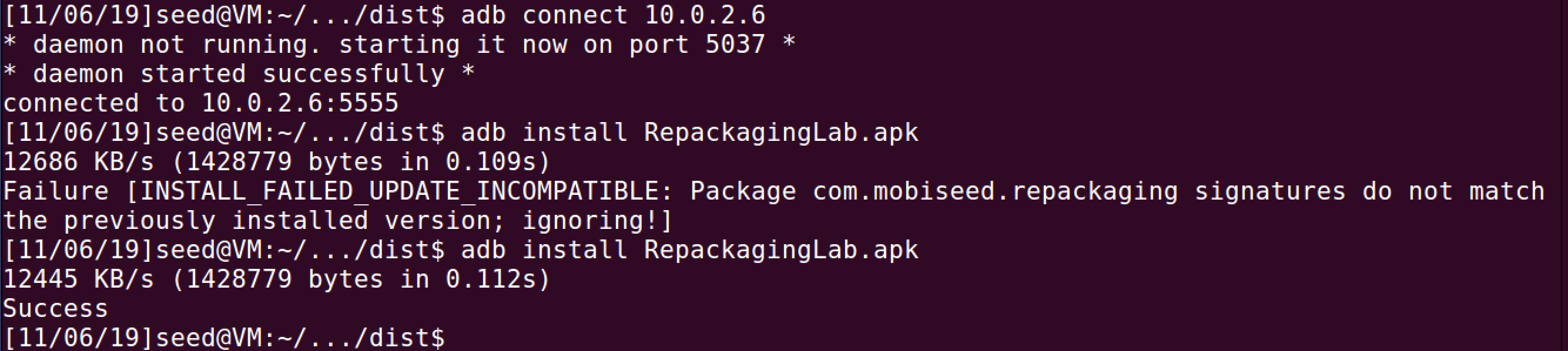


1. Sign the APK file again

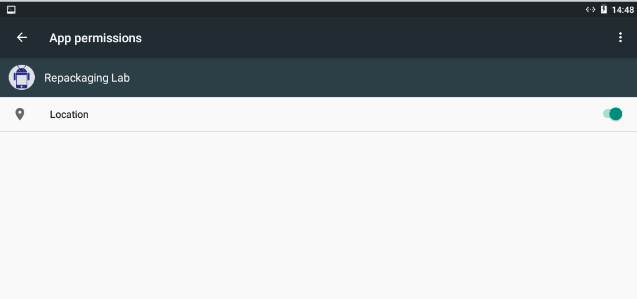




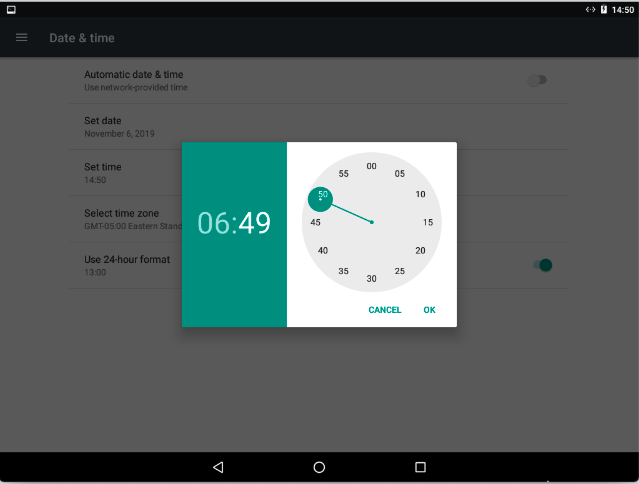
1. install the APK again



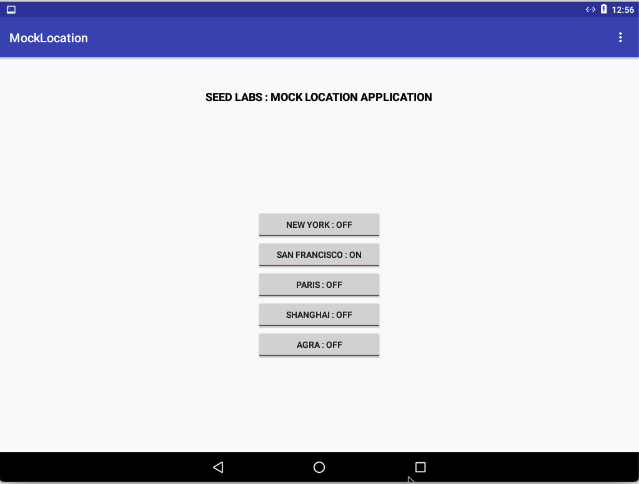
Step 4: Enabling the permission on the android VM

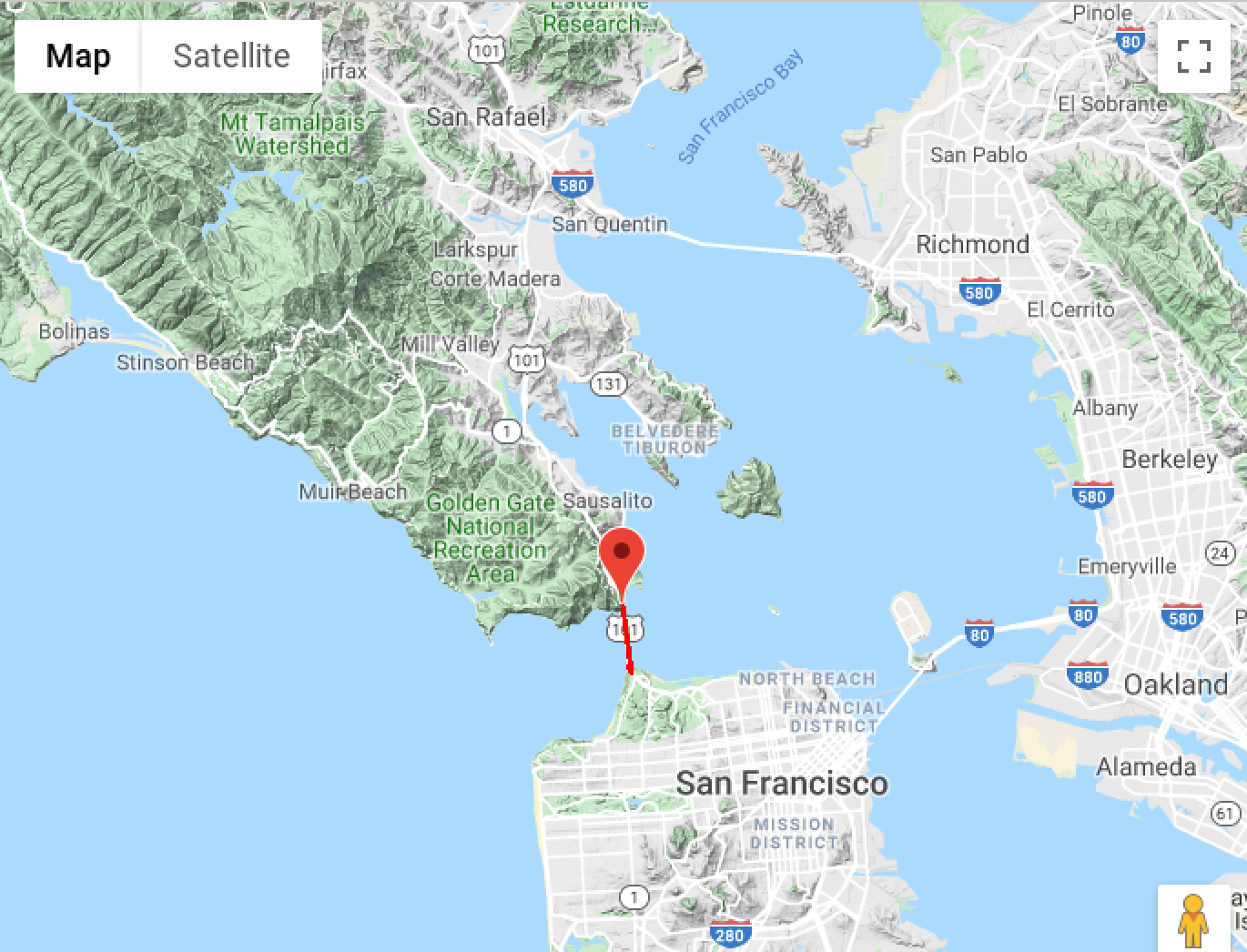


Step5: Triggering the attacking code by setting time



Step 6: Tracking the victim





Observations: We can track the location of the user whenever the location is changed.