Malicious APK File Analysis

No. 7

Analyzing a Malicious Android App

Tools Used: Androguard

OS Platform: Santoku

The first thing we do is figure out if the sample we are analyzing has the correct format. APK packages are nothing more than ZIP files with a predefined structure (including for example a manifest file).

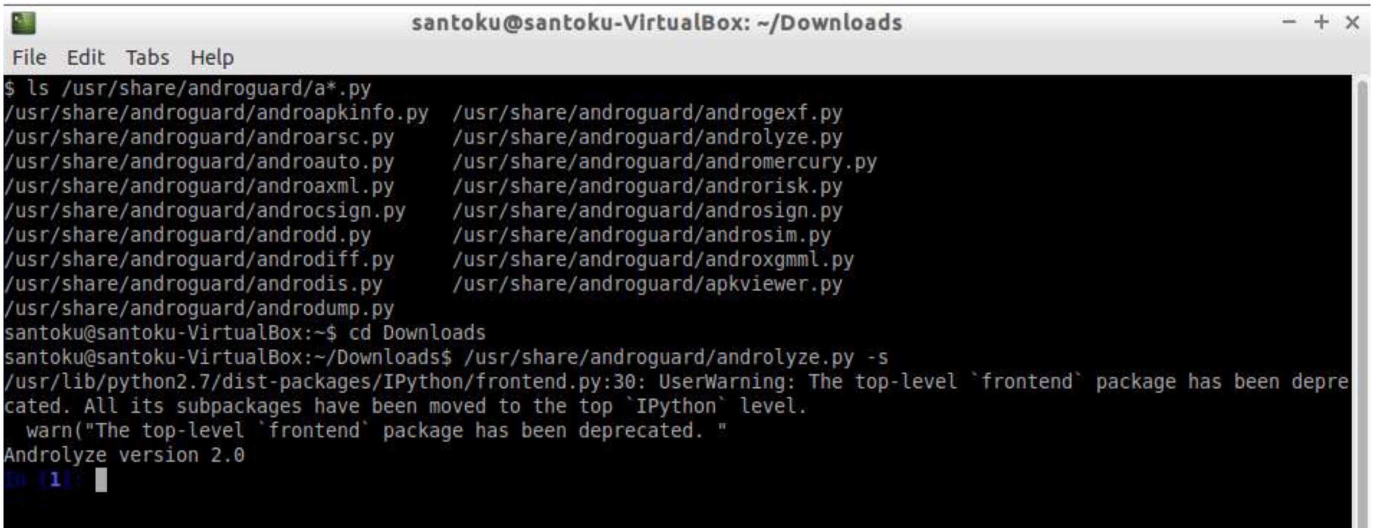
To get started with the analysis of the android app, we need to decompile the APK.

Androguard works with three different decompilers:

- dex2jar
- dad
- ded

For interactive analysis of the .apk, run the Androlyze tool of Androguard as follows, this will start an interactive Ipython shell as shown in the screen shot:

/usr/share/androguard/androlyze.py –s

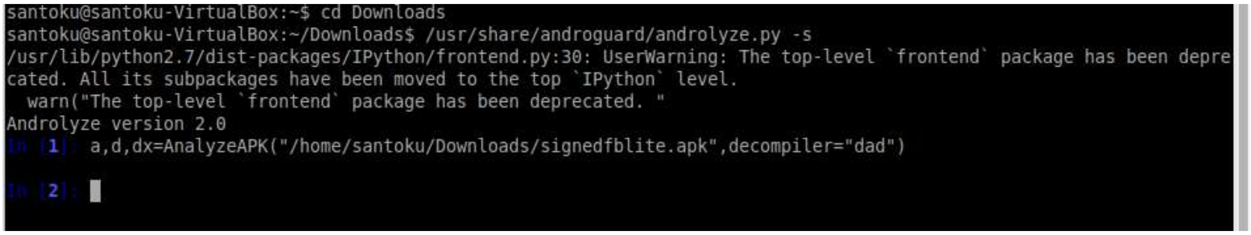


```
$ ls /usr/share/androguard/a*.py
/usr/share/androguard/androapkinfo.py  /usr/share/androguard/androgexf.py
/usr/share/androguard/androarsc.py     /usr/share/androguard/androlyze.py
/usr/share/androguard/androauto.py     /usr/share/androguard/andromercury.py
/usr/share/androguard/androaxml.py     /usr/share/androguard/androrisk.py
/usr/share/androguard/androcsign.py    /usr/share/androguard/androsign.py
/usr/share/androguard/androdd.py       /usr/share/androguard/androsim.py
/usr/share/androguard/androdiff.py     /usr/share/androguard/androxgmml.py
/usr/share/androguard/androdis.py      /usr/share/androguard/apkviewer.py
/usr/share/androguard/androdump.py
santoku@santoku-VirtualBox:~$ cd Downloads
santoku@santoku-VirtualBox:~/Downloads$ /usr/share/androguard/androlyze.py -s
/usr/lib/python2.7/dist-packages/IPython/frontend.py:30: UserWarning: The top-level `frontend` package has been depre
cated. All its subpackages have been moved to the top `IPython` level.
  warn("The top-level `frontend` package has been deprecated. "
Androlyze version 2.0
In [1]:
```

Next in the interpreter window load the apk file for analysis by running the following command:

a, d, dx = AnalyzeAPK("/home/santoku/Downloads/signedfblite.apk", decompiler = “dad”)

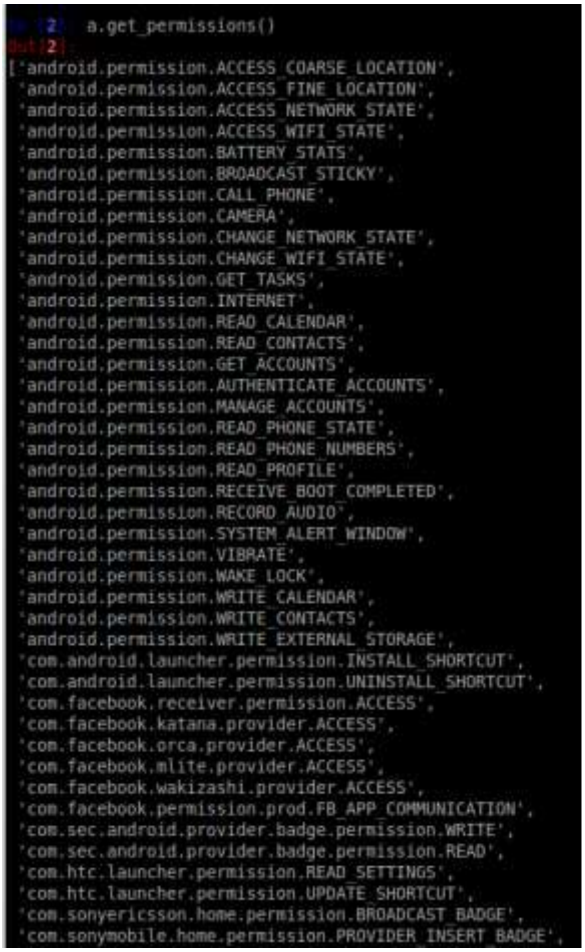
The wrapper functions AnalyzeAPK , returns three objects. a is an APK object, d is an array of DalvikVMFormat object and dx is an Analysis object.



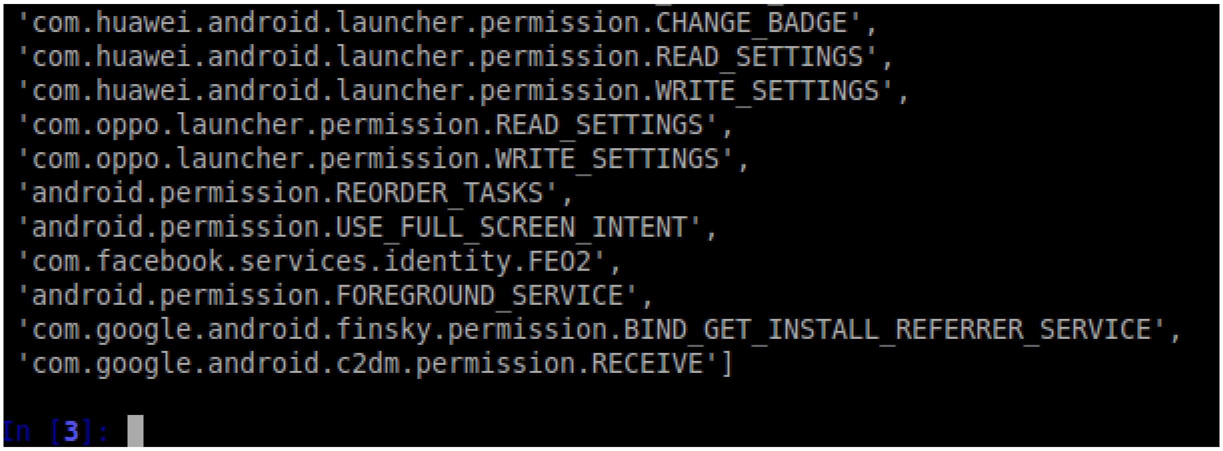
```
santoku@santoku-VirtualBox:~$ cd Downloads
santoku@santoku-VirtualBox:~/Downloads$ /usr/share/androguard/androlyze.py -s
/usr/lib/python2.7/dist-packages/IPython/frontend.py:30: UserWarning: The top-level `frontend` package has been depre
cated. All its subpackages have been moved to the top `IPython` level.
  warn("The top-level `frontend` package has been deprecated. "
Androlyze version 2.0
In [1]: a,d,dx=AnalyzeAPK("/home/santoku/Downloads/signedfblite.apk",decompiler="dad")

In [2]: 
```

Once we have the APK object, we can extract more detail information of the apk file.

- To get what kind of permissions the app needs, we execute the below mentioned command.
  Command: a.get_permissions()



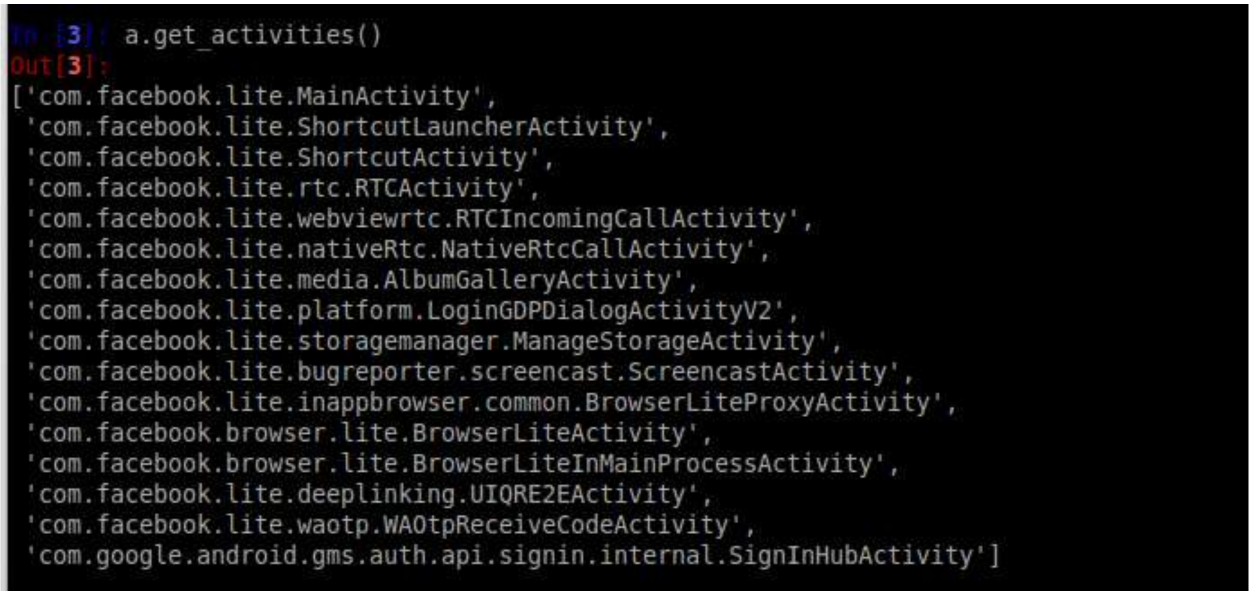
```
In [2]: a.get_permissions()
Out[2]:
['android.permission.ACCESS_COARSE_LOCATION',
 'android.permission.ACCESS_FINE_LOCATION',
 'android.permission.ACCESS_NETWORK_STATE',
 'android.permission.ACCESS_WIFI_STATE',
 'android.permission.BATTERY_STATS',
 'android.permission.BROADCAST_STICKY',
 'android.permission.CALL_PHONE',
 'android.permission.CAMERA',
 'android.permission.CHANGE_NETWORK_STATE',
 'android.permission.CHANGE_WIFI_STATE',
 'android.permission.GET_TASKS',
 'android.permission.INTERNET',
 'android.permission.READ_CALENDAR',
 'android.permission.READ_CONTACTS',
 'android.permission.GET_ACCOUNTS',
 'android.permission.AUTHENTICATE_ACCOUNTS',
 'android.permission.MANAGE_ACCOUNTS',
 'android.permission.READ_PHONE_STATE',
 'android.permission.READ_PHONE_NUMBERS',
 'android.permission.READ_PROFILE',
 'android.permission.RECEIVE_BOOT_COMPLETED',
 'android.permission.RECORD_AUDIO',
 'android.permission.SYSTEM_ALERT_WINDOW',
 'android.permission.VIBRATE',
 'android.permission.WAKE_LOCK',
 'android.permission.WRITE_CALENDAR',
 'android.permission.WRITE_CONTACTS',
 'android.permission.WRITE_EXTERNAL_STORAGE',
 'com.android.launcher.permission.INSTALL_SHORTCUT',
 'com.android.launcher.permission.UNINSTALL_SHORTCUT',
 'com.facebook.receiver.permission.ACCESS',
 'com.facebook.katana.provider.ACCESS',
 'com.facebook.orca.provider.ACCESS',
 'com.facebook.mlite.provider.ACCESS',
 'com.facebook.wakizashi.provider.ACCESS',
 'com.facebook.permission.prod.FB_APP_COMMUNICATION',
 'com.sec.android.provider.badge.permission.WRITE',
 'com.sec.android.provider.badge.permission.READ',
 'com.htc.launcher.permission.READ_SETTINGS',
 'com.htc.launcher.permission.UPDATE_SHORTCUT',
 'com.sonyericsson.home.permission.BROADCAST_BADGE',
 'com.sonymobile.home.permission.PROVIDER_INSERT_BADGE',
```



```
'com.huawei.android.launcher.permission.CHANGE_BADGE',
'com.huawei.android.launcher.permission.READ_SETTINGS',
'com.huawei.android.launcher.permission.WRITE_SETTINGS',
'com.oppo.launcher.permission.READ_SETTINGS',
'com.oppo.launcher.permission.WRITE_SETTINGS',
'android.permission.REORDER_TASKS',
'android.permission.USE_FULL_SCREEN_INTENT',
'com.facebook.services.identity.FEO2',
'android.permission.FOREGROUND_SERVICE',
'com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE',
'com.google.android.c2dm.permission.RECEIVE']

In [3]:
```

- To get activities listed in the APK file, we use the below mentioned command.

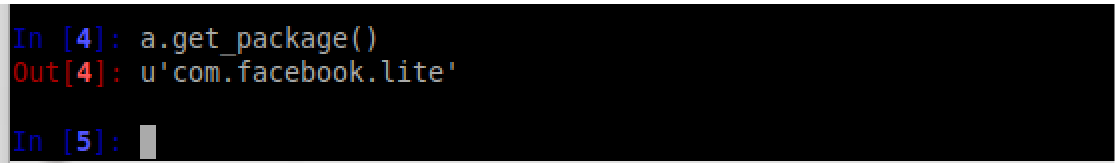
Command: a.get_activities()



```
In [3]: a.get_activities()
Out[3]:
['com.facebook.lite.MainActivity',
 'com.facebook.lite.ShortcutLauncherActivity',
 'com.facebook.lite.ShortcutActivity',
 'com.facebook.lite.rtc.RTCActivity',
 'com.facebook.lite.webviewrtc.RTCIncomingCallActivity',
 'com.facebook.lite.nativeRtc.NativeRtcCallActivity',
 'com.facebook.lite.media.AlbumGalleryActivity',
 'com.facebook.lite.platform.LoginGDPDialogActivityV2',
 'com.facebook.lite.storagemanager.ManageStorageActivity',
 'com.facebook.lite.bugreporter.screencast.ScreencastActivity',
 'com.facebook.lite.inappbrowser.common.BrowserLiteProxyActivity',
 'com.facebook.browser.lite.BrowserLiteActivity',
 'com.facebook.browser.lite.BrowserLiteInMainProcessActivity',
 'com.facebook.lite.deeplinking.UIQRE2EActivity',
 'com.facebook.lite.waotp.WAOtpReceiveCodeActivity',
 'com.google.android.gms.auth.api.signin.internal.SignInHubActivity']
```

- To get the package of the APK.

  Command: a.get_package()



```
In [4]: a.get_package()
Out[4]: u'com.facebook.lite'

In [5]: 
```

- To get the android version code, android version name, min sdk version, max sdk version, target sdk version:

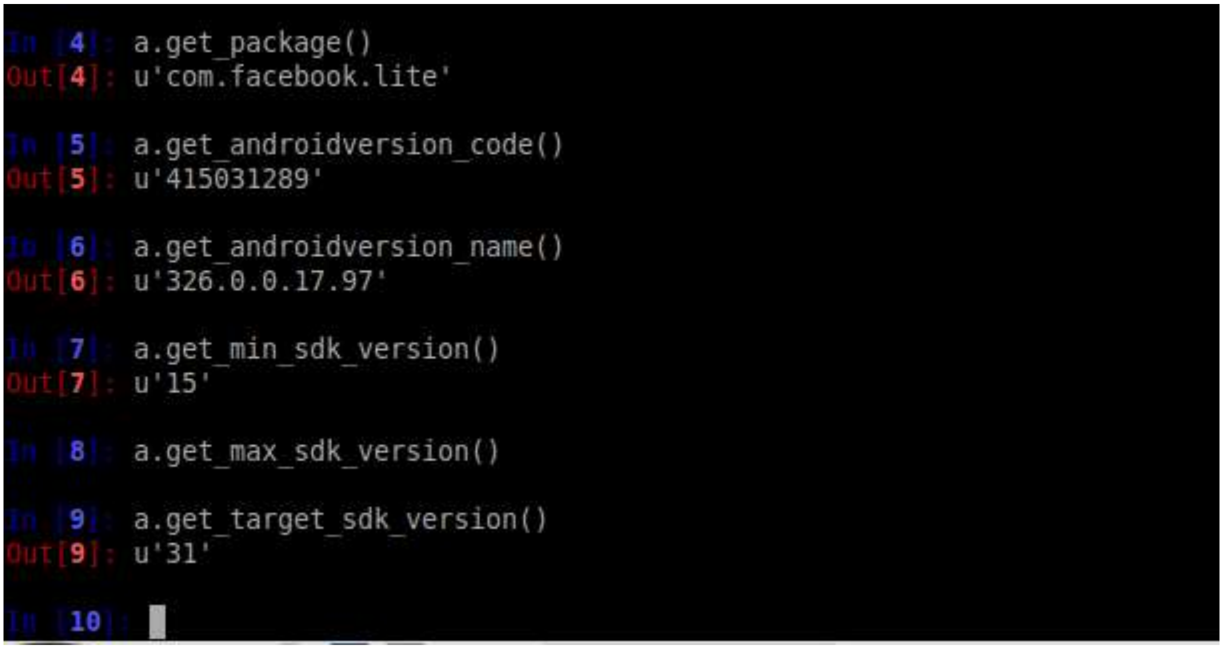
  Command: a.get_androidversion_code()

  Command: a.get_androidversion_name()

  Command: a.get_min_sdk_version()

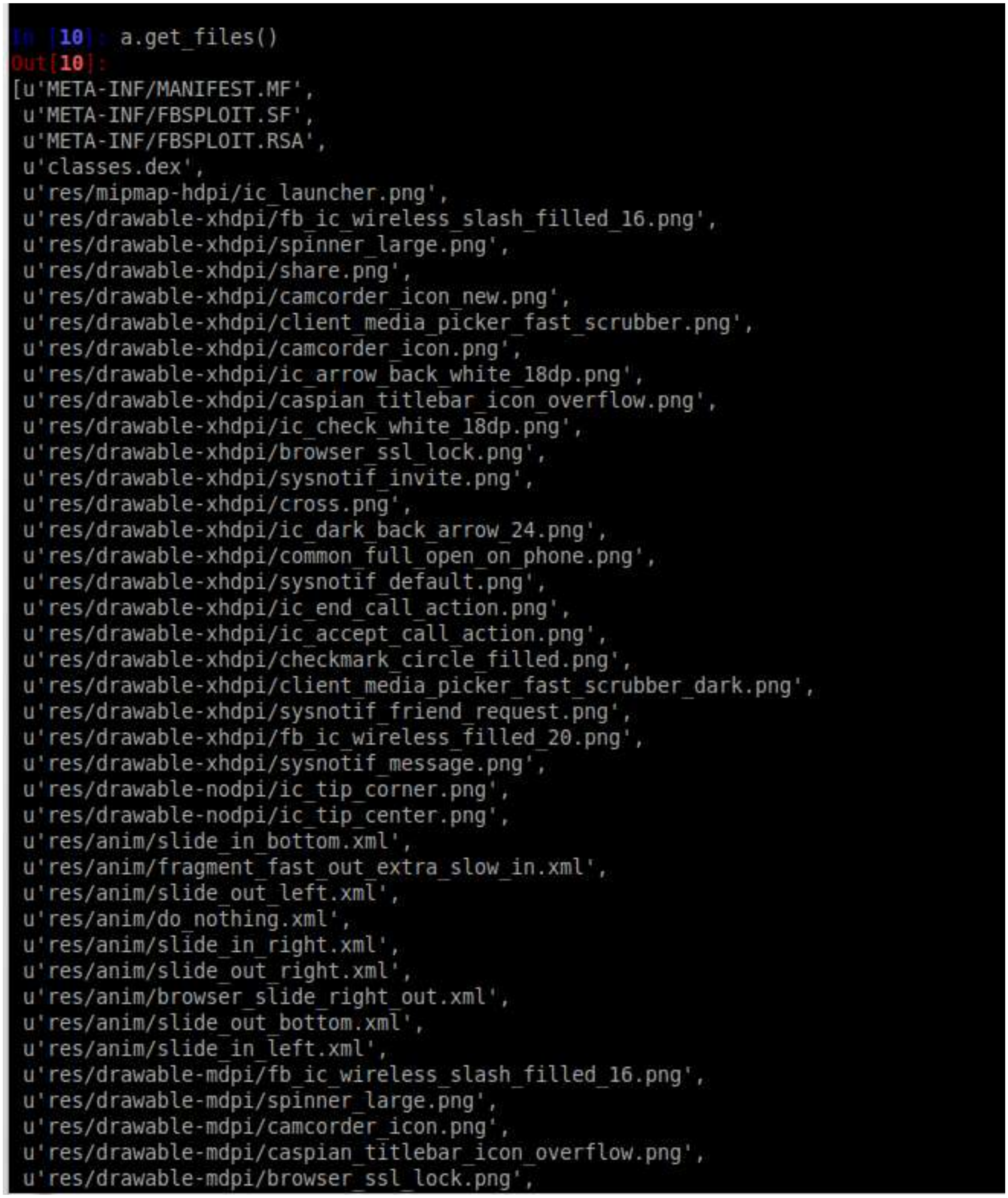
  Command: a.get_max_sdk_version()

  Command: a.get_target_sdk_version()

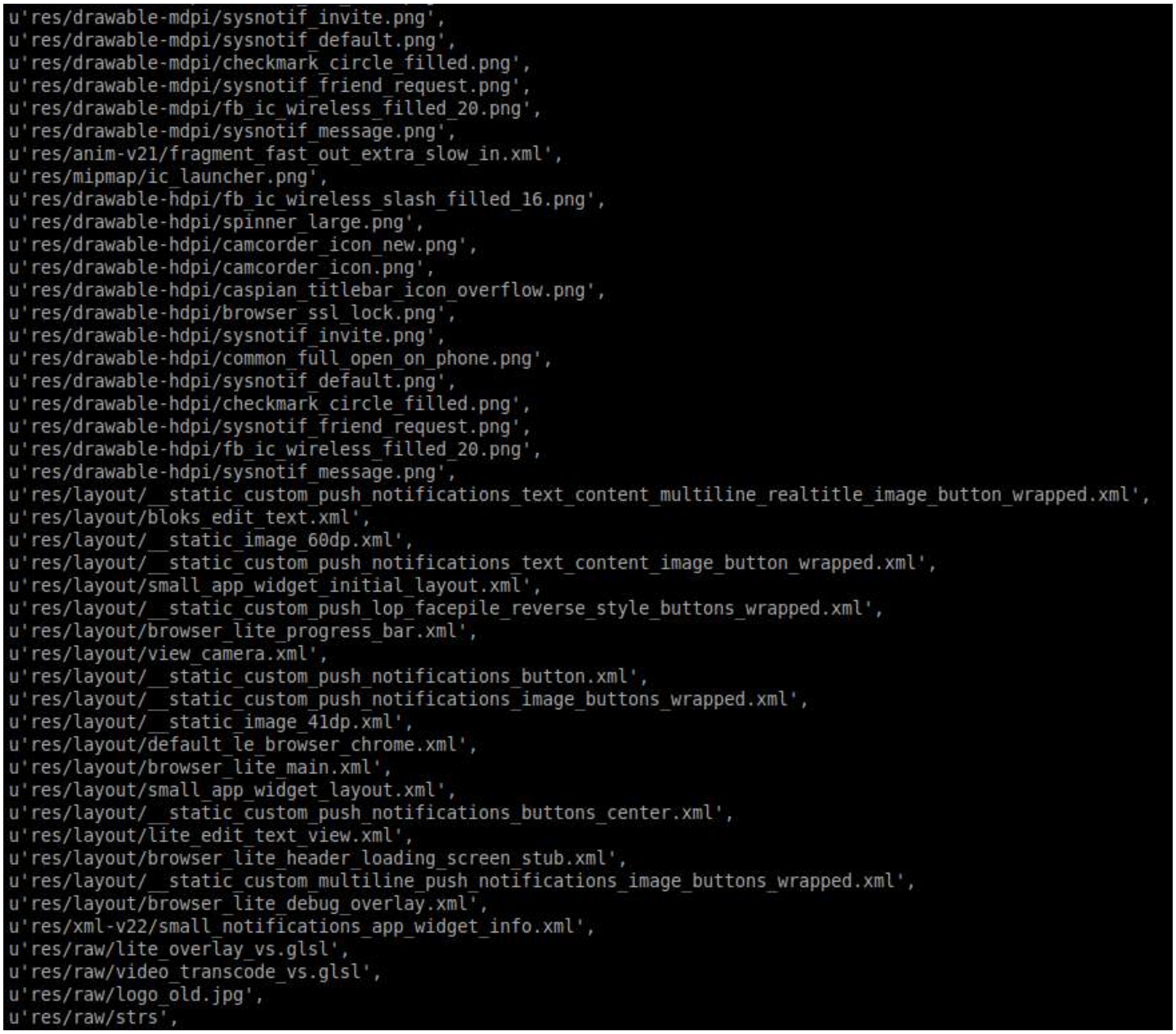


```
In [4]: a.get_package()
Out[4]: u'com.facebook.lite'

In [5]: a.get_androidversion_code()
Out[5]: u'415031289'

In [6]: a.get_androidversion_name()
Out[6]: u'326.0.0.17.97'

In [7]: a.get_min_sdk_version()
Out[7]: u'15'

In [8]: a.get_max_sdk_version()

In [9]: a.get_target_sdk_version()
Out[9]: u'31'

In [10]: 
```

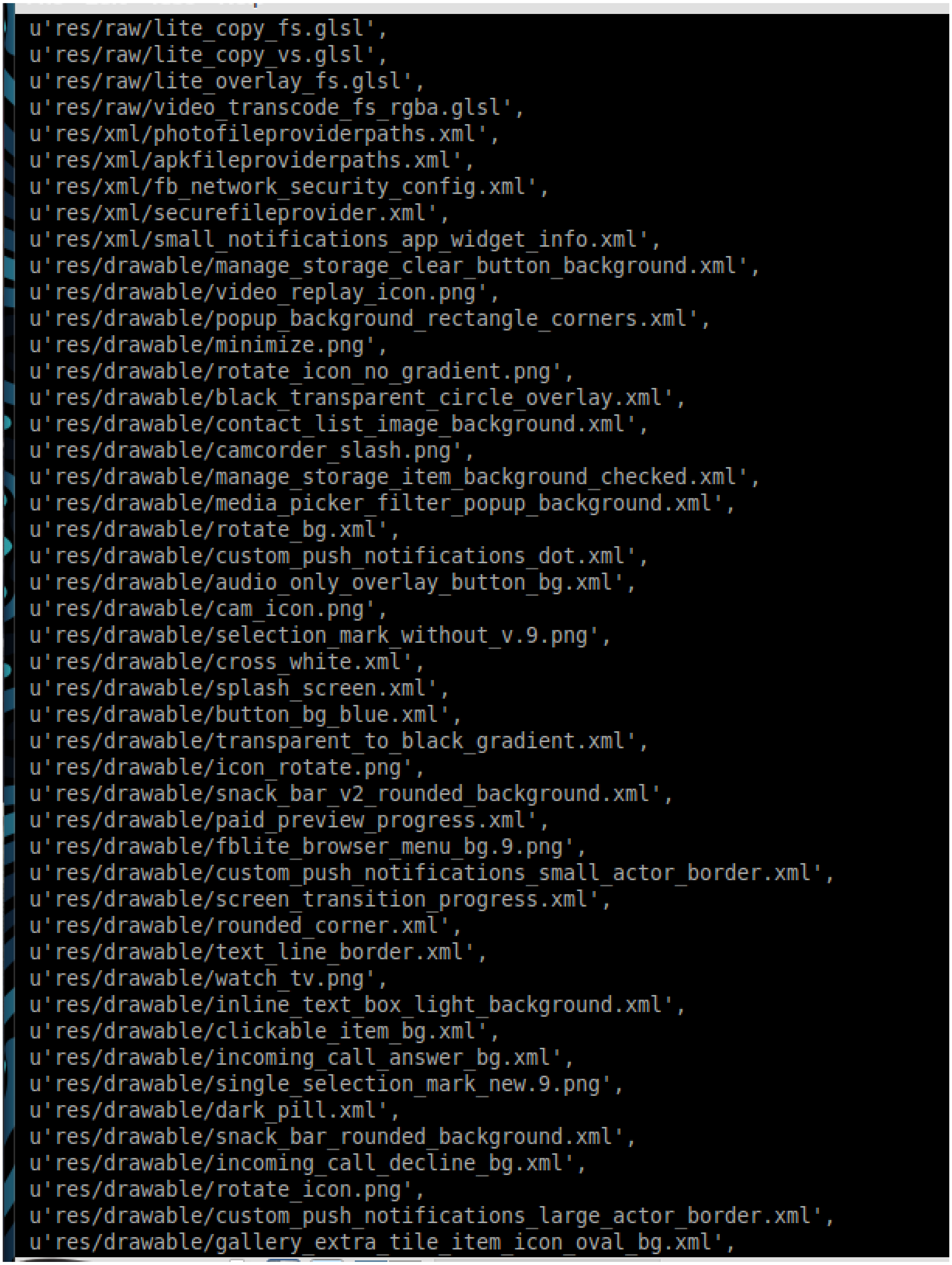
- Extract the files contained in the app:
  Command: a.get_files()



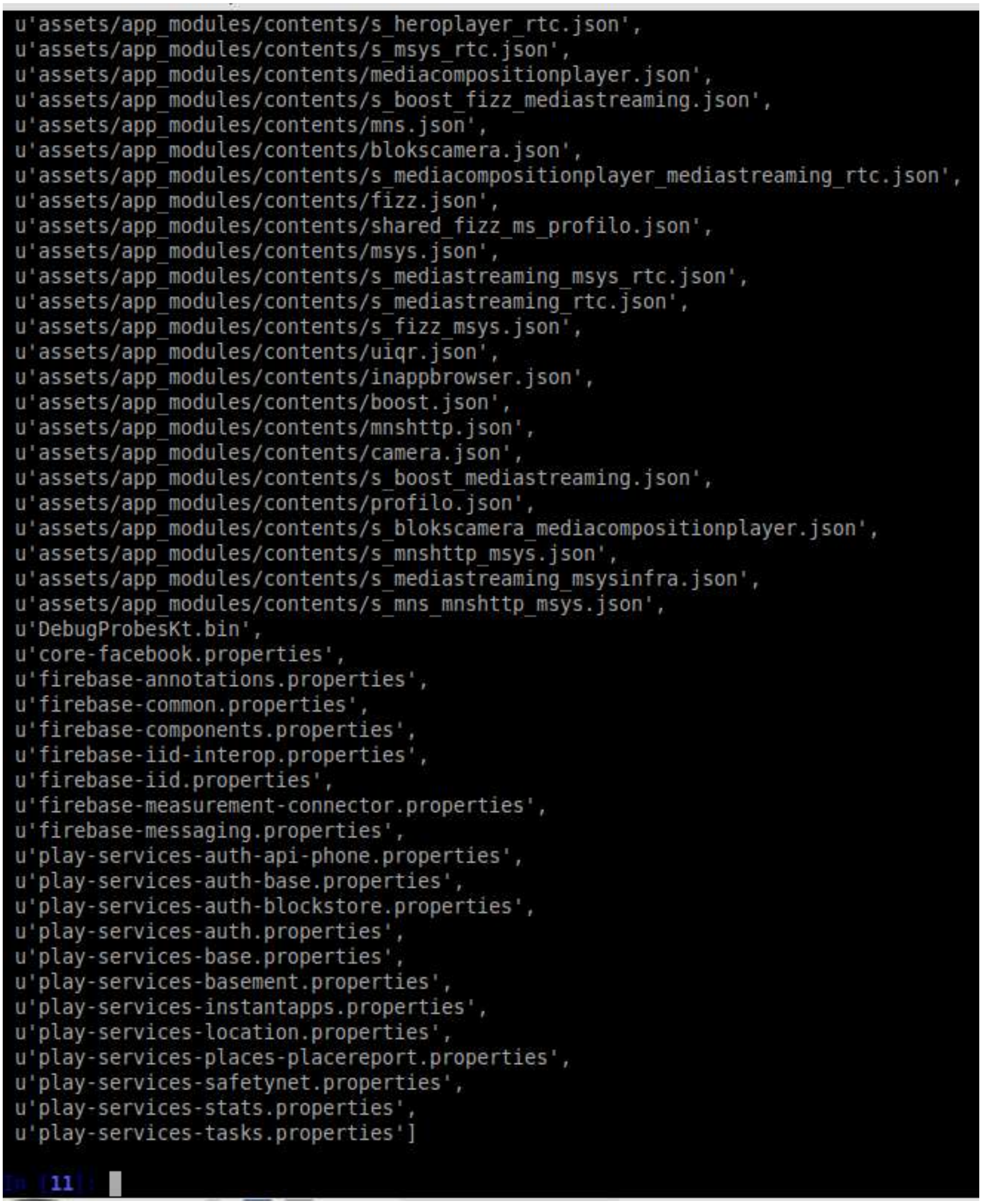
```
In [10]: a.get_files()
Out[10]:
[u'META-INF/MANIFEST.MF',
 u'META-INF/FBSPLOIT.SF',
 u'META-INF/FBSPLOIT.RSA',
 u'classes.dex',
 u'res/mipmap-hdpi/ic_launcher.png',
 u'res/drawable-xhdpi/fb_ic_wireless_slash_filled_16.png',
 u'res/drawable-xhdpi/spinner_large.png',
 u'res/drawable-xhdpi/share.png',
 u'res/drawable-xhdpi/camcorder_icon_new.png',
 u'res/drawable-xhdpi/client_media_picker_fast_scrubber.png',
 u'res/drawable-xhdpi/camcorder_icon.png',
 u'res/drawable-xhdpi/ic_arrow_back_white_18dp.png',
 u'res/drawable-xhdpi/caspian_titlebar_icon_overflow.png',
 u'res/drawable-xhdpi/ic_check_white_18dp.png',
 u'res/drawable-xhdpi/browser_ssl_lock.png',
 u'res/drawable-xhdpi/sysnotif_invite.png',
 u'res/drawable-xhdpi/cross.png',
 u'res/drawable-xhdpi/ic_dark_back_arrow_24.png',
 u'res/drawable-xhdpi/common_full_open_on_phone.png',
 u'res/drawable-xhdpi/sysnotif_default.png',
 u'res/drawable-xhdpi/ic_end_call_action.png',
 u'res/drawable-xhdpi/ic_accept_call_action.png',
 u'res/drawable-xhdpi/checkmark_circle_filled.png',
 u'res/drawable-xhdpi/client_media_picker_fast_scrubber_dark.png',
 u'res/drawable-xhdpi/sysnotif_friend_request.png',
 u'res/drawable-xhdpi/fb_ic_wireless_filled_20.png',
 u'res/drawable-xhdpi/sysnotif_message.png',
 u'res/drawable-nodpi/ic_tip_corner.png',
 u'res/drawable-nodpi/ic_tip_center.png',
 u'res/anim/slide_in_bottom.xml',
 u'res/anim/fragment_fast_out_extra_slow_in.xml',
 u'res/anim/slide_out_left.xml',
 u'res/anim/do_nothing.xml',
 u'res/anim/slide_in_right.xml',
 u'res/anim/slide_out_right.xml',
 u'res/anim/browser_slide_right_out.xml',
 u'res/anim/slide_out_bottom.xml',
 u'res/anim/slide_in_left.xml',
 u'res/drawable-mdpi/fb_ic_wireless_slash_filled_16.png',
 u'res/drawable-mdpi/spinner_large.png',
 u'res/drawable-mdpi/camcorder_icon.png',
 u'res/drawable-mdpi/caspian_titlebar_icon_overflow.png',
 u'res/drawable-mdpi/browser_ssl_lock.png',
```



```
u'res/drawable-mdpi/sysnotif_invite.png',
u'res/drawable-mdpi/sysnotif_default.png',
u'res/drawable-mdpi/checkmark_circle_filled.png',
u'res/drawable-mdpi/sysnotif_friend_request.png',
u'res/drawable-mdpi/fb_ic_wireless_filled_20.png',
u'res/drawable-mdpi/sysnotif_message.png',
u'res/anim-v21/fragment_fast_out_extra_slow_in.xml',
u'res/mipmap/ic_launcher.png',
u'res/drawable-hdpi/fb_ic_wireless_slash_filled_16.png',
u'res/drawable-hdpi/spinner_large.png',
u'res/drawable-hdpi/camcorder_icon_new.png',
u'res/drawable-hdpi/camcorder_icon.png',
u'res/drawable-hdpi/caspian_titlebar_icon_overflow.png',
u'res/drawable-hdpi/browser_ssl_lock.png',
u'res/drawable-hdpi/sysnotif_invite.png',
u'res/drawable-hdpi/common_full_open_on_phone.png',
u'res/drawable-hdpi/sysnotif_default.png',
u'res/drawable-hdpi/checkmark_circle_filled.png',
u'res/drawable-hdpi/sysnotif_friend_request.png',
u'res/drawable-hdpi/fb_ic_wireless_filled_20.png',
u'res/drawable-hdpi/sysnotif_message.png',
u'res/layout/__static_custom_push_notifications_text_content_multiline_realtitle_image_button_wrapped.xml',
u'res/layout/bloks_edit_text.xml',
u'res/layout/__static_image_60dp.xml',
u'res/layout/__static_custom_push_notifications_text_content_image_button_wrapped.xml',
u'res/layout/small_app_widget_initial_layout.xml',
u'res/layout/__static_custom_push_lop_facepile_reverse_style_buttons_wrapped.xml',
u'res/layout/browser_lite_progress_bar.xml',
u'res/layout/view_camera.xml',
u'res/layout/__static_custom_push_notifications_button.xml',
u'res/layout/__static_custom_push_notifications_image_buttons_wrapped.xml',
u'res/layout/__static_image_41dp.xml',
u'res/layout/default_le_browser_chrome.xml',
u'res/layout/browser_lite_main.xml',
u'res/layout/small_app_widget_layout.xml',
u'res/layout/__static_custom_push_notifications_buttons_center.xml',
u'res/layout/lite_edit_text_view.xml',
u'res/layout/browser_lite_header_loading_screen_stub.xml',
u'res/layout/__static_custom_multiline_push_notifications_image_buttons_wrapped.xml',
u'res/layout/browser_lite_debug_overlay.xml',
u'res/xml-v22/small_notifications_app_widget_info.xml',
u'res/raw/lite_overlay_vs.glsl',
u'res/raw/video_transcode_vs.glsl',
u'res/raw/logo_old.jpg',
u'res/raw/strs',
```



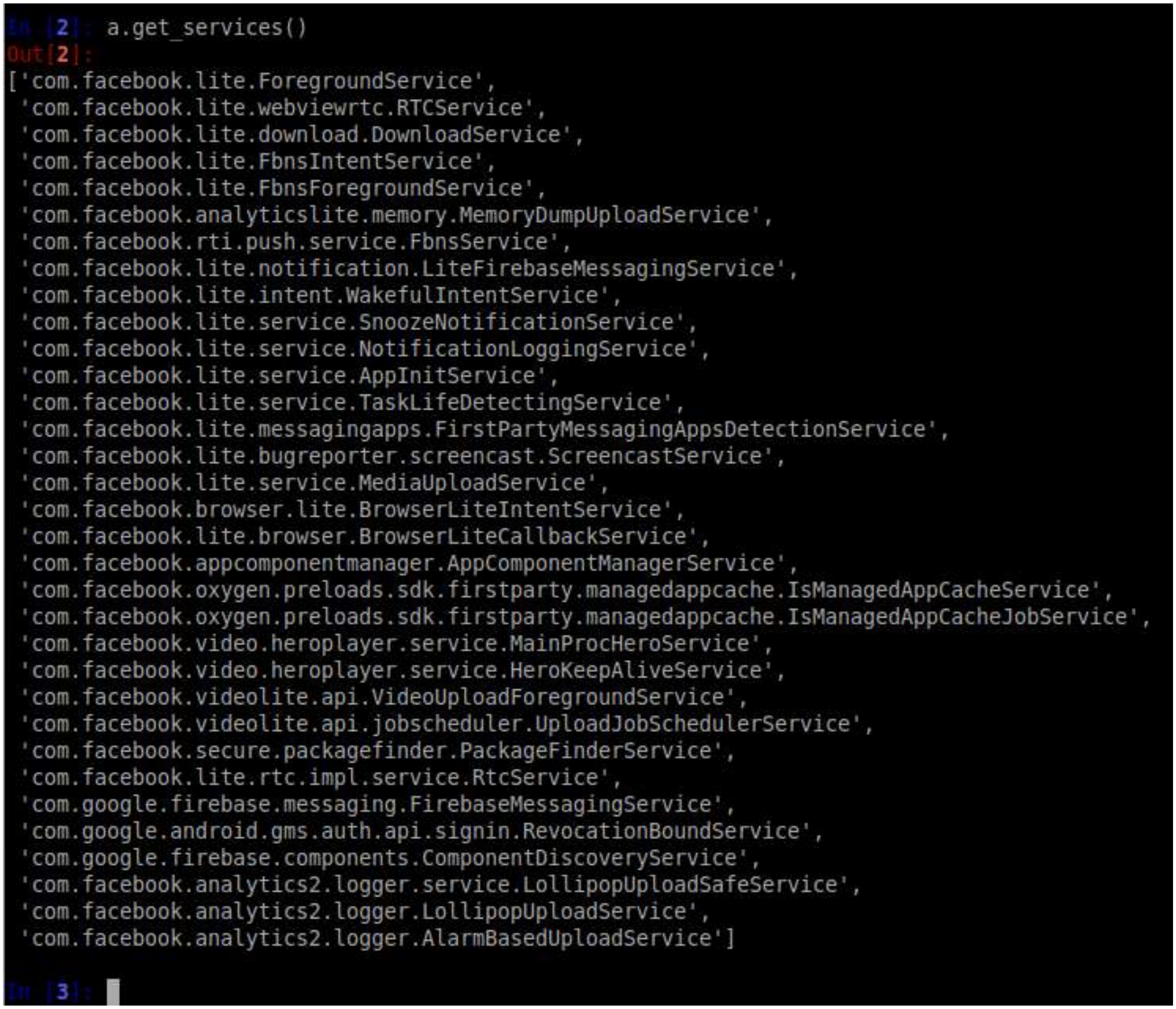
```
u'res/raw/lite_copy_fs.glsl',
u'res/raw/lite_copy_vs.glsl',
u'res/raw/lite_overlay_fs.glsl',
u'res/raw/video_transcode_fs_rgba.glsl',
u'res/xml/photofileproviderpaths.xml',
u'res/xml/apkfileproviderpaths.xml',
u'res/xml/fb_network_security_config.xml',
u'res/xml/securefileprovider.xml',
u'res/xml/small_notifications_app_widget_info.xml',
u'res/drawable/manage_storage_clear_button_background.xml',
u'res/drawable/video_replay_icon.png',
u'res/drawable/popup_background_rectangle_corners.xml',
u'res/drawable/minimize.png',
u'res/drawable/rotate_icon_no_gradient.png',
u'res/drawable/black_transparent_circle_overlay.xml',
u'res/drawable/contact_list_image_background.xml',
u'res/drawable/camcorder_slash.png',
u'res/drawable/manage_storage_item_background_checked.xml',
u'res/drawable/media_picker_filter_popup_background.xml',
u'res/drawable/rotate_bg.xml',
u'res/drawable/custom_push_notifications_dot.xml',
u'res/drawable/audio_only_overlay_button_bg.xml',
u'res/drawable/cam_icon.png',
u'res/drawable/selection_mark_without_v.9.png',
u'res/drawable/cross_white.xml',
u'res/drawable/splash_screen.xml',
u'res/drawable/button_bg_blue.xml',
u'res/drawable/transparent_to_black_gradient.xml',
u'res/drawable/icon_rotate.png',
u'res/drawable/snack_bar_v2_rounded_background.xml',
u'res/drawable/paid_preview_progress.xml',
u'res/drawable/fblite_browser_menu_bg.9.png',
u'res/drawable/custom_push_notifications_small_actor_border.xml',
u'res/drawable/screen_transition_progress.xml',
u'res/drawable/rounded_corner.xml',
u'res/drawable/text_line_border.xml',
u'res/drawable/watch_tv.png',
u'res/drawable/inline_text_box_light_background.xml',
u'res/drawable/clickable_item_bg.xml',
u'res/drawable/incoming_call_answer_bg.xml',
u'res/drawable/single_selection_mark_new.9.png',
u'res/drawable/dark_pill.xml',
u'res/drawable/snack_bar_rounded_background.xml',
u'res/drawable/incoming_call_decline_bg.xml',
u'res/drawable/rotate_icon.png',
u'res/drawable/custom_push_notifications_large_actor_border.xml',
u'res/drawable/gallery_extra_tile_item_icon_oval_bg.xml',
```



```
u'assets/app_modules/contents/s_heroplayer_rtc.json',
u'assets/app_modules/contents/s_msys_rtc.json',
u'assets/app_modules/contents/mediacompositionplayer.json',
u'assets/app_modules/contents/s_boost_fizz_mediastreaming.json',
u'assets/app_modules/contents/mns.json',
u'assets/app_modules/contents/blokscamera.json',
u'assets/app_modules/contents/s_mediacompositionplayer_mediastreaming_rtc.json',
u'assets/app_modules/contents/fizz.json',
u'assets/app_modules/contents/shared_fizz_ms_profilo.json',
u'assets/app_modules/contents/msys.json',
u'assets/app_modules/contents/s_mediastreaming_msys_rtc.json',
u'assets/app_modules/contents/s_mediastreaming_rtc.json',
u'assets/app_modules/contents/s_fizz_msys.json',
u'assets/app_modules/contents/uiqr.json',
u'assets/app_modules/contents/inappbrowser.json',
u'assets/app_modules/contents/boost.json',
u'assets/app_modules/contents/mnshttp.json',
u'assets/app_modules/contents/camera.json',
u'assets/app_modules/contents/s_boost_mediastreaming.json',
u'assets/app_modules/contents/profilo.json',
u'assets/app_modules/contents/s_blokscamera_mediacompositionplayer.json',
u'assets/app_modules/contents/s_mnshttp_msys.json',
u'assets/app_modules/contents/s_mediastreaming_msysinfra.json',
u'assets/app_modules/contents/s_mns_mnshttp_msys.json',
u'DebugProbesKt.bin',
u'core-facebook.properties',
u'firebase-annotations.properties',
u'firebase-common.properties',
u'firebase-components.properties',
u'firebase-iid-interop.properties',
u'firebase-iid.properties',
u'firebase-measurement-connector.properties',
u'firebase-messaging.properties',
u'play-services-auth-api-phone.properties',
u'play-services-auth-base.properties',
u'play-services-auth-blockstore.properties',
u'play-services-auth.properties',
u'play-services-base.properties',
u'play-services-basement.properties',
u'play-services-instantapps.properties',
u'play-services-location.properties',
u'play-services-places-placereport.properties',
u'play-services-safetynet.properties',
u'play-services-stats.properties',
u'play-services-tasks.properties']

In [11]:
```

- A service is a general entry point for keeping an app running in the background. Services used by the app can be known by using the following command:

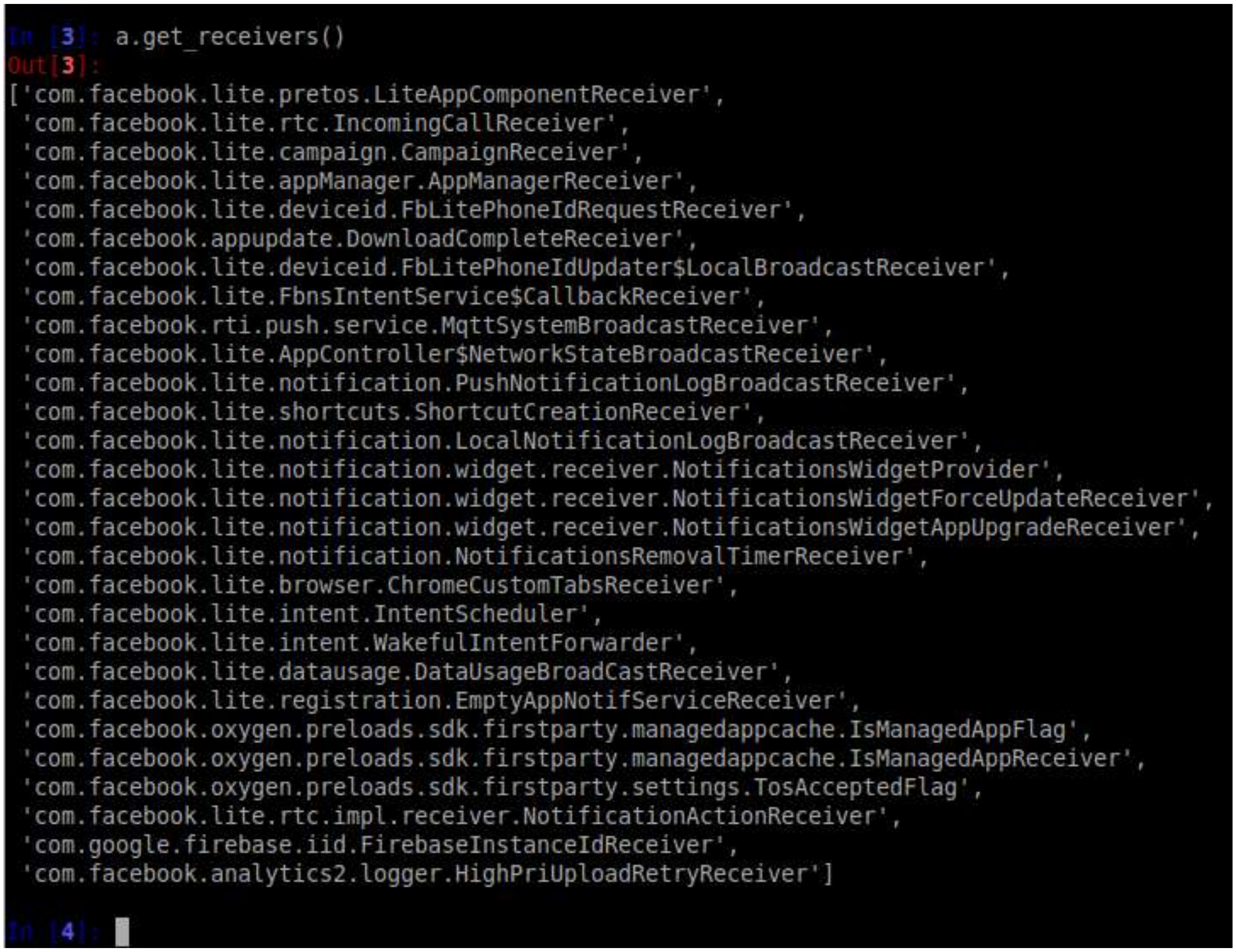
  a.get_services()



```
In [2]: a.get_services()
Out[2]:
['com.facebook.lite.ForegroundService',
 'com.facebook.lite.webviewrtc.RTCService',
 'com.facebook.lite.download.DownloadService',
 'com.facebook.lite.FbnsIntentService',
 'com.facebook.lite.FbnsForegroundService',
 'com.facebook.analyticslite.memory.MemoryDumpUploadService',
 'com.facebook.rti.push.service.FbnsService',
 'com.facebook.lite.notification.LiteFirebaseMessagingService',
 'com.facebook.lite.intent.WakefulIntentService',
 'com.facebook.lite.service.SnoozeNotificationService',
 'com.facebook.lite.service.NotificationLoggingService',
 'com.facebook.lite.service.AppInitService',
 'com.facebook.lite.service.TaskLifeDetectingService',
 'com.facebook.lite.messagingapps.FirstPartyMessagingAppsDetectionService',
 'com.facebook.lite.bugreporter.screencast.ScreencastService',
 'com.facebook.lite.service.MediaUploadService',
 'com.facebook.browser.lite.BrowserLiteIntentService',
 'com.facebook.lite.browser.BrowserLiteCallbackService',
 'com.facebook.appcomponentmanager.AppComponentManagerService',
 'com.facebook.oxygen.preloads.sdk.firstparty.managedappcache.IsManagedAppCacheService',
 'com.facebook.oxygen.preloads.sdk.firstparty.managedappcache.IsManagedAppCacheJobService',
 'com.facebook.video.heroplayer.service.MainProcHeroService',
 'com.facebook.video.heroplayer.service.HeroKeepAliveService',
 'com.facebook.videolite.api.VideoUploadForegroundService',
 'com.facebook.videolite.api.jobscheduler.UploadJobSchedulerService',
 'com.facebook.secure.packagefinder.PackageFinderService',
 'com.facebook.lite.rtc.impl.service.RtcService',
 'com.google.firebase.messaging.FirebaseMessagingService',
 'com.google.android.gms.auth.api.signin.RevocationBoundService',
 'com.google.firebase.components.ComponentDiscoveryService',
 'com.facebook.analytics2.logger.service.LollipopUploadSafeService',
 'com.facebook.analytics2.logger.LollipopUploadService',
 'com.facebook.analytics2.logger.AlarmBasedUploadService']

In [3]:
```

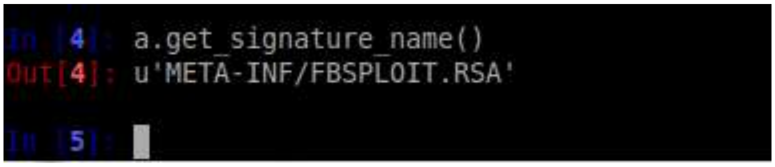
- A receiver is a component that enables the system to deliver events to the app outside of a regular user flow, allowing the app to respond to system-wide broadcast announcements. This can be obtained using the below command.

$ a.get_receivers()



```
In [3]: a.get_receivers()
Out[3]:
['com.facebook.lite.pretos.LiteAppComponentReceiver',
 'com.facebook.lite.rtc.IncomingCallReceiver',
 'com.facebook.lite.campaign.CampaignReceiver',
 'com.facebook.lite.appManager.AppManagerReceiver',
 'com.facebook.lite.deviceid.FbLitePhoneIdRequestReceiver',
 'com.facebook.appupdate.DownloadCompleteReceiver',
 'com.facebook.lite.deviceid.FbLitePhoneIdUpdater$LocalBroadcastReceiver',
 'com.facebook.lite.FbnsIntentService$CallbackReceiver',
 'com.facebook.rti.push.service.MqttSystemBroadcastReceiver',
 'com.facebook.lite.AppController$NetworkStateBroadcastReceiver',
 'com.facebook.lite.notification.PushNotificationLogBroadcastReceiver',
 'com.facebook.lite.shortcuts.ShortcutCreationReceiver',
 'com.facebook.lite.notification.LocalNotificationLogBroadcastReceiver',
 'com.facebook.lite.notification.widget.receiver.NotificationsWidgetProvider',
 'com.facebook.lite.notification.widget.receiver.NotificationsWidgetForceUpdateReceiver',
 'com.facebook.lite.notification.widget.receiver.NotificationsWidgetAppUpgradeReceiver',
 'com.facebook.lite.notification.NotificationsRemovalTimerReceiver',
 'com.facebook.lite.browser.ChromeCustomTabsReceiver',
 'com.facebook.lite.intent.IntentScheduler',
 'com.facebook.lite.intent.WakefulIntentForwarder',
 'com.facebook.lite.datausage.DataUsageBroadCastReceiver',
 'com.facebook.lite.registration.EmptyAppNotifServiceReceiver',
 'com.facebook.oxygen.preloads.sdk.firstparty.managedappcache.IsManagedAppFlag',
 'com.facebook.oxygen.preloads.sdk.firstparty.managedappcache.IsManagedAppReceiver',
 'com.facebook.oxygen.preloads.sdk.firstparty.settings.TosAcceptedFlag',
 'com.facebook.lite.rtc.impl.receiver.NotificationActionReceiver',
 'com.google.firebase.iid.FirebaseInstanceIdReceiver',
 'com.facebook.analytics2.logger.HighPriUploadRetryReceiver']

In [4]:
```

- We can check where the app signature is located. This displays that the Signature/KEY file is located in the META-INF folder of the APK

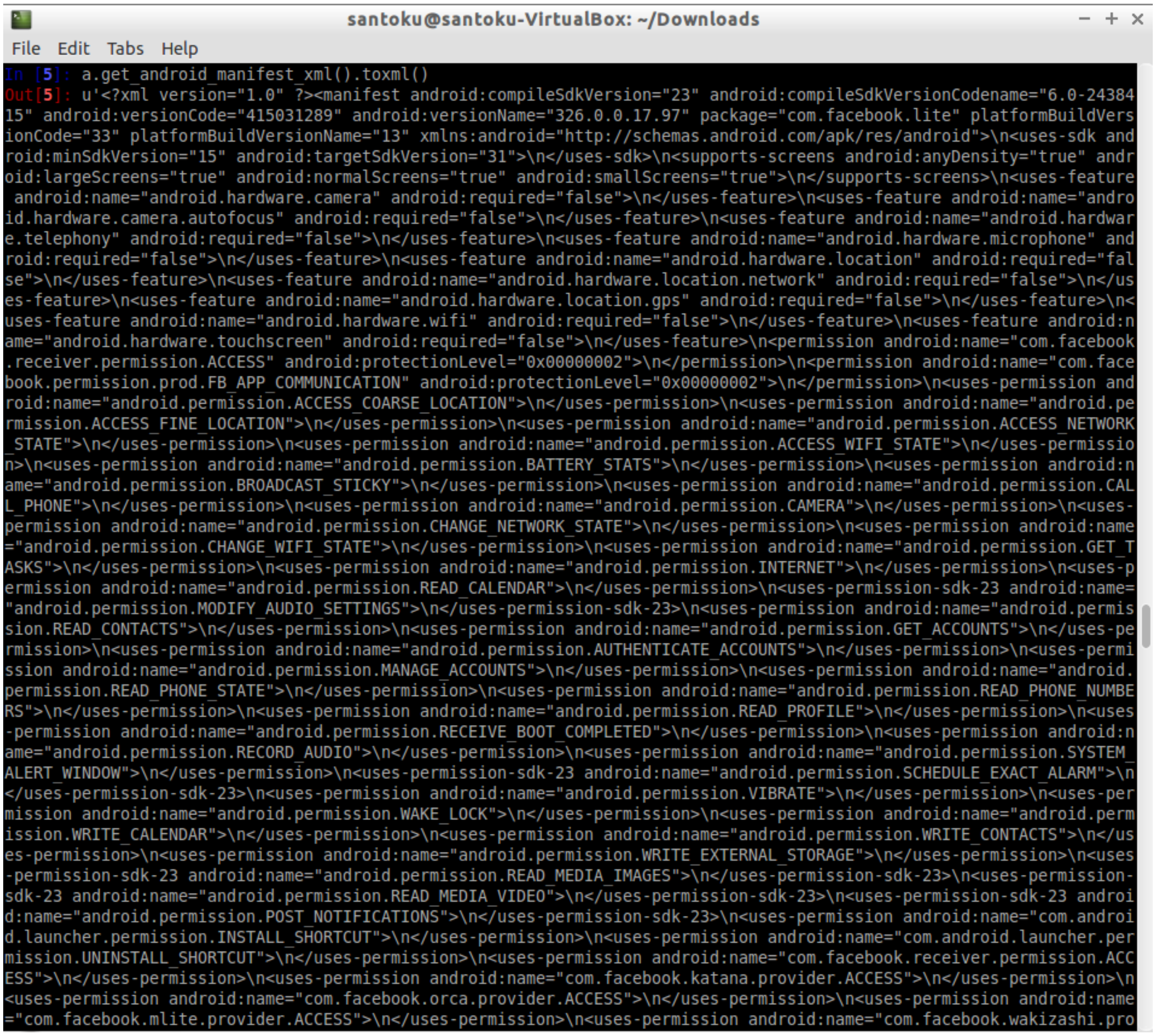


```
In [4]: a.get_signature_name()
Out[4]: u'META-INF/FBSPLOIT.RSA'

In [5]:
```

- Get the AndroidManifest.xml:

Use the following command:

```
a.get_android_manifest_xml().toxml()
```

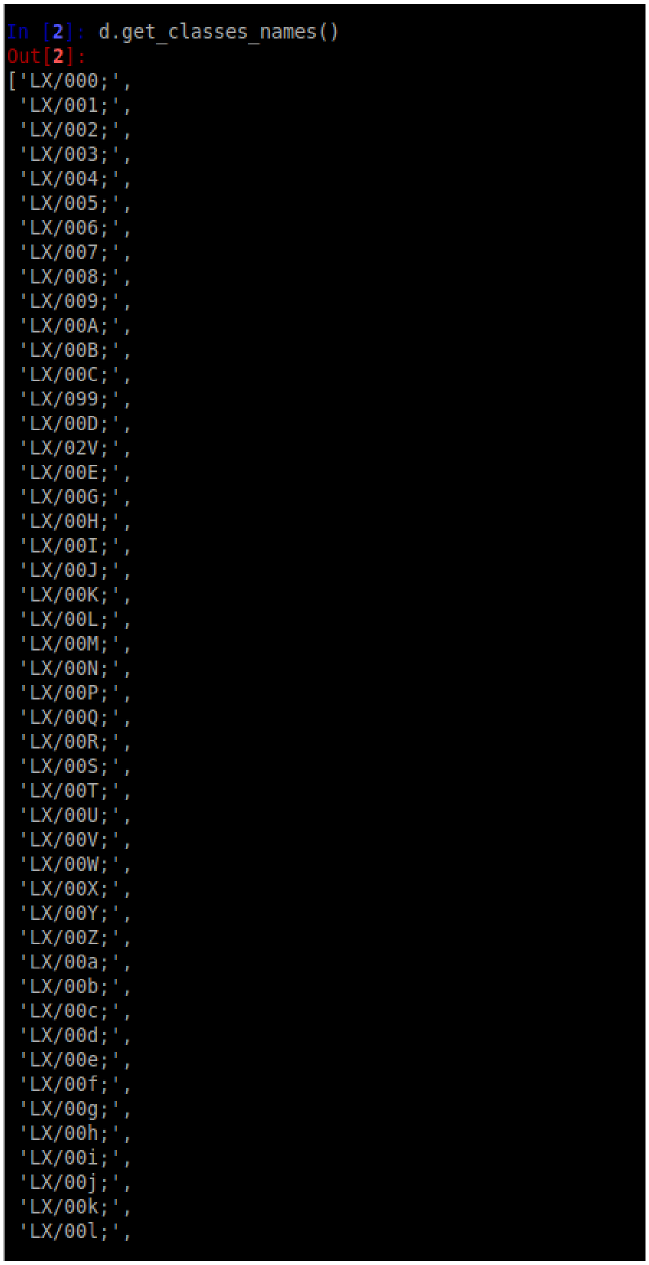


```
In [5]: a.get_android_manifest_xml().toxml()
Out[5]: u'<?xml version="1.0" ?><manifest android:compileSdkVersion="23" android:compileSdkVersionCodename="6.0-24384
15" android:versionCode="415031289" android:versionName="326.0.0.17.97" package="com.facebook.lite" platformBuildVers
ionCode="33" platformBuildVersionName="13" xmlns:android="http://schemas.android.com/apk/res/android">\n<uses-sdk and
roid:minSdkVersion="15" android:targetSdkVersion="31">\n</uses-sdk>\n<supports-screens android:anyDensity="true" andr
oid:largeScreens="true" android:normalScreens="true" android:smallScreens="true">\n</supports-screens>\n<uses-feature
 android:name="android.hardware.camera" android:required="false">\n</uses-feature>\n<uses-feature android:name="andro
id.hardware.camera.autofocus" android:required="false">\n</uses-feature>\n<uses-feature android:name="android.hardwar
e.telephony" android:required="false">\n</uses-feature>\n<uses-feature android:name="android.hardware.microphone" and
roid:required="false">\n</uses-feature>\n<uses-feature android:name="android.hardware.location" android:required="fal
se">\n</uses-feature>\n<uses-feature android:name="android.hardware.location.network" android:required="false">\n</us
es-feature>\n<uses-feature android:name="android.hardware.location.gps" android:required="false">\n</uses-feature>\n<
uses-feature android:name="android.hardware.wifi" android:required="false">\n</uses-feature>\n<uses-feature android:n
ame="android.hardware.touchscreen" android:required="false">\n</uses-feature>\n<permission android:name="com.facebook
.receiver.permission.ACCESS" android:protectionLevel="0x00000002">\n</permission>\n<permission android:name="com.face
book.permission.prod.FB_APP_COMMUNICATION" android:protectionLevel="0x00000002">\n</permission>\n<uses-permission and
roid:name="android.permission.ACCESS_COARSE_LOCATION">\n</uses-permission>\n<uses-permission android:name="android.pe
rmission.ACCESS_FINE_LOCATION">\n</uses-permission>\n<uses-permission android:name="android.permission.ACCESS_NETWORK
_STATE">\n</uses-permission>\n<uses-permission android:name="android.permission.ACCESS_WIFI_STATE">\n</uses-permissio
n>\n<uses-permission android:name="android.permission.BATTERY_STATS">\n</uses-permission>\n<uses-permission android:n
ame="android.permission.BROADCAST_STICKY">\n</uses-permission>\n<uses-permission android:name="android.permission.CAL
L_PHONE">\n</uses-permission>\n<uses-permission android:name="android.permission.CAMERA">\n</uses-permission>\n<uses-
permission android:name="android.permission.CHANGE_NETWORK_STATE">\n</uses-permission>\n<uses-permission android:name
="android.permission.CHANGE_WIFI_STATE">\n</uses-permission>\n<uses-permission android:name="android.permission.GET_T
ASKS">\n</uses-permission>\n<uses-permission android:name="android.permission.INTERNET">\n</uses-permission>\n<uses-p
ermission android:name="android.permission.READ_CALENDAR">\n</uses-permission>\n<uses-permission-sdk-23 android:name=
"android.permission.MODIFY_AUDIO_SETTINGS">\n</uses-permission-sdk-23>\n<uses-permission android:name="android.permis
sion.READ_CONTACTS">\n</uses-permission>\n<uses-permission android:name="android.permission.GET_ACCOUNTS">\n</uses-pe
rmission>\n<uses-permission android:name="android.permission.AUTHENTICATE_ACCOUNTS">\n</uses-permission>\n<uses-permi
ssion android:name="android.permission.MANAGE_ACCOUNTS">\n</uses-permission>\n<uses-permission android:name="android.
permission.READ_PHONE_STATE">\n</uses-permission>\n<uses-permission android:name="android.permission.READ_PHONE_NUMBE
RS">\n</uses-permission>\n<uses-permission android:name="android.permission.READ_PROFILE">\n</uses-permission>\n<uses
-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED">\n</uses-permission>\n<uses-permission android:n
ame="android.permission.RECORD_AUDIO">\n</uses-permission>\n<uses-permission android:name="android.permission.SYSTEM_
ALERT_WINDOW">\n</uses-permission>\n<uses-permission-sdk-23 android:name="android.permission.SCHEDULE_EXACT_ALARM">\n
</uses-permission-sdk-23>\n<uses-permission android:name="android.permission.VIBRATE">\n</uses-permission>\n<uses-per
mission android:name="android.permission.WAKE_LOCK">\n</uses-permission>\n<uses-permission android:name="android.perm
ission.WRITE_CALENDAR">\n</uses-permission>\n<uses-permission android:name="android.permission.WRITE_CONTACTS">\n</us
es-permission>\n<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE">\n</uses-permission>\n<uses
-permission-sdk-23 android:name="android.permission.READ_MEDIA_IMAGES">\n</uses-permission-sdk-23>\n<uses-permission-
sdk-23 android:name="android.permission.READ_MEDIA_VIDEO">\n</uses-permission-sdk-23>\n<uses-permission-sdk-23 androi
d:name="android.permission.POST_NOTIFICATIONS">\n</uses-permission-sdk-23>\n<uses-permission android:name="com.androi
d.launcher.permission.INSTALL_SHORTCUT">\n</uses-permission>\n<uses-permission android:name="com.android.launcher.per
mission.UNINSTALL_SHORTCUT">\n</uses-permission>\n<uses-permission android:name="com.facebook.receiver.permission.ACC
ESS">\n</uses-permission>\n<uses-permission android:name="com.facebook.katana.provider.ACCESS">\n</uses-permission>\n
<uses-permission android:name="com.facebook.orca.provider.ACCESS">\n</uses-permission>\n<uses-permission android:name
="com.facebook.mlite.provider.ACCESS">\n</uses-permission>\n<uses-permission android:name="com.facebook.wakizashi.pro
```

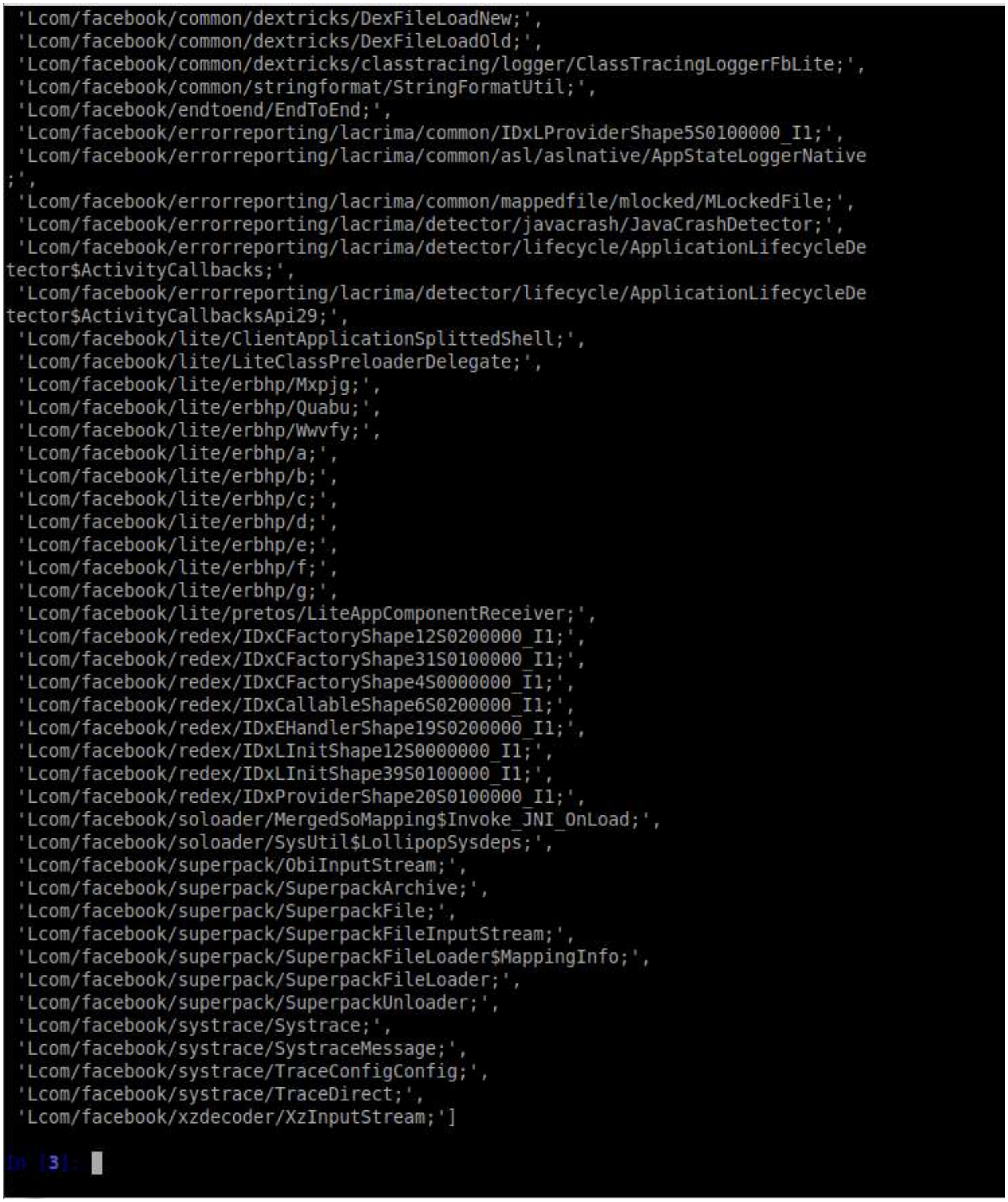
- We can also get the names of the classes we in the app.

To get the classes, run the below command:

d.get_classes_names()



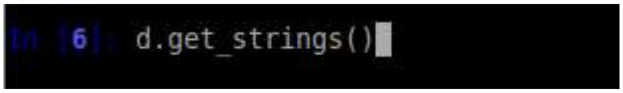
```
In [2]: d.get_classes_names()
Out[2]:
['LX/000;',
 'LX/001;',
 'LX/002;',
 'LX/003;',
 'LX/004;',
 'LX/005;',
 'LX/006;',
 'LX/007;',
 'LX/008;',
 'LX/009;',
 'LX/00A;',
 'LX/00B;',
 'LX/00C;',
 'LX/099;',
 'LX/00D;',
 'LX/02V;',
 'LX/00E;',
 'LX/00G;',
 'LX/00H;',
 'LX/00I;',
 'LX/00J;',
 'LX/00K;',
 'LX/00L;',
 'LX/00M;',
 'LX/00N;',
 'LX/00P;',
 'LX/00Q;',
 'LX/00R;',
 'LX/00S;',
 'LX/00T;',
 'LX/00U;',
 'LX/00V;',
 'LX/00W;',
 'LX/00X;',
 'LX/00Y;',
 'LX/00Z;',
 'LX/00a;',
 'LX/00b;',
 'LX/00c;',
 'LX/00d;',
 'LX/00e;',
 'LX/00f;',
 'LX/00g;',
 'LX/00h;',
 'LX/00i;',
 'LX/00j;',
 'LX/00k;',
 'LX/00l;',
```



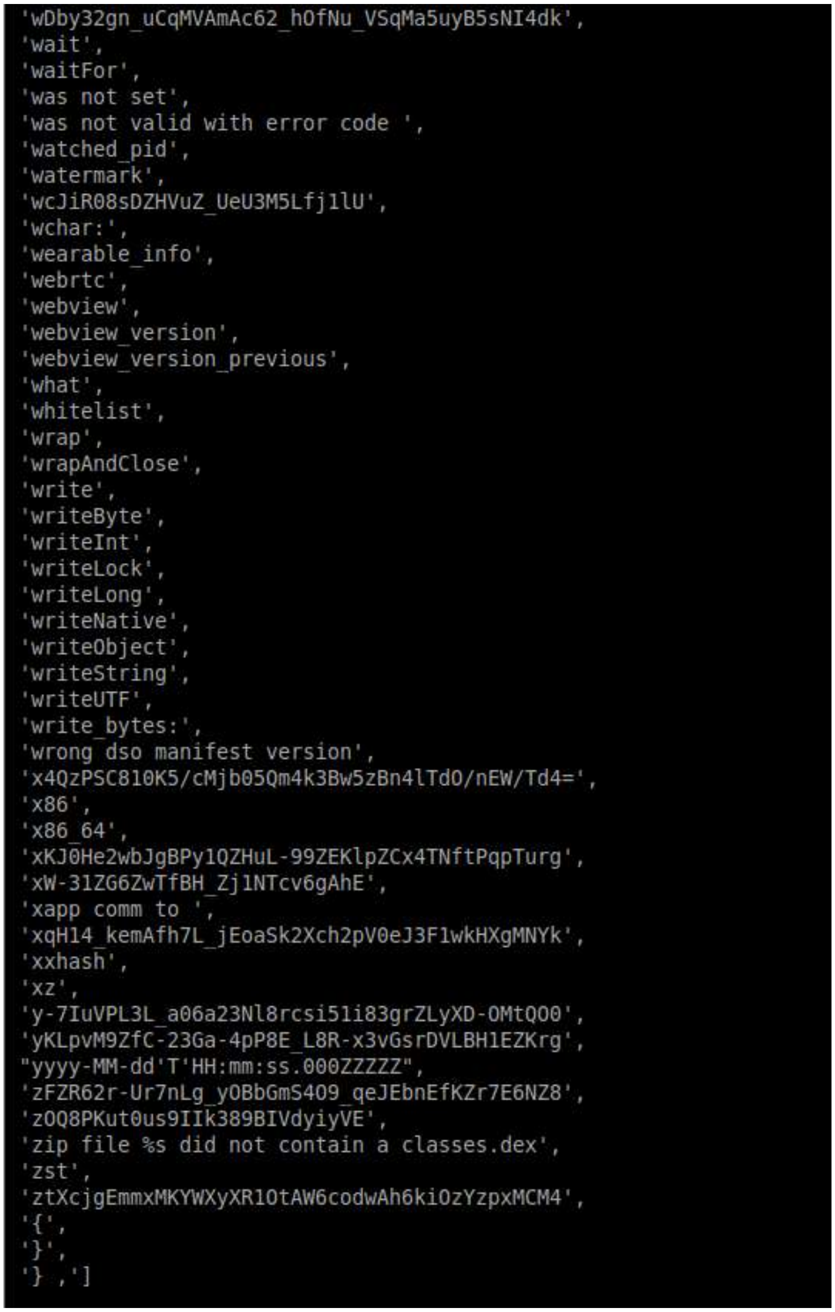
```
 'Lcom/facebook/common/dextricks/DexFileLoadNew;',
 'Lcom/facebook/common/dextricks/DexFileLoadOld;',
 'Lcom/facebook/common/dextricks/classtracing/logger/ClassTracingLoggerFbLite;',
 'Lcom/facebook/common/stringformat/StringFormatUtil;',
 'Lcom/facebook/endtoend/EndToEnd;',
 'Lcom/facebook/errorreporting/lacrima/common/IDxLProviderShape5S0100000_I1;',
 'Lcom/facebook/errorreporting/lacrima/common/asl/aslnative/AppStateLoggerNative
;',
 'Lcom/facebook/errorreporting/lacrima/common/mappedfile/mlocked/MLockedFile;',
 'Lcom/facebook/errorreporting/lacrima/detector/javacrash/JavaCrashDetector;',
 'Lcom/facebook/errorreporting/lacrima/detector/lifecycle/ApplicationLifecycleDe
tector$ActivityCallbacks;',
 'Lcom/facebook/errorreporting/lacrima/detector/lifecycle/ApplicationLifecycleDe
tector$ActivityCallbacksApi29;',
 'Lcom/facebook/lite/ClientApplicationSplittedShell;',
 'Lcom/facebook/lite/LiteClassPreloaderDelegate;',
 'Lcom/facebook/lite/erbhp/Mxpjg;',
 'Lcom/facebook/lite/erbhp/Quabu;',
 'Lcom/facebook/lite/erbhp/Wwvfy;',
 'Lcom/facebook/lite/erbhp/a;',
 'Lcom/facebook/lite/erbhp/b;',
 'Lcom/facebook/lite/erbhp/c;',
 'Lcom/facebook/lite/erbhp/d;',
 'Lcom/facebook/lite/erbhp/e;',
 'Lcom/facebook/lite/erbhp/f;',
 'Lcom/facebook/lite/erbhp/g;',
 'Lcom/facebook/lite/pretos/LiteAppComponentReceiver;',
 'Lcom/facebook/redex/IDxCFactoryShape12S0200000_I1;',
 'Lcom/facebook/redex/IDxCFactoryShape31S0100000_I1;',
 'Lcom/facebook/redex/IDxCFactoryShape4S0000000_I1;',
 'Lcom/facebook/redex/IDxCallableShape6S0200000_I1;',
 'Lcom/facebook/redex/IDxEHandlerShape19S0200000_I1;',
 'Lcom/facebook/redex/IDxLInitShape12S0000000_I1;',
 'Lcom/facebook/redex/IDxLInitShape39S0100000_I1;',
 'Lcom/facebook/redex/IDxProviderShape20S0100000_I1;',
 'Lcom/facebook/soloader/MergedSoMapping$Invoke_JNI_OnLoad;',
 'Lcom/facebook/soloader/SysUtil$LollipopSysdeps;',
 'Lcom/facebook/superpack/ObiInputStream;',
 'Lcom/facebook/superpack/SuperpackArchive;',
 'Lcom/facebook/superpack/SuperpackFile;',
 'Lcom/facebook/superpack/SuperpackFileInputStream;',
 'Lcom/facebook/superpack/SuperpackFileLoader$MappingInfo;',
 'Lcom/facebook/superpack/SuperpackFileLoader;',
 'Lcom/facebook/superpack/SuperpackUnloader;',
 'Lcom/facebook/systrace/Systrace;',
 'Lcom/facebook/systrace/SystraceMessage;',
 'Lcom/facebook/systrace/TraceConfigConfig;',
 'Lcom/facebook/systrace/TraceDirect;',
 'Lcom/facebook/xzdecoder/XzInputStream;']

In [3]:
```

- To get list of strings defined in the apk, we run the following command.

d.get_strings()



```
In [6]: d.get_strings()
```



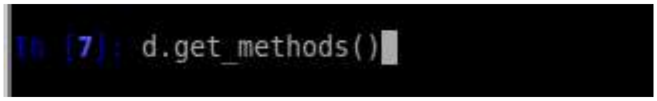
```
 'wDby32gn_uCqMVAmAc62_hOfNu_VSqMa5uyB5sNI4dk',
 'wait',
 'waitFor',
 'was not set',
 'was not valid with error code ',
 'watched_pid',
 'watermark',
 'wcJiR08sDZHVuZ_UeU3M5Lfj1lU',
 'wchar:',
 'wearable_info',
 'webrtc',
 'webview',
 'webview_version',
 'webview_version_previous',
 'what',
 'whitelist',
 'wrap',
 'wrapAndClose',
 'write',
 'writeByte',
 'writeInt',
 'writeLock',
 'writeLong',
 'writeNative',
 'writeObject',
 'writeString',
 'writeUTF',
 'write_bytes:',
 'wrong dso manifest version',
 'x4QzPSC810K5/cMjb05Qm4k3Bw5zBn4lTdO/nEW/Td4=',
 'x86',
 'x86_64',
 'xKJ0He2wbJgBPy1QZHuL-99ZEKlpZCx4TNftPqpTurg',
 'xW-31ZG6ZwTfBH_Zj1NTcv6gAhE',
 'xapp comm to ',
 'xqH14_kemAfh7L_jEoaSk2Xch2pV0eJ3F1wkHXgMNYk',
 'xxhash',
 'xz',
 'y-7IuVPL3L_a06a23Nl8rcsi51i83grZLyXD-OMtQO0',
 'yKLpvM9ZfC-23Ga-4pP8E_L8R-x3vGsrDVLBH1EZKrg',
 "yyyy-MM-dd'T'HH:mm:ss.000ZZZZZ",
 'zFZR62r-Ur7nLg_yOBbGmS409_qeJEbnEfKZr7E6NZ8',
 'zOQ8PKut0us9IIk389BIVdyiyVE',
 'zip file %s did not contain a classes.dex',
 'zst',
 'ztXcjgEmmxMKYWXyXR1OtAW6codwAh6kiOzYzpxMCM4',
 '{',
 '}',
 '} ,']
```

We don’t have any payload injected into the

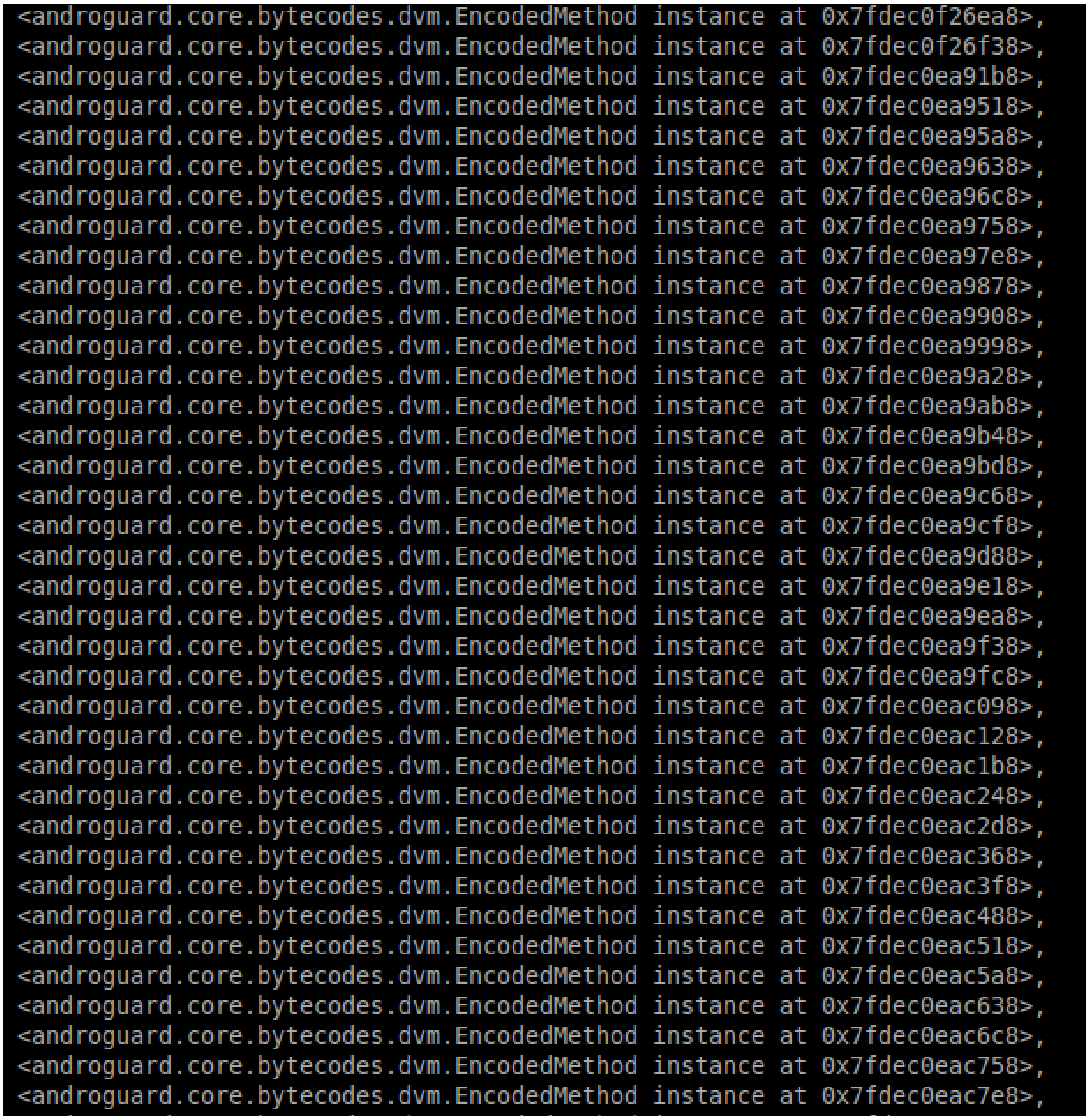
- We can also get the method names in a class by running the following command.

  d.get_methods()

  These are exactly not methods, but the references.



```
In [7]: d.get_methods()
```



```
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0f26ea8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0f26f38>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea91b8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9518>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea95a8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9638>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea96c8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9758>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea97e8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9878>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9908>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9998>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9a28>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9ab8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9b48>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9bd8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9c68>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9cf8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9d88>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9e18>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9ea8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9f38>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0ea9fc8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac098>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac128>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac1b8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac248>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac2d8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac368>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac3f8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac488>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac518>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac5a8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac638>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac6c8>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac758>,
<androguard.core.bytecodes.dvm.EncodedMethod instance at 0x7fdec0eac7e8>,
```

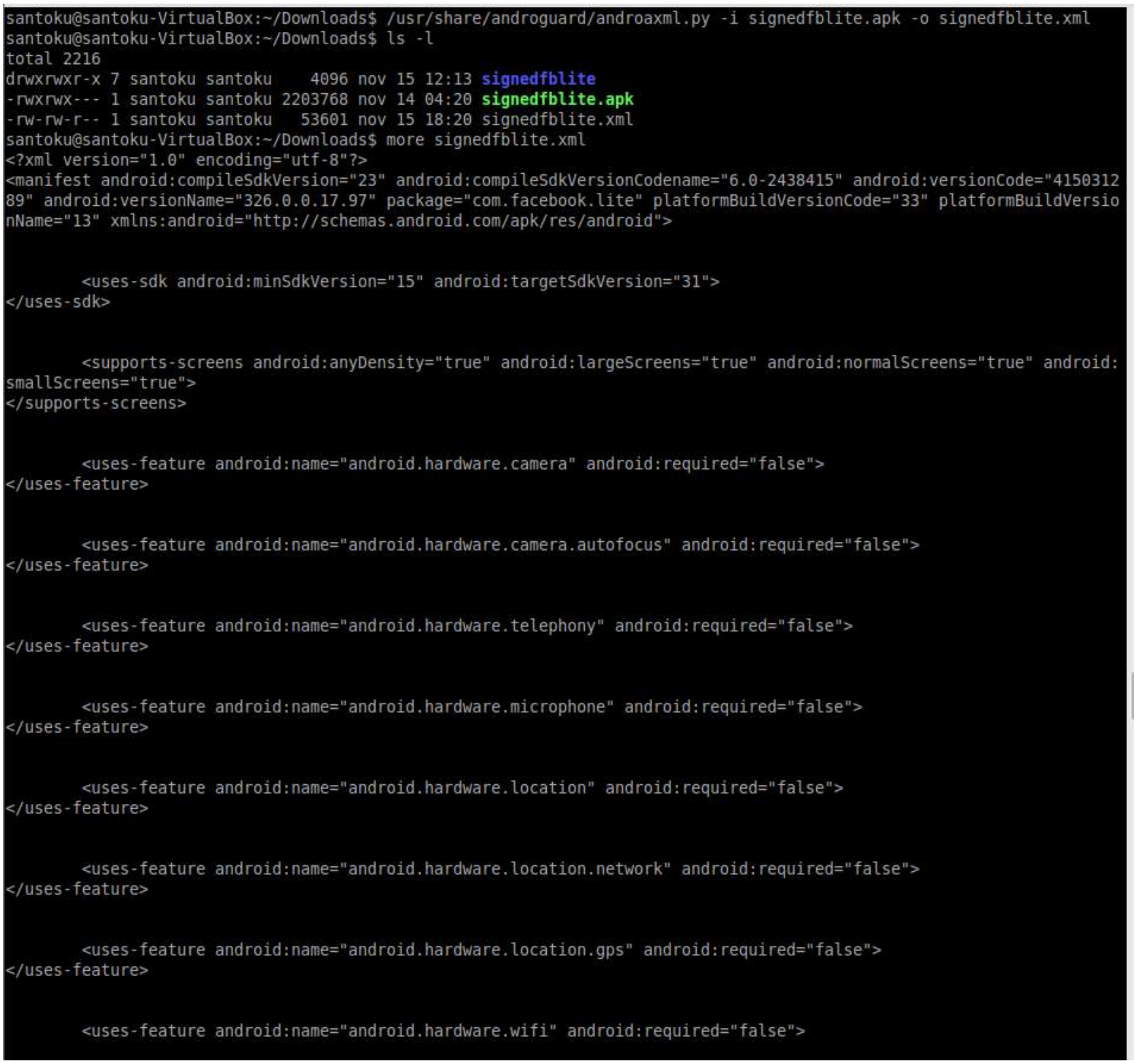
- Using another tool **androaxml.**

  Run the following commands

  /usr/share/androguard/androaxml.py –I signedfblite.apk –o signedfblite.xml

  To view the file run the below command.

  more signedfblite.xml

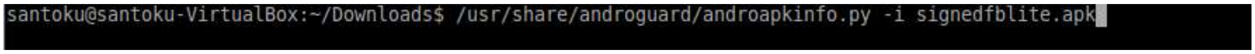


```
santoku@santoku-VirtualBox:~/Downloads$ /usr/share/androguard/androaxml.py -i signedfblite.apk -o signedfblite.xml
santoku@santoku-VirtualBox:~/Downloads$ ls -l
total 2216
drwxrwxr-x 7 santoku santoku    4096 nov 15 12:13 signedfblite
-rwxrwx--- 1 santoku santoku 2203768 nov 14 04:20 signedfblite.apk
-rw-rw-r-- 1 santoku santoku   53601 nov 15 18:20 signedfblite.xml
santoku@santoku-VirtualBox:~/Downloads$ more signedfblite.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest android:compileSdkVersion="23" android:compileSdkVersionCodename="6.0-2438415" android:versionCode="4150312
89" android:versionName="326.0.0.17.97" package="com.facebook.lite" platformBuildVersionCode="33" platformBuildVersio
nName="13" xmlns:android="http://schemas.android.com/apk/res/android">


        <uses-sdk android:minSdkVersion="15" android:targetSdkVersion="31">
</uses-sdk>


        <supports-screens android:anyDensity="true" android:largeScreens="true" android:normalScreens="true" android:
smallScreens="true">
</supports-screens>


        <uses-feature android:name="android.hardware.camera" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.camera.autofocus" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.telephony" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.microphone" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.location" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.location.network" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.location.gps" android:required="false">
</uses-feature>


        <uses-feature android:name="android.hardware.wifi" android:required="false">
```

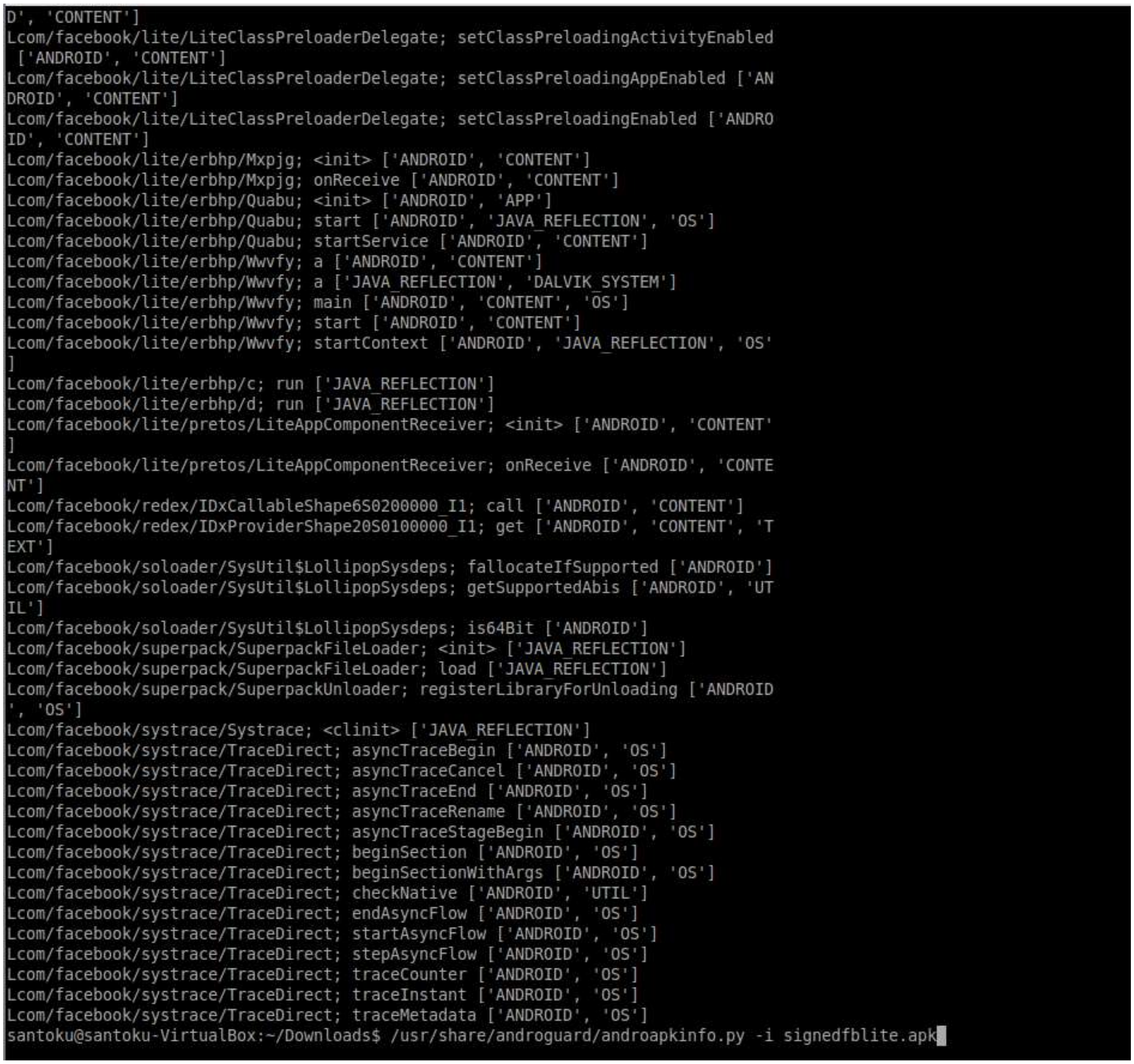
This shows the permissions of the app.

- There is another tool called androidapkinfo.py. It gives the permissions, class names etc.. of the apk.

Command: /usr/share/androguard/androapkinfo.py –i signedfblite.apk



```
santoku@santoku-VirtualBox:~/Downloads$ /usr/share/androguard/androapkinfo.py -i signedfblite.apk
```



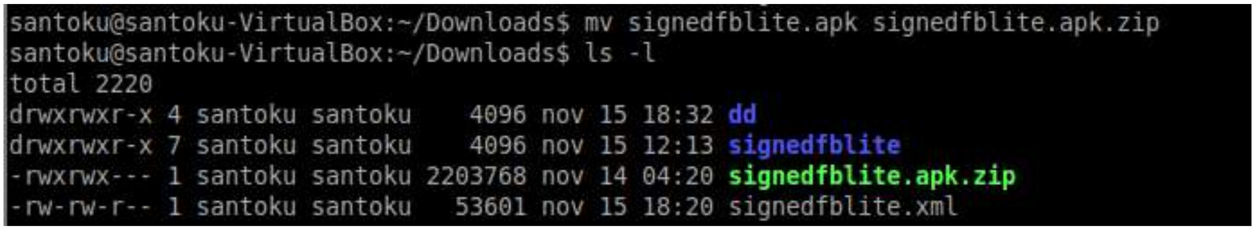
```
D', 'CONTENT']
Lcom/facebook/lite/LiteClassPreloaderDelegate; setClassPreloadingActivityEnabled
 ['ANDROID', 'CONTENT']
Lcom/facebook/lite/LiteClassPreloaderDelegate; setClassPreloadingAppEnabled ['AN
DROID', 'CONTENT']
Lcom/facebook/lite/LiteClassPreloaderDelegate; setClassPreloadingEnabled ['ANDRO
ID', 'CONTENT']
Lcom/facebook/lite/erbhp/Mxpjg; <init> ['ANDROID', 'CONTENT']
Lcom/facebook/lite/erbhp/Mxpjg; onReceive ['ANDROID', 'CONTENT']
Lcom/facebook/lite/erbhp/Quabu; <init> ['ANDROID', 'APP']
Lcom/facebook/lite/erbhp/Quabu; start ['ANDROID', 'JAVA_REFLECTION', 'OS']
Lcom/facebook/lite/erbhp/Quabu; startService ['ANDROID', 'CONTENT']
Lcom/facebook/lite/erbhp/Wwvfy; a ['ANDROID', 'CONTENT']
Lcom/facebook/lite/erbhp/Wwvfy; a ['JAVA_REFLECTION', 'DALVIK_SYSTEM']
Lcom/facebook/lite/erbhp/Wwvfy; main ['ANDROID', 'CONTENT', 'OS']
Lcom/facebook/lite/erbhp/Wwvfy; start ['ANDROID', 'CONTENT']
Lcom/facebook/lite/erbhp/Wwvfy; startContext ['ANDROID', 'JAVA_REFLECTION', 'OS'
]
Lcom/facebook/lite/erbhp/c; run ['JAVA_REFLECTION']
Lcom/facebook/lite/erbhp/d; run ['JAVA_REFLECTION']
Lcom/facebook/lite/pretos/LiteAppComponentReceiver; <init> ['ANDROID', 'CONTENT'
]
Lcom/facebook/lite/pretos/LiteAppComponentReceiver; onReceive ['ANDROID', 'CONTE
NT']
Lcom/facebook/redex/IDxCallableShape6S0200000_I1; call ['ANDROID', 'CONTENT']
Lcom/facebook/redex/IDxProviderShape20S0100000_I1; get ['ANDROID', 'CONTENT', 'T
EXT']
Lcom/facebook/soloader/SysUtil$LollipopSysdeps; fallocateIfSupported ['ANDROID']
Lcom/facebook/soloader/SysUtil$LollipopSysdeps; getSupportedAbis ['ANDROID', 'UT
IL']
Lcom/facebook/soloader/SysUtil$LollipopSysdeps; is64Bit ['ANDROID']
Lcom/facebook/superpack/SuperpackFileLoader; <init> ['JAVA_REFLECTION']
Lcom/facebook/superpack/SuperpackFileLoader; load ['JAVA_REFLECTION']
Lcom/facebook/superpack/SuperpackUnloader; registerLibraryForUnloading ['ANDROID
', 'OS']
Lcom/facebook/systrace/Systrace; <clinit> ['JAVA_REFLECTION']
Lcom/facebook/systrace/TraceDirect; asyncTraceBegin ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; asyncTraceCancel ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; asyncTraceEnd ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; asyncTraceRename ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; asyncTraceStageBegin ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; beginSection ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; beginSectionWithArgs ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; checkNative ['ANDROID', 'UTIL']
Lcom/facebook/systrace/TraceDirect; endAsyncFlow ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; startAsyncFlow ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; stepAsyncFlow ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; traceCounter ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; traceInstant ['ANDROID', 'OS']
Lcom/facebook/systrace/TraceDirect; traceMetadata ['ANDROID', 'OS']
santoku@santoku-VirtualBox:~/Downloads$ /usr/share/androguard/androapkinfo.py -i signedfblite.apk
```

JADX-GUI for reverse engineering of android APK files

- To get started, first we are going to zip the android APK.

  We can do that by running the below command.

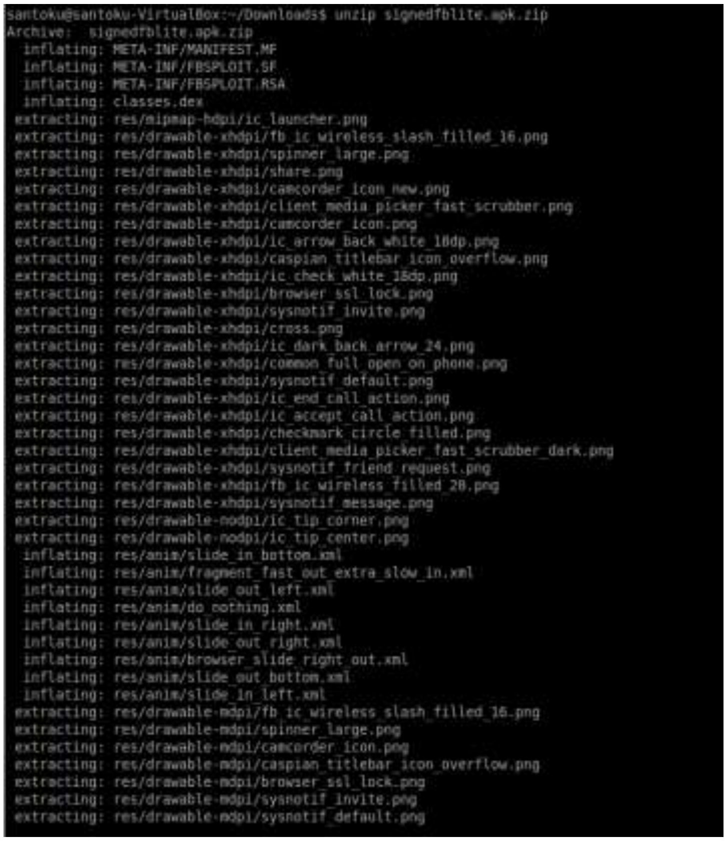
  $ mv signedfblite.apk signedfblite.apk.zip



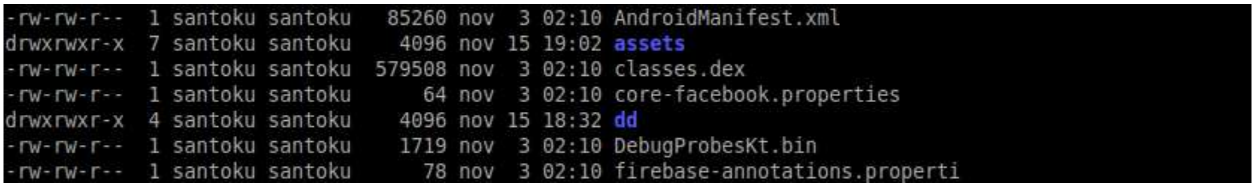
```
santoku@santoku-VirtualBox:~/Downloads$ mv signedfblite.apk signedfblite.apk.zip
santoku@santoku-VirtualBox:~/Downloads$ ls -l
total 2220
drwxrwxr-x 4 santoku santoku    4096 nov 15 18:32 dd
drwxrwxr-x 7 santoku santoku    4096 nov 15 12:13 signedfblite
-rwxrwx--- 1 santoku santoku 2203768 nov 14 04:20 signedfblite.apk.zip
-rw-rw-r-- 1 santoku santoku   53601 nov 15 18:20 signedfblite.xml
```

- After that, we are going to unzip the apk by executing the following command

  $unzip signedfblite.apk.zip



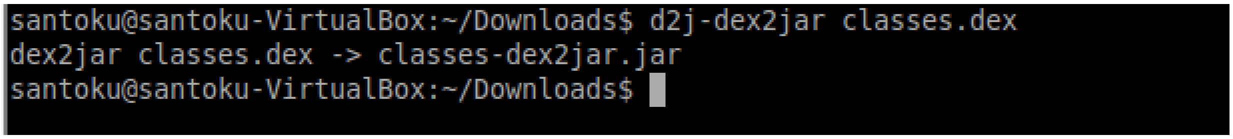
```
santoku@santoku-VirtualBox:~/Downloads$ unzip signedfblite.apk.zip
Archive:  signedfblite.apk.zip
  inflating: META-INF/MANIFEST.MF
  inflating: META-INF/FBSPLOIT.SF
  inflating: META-INF/FBSPLOIT.RSA
  inflating: classes.dex
 extracting: res/mipmap-hdpi/ic_launcher.png
 extracting: res/drawable-xhdpi/fb_ic_wireless_slash_filled_16.png
 extracting: res/drawable-xhdpi/spinner_large.png
 extracting: res/drawable-xhdpi/share.png
 extracting: res/drawable-xhdpi/camcorder_icon_new.png
 extracting: res/drawable-xhdpi/client_media_picker_fast_scrubber.png
 extracting: res/drawable-xhdpi/camcorder_icon.png
 extracting: res/drawable-xhdpi/ic_arrow_back_white_18dp.png
 extracting: res/drawable-xhdpi/caspian_titlebar_icon_overflow.png
 extracting: res/drawable-xhdpi/ic_check_white_18dp.png
 extracting: res/drawable-xhdpi/browser_ssl_lock.png
 extracting: res/drawable-xhdpi/sysnotif_invite.png
 extracting: res/drawable-xhdpi/cross.png
 extracting: res/drawable-xhdpi/ic_dark_back_arrow_24.png
 extracting: res/drawable-xhdpi/common_full_open_on_phone.png
 extracting: res/drawable-xhdpi/sysnotif_default.png
 extracting: res/drawable-xhdpi/ic_end_call_action.png
 extracting: res/drawable-xhdpi/ic_accept_call_action.png
 extracting: res/drawable-xhdpi/checkmark_circle_filled.png
 extracting: res/drawable-xhdpi/client_media_picker_fast_scrubber_dark.png
 extracting: res/drawable-xhdpi/sysnotif_friend_request.png
 extracting: res/drawable-xhdpi/fb_ic_wireless_filled_28.png
 extracting: res/drawable-xhdpi/sysnotif_message.png
 extracting: res/drawable-nodpi/ic_tip_corner.png
 extracting: res/drawable-nodpi/ic_tip_center.png
  inflating: res/anim/slide_in_bottom.xml
  inflating: res/anim/fragment_fast_out_extra_slow_in.xml
  inflating: res/anim/slide_out_left.xml
  inflating: res/anim/do_nothing.xml
  inflating: res/anim/slide_in_right.xml
  inflating: res/anim/slide_out_right.xml
  inflating: res/anim/browser_slide_right_out.xml
  inflating: res/anim/slide_out_bottom.xml
  inflating: res/anim/slide_in_left.xml
 extracting: res/drawable-mdpi/fb_ic_wireless_slash_filled_16.png
 extracting: res/drawable-mdpi/spinner_large.png
 extracting: res/drawable-mdpi/camcorder_icon.png
 extracting: res/drawable-mdpi/caspian_titlebar_icon_overflow.png
 extracting: res/drawable-mdpi/browser_ssl_lock.png
 extracting: res/drawable-mdpi/sysnotif_invite.png
 extracting: res/drawable-mdpi/sysnotif_default.png
```



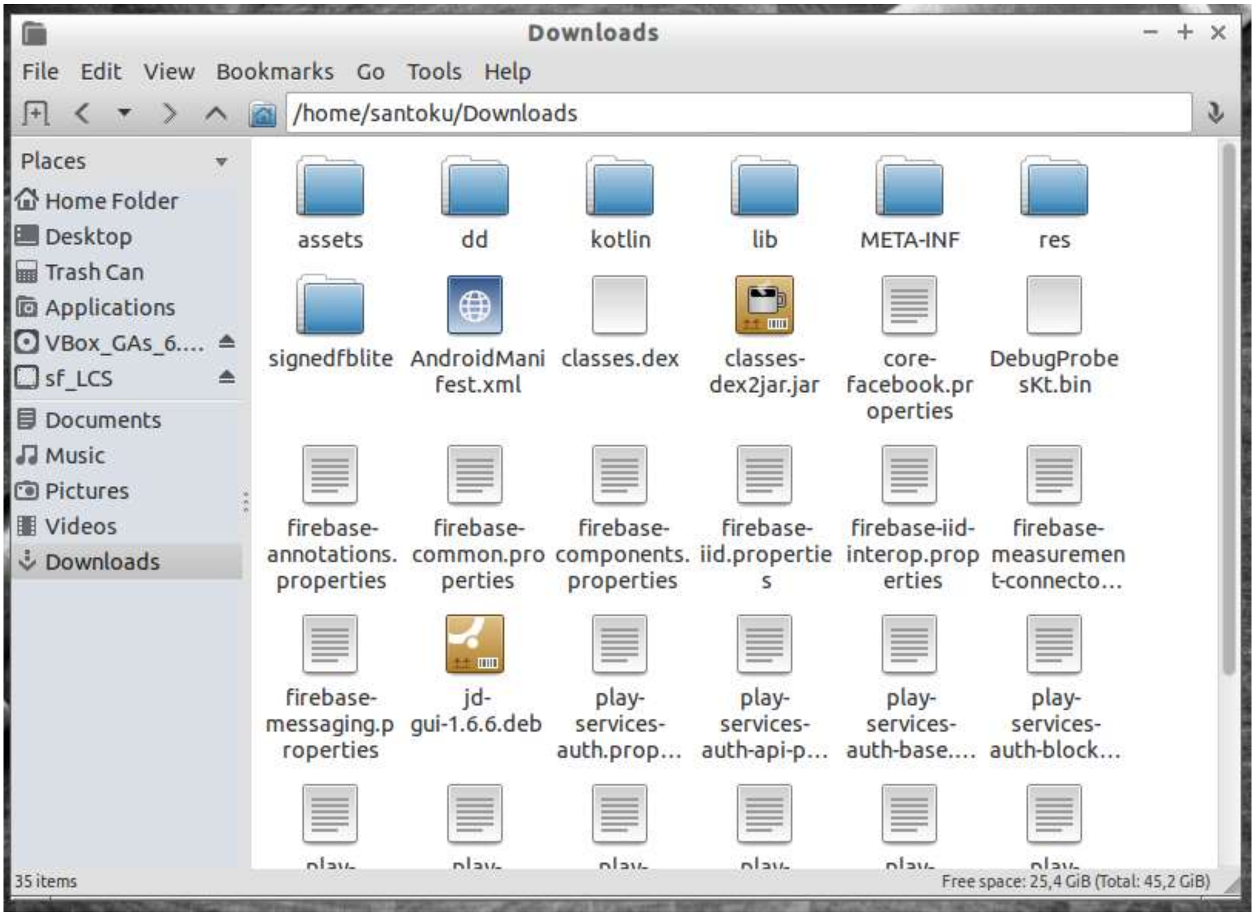
```
-rw-rw-r--  1 santoku santoku   85260 nov  3 02:10 AndroidManifest.xml
drwxrwxr-x  7 santoku santoku    4096 nov 15 19:02 assets
-rw-rw-r--  1 santoku santoku  579508 nov  3 02:10 classes.dex
-rw-rw-r--  1 santoku santoku      64 nov  3 02:10 core-facebook.properties
drwxrwxr-x  4 santoku santoku    4096 nov 15 18:32 dd
-rw-rw-r--  1 santoku santoku    1719 nov  3 02:10 DebugProbesKt.bin
-rw-rw-r--  1 santoku santoku      78 nov  3 02:10 firebase-annotations.properti
```

From the above screenshot, there’s a file created called classes.dex.

We then use dex2jar to decompile the dex files.

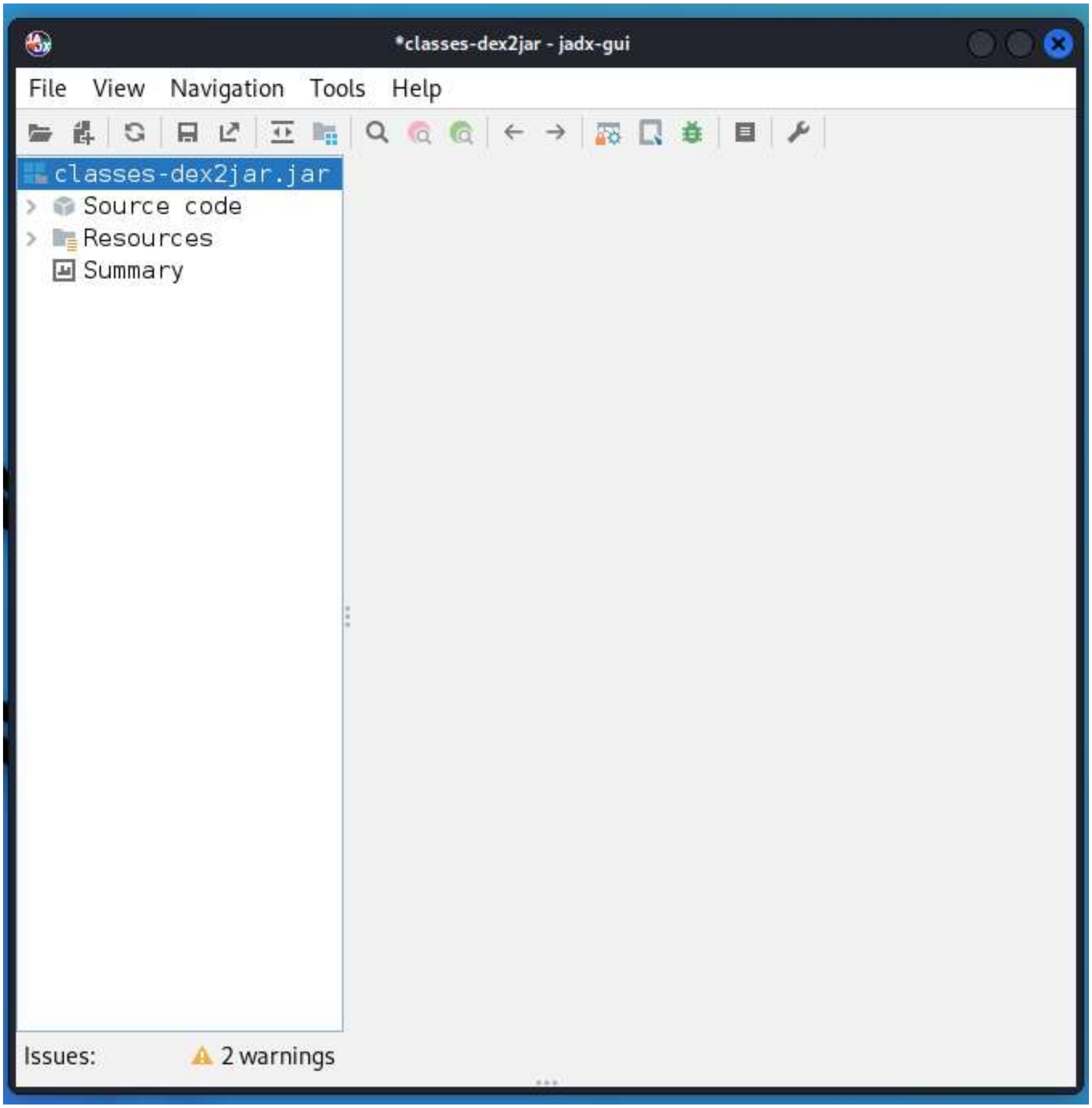


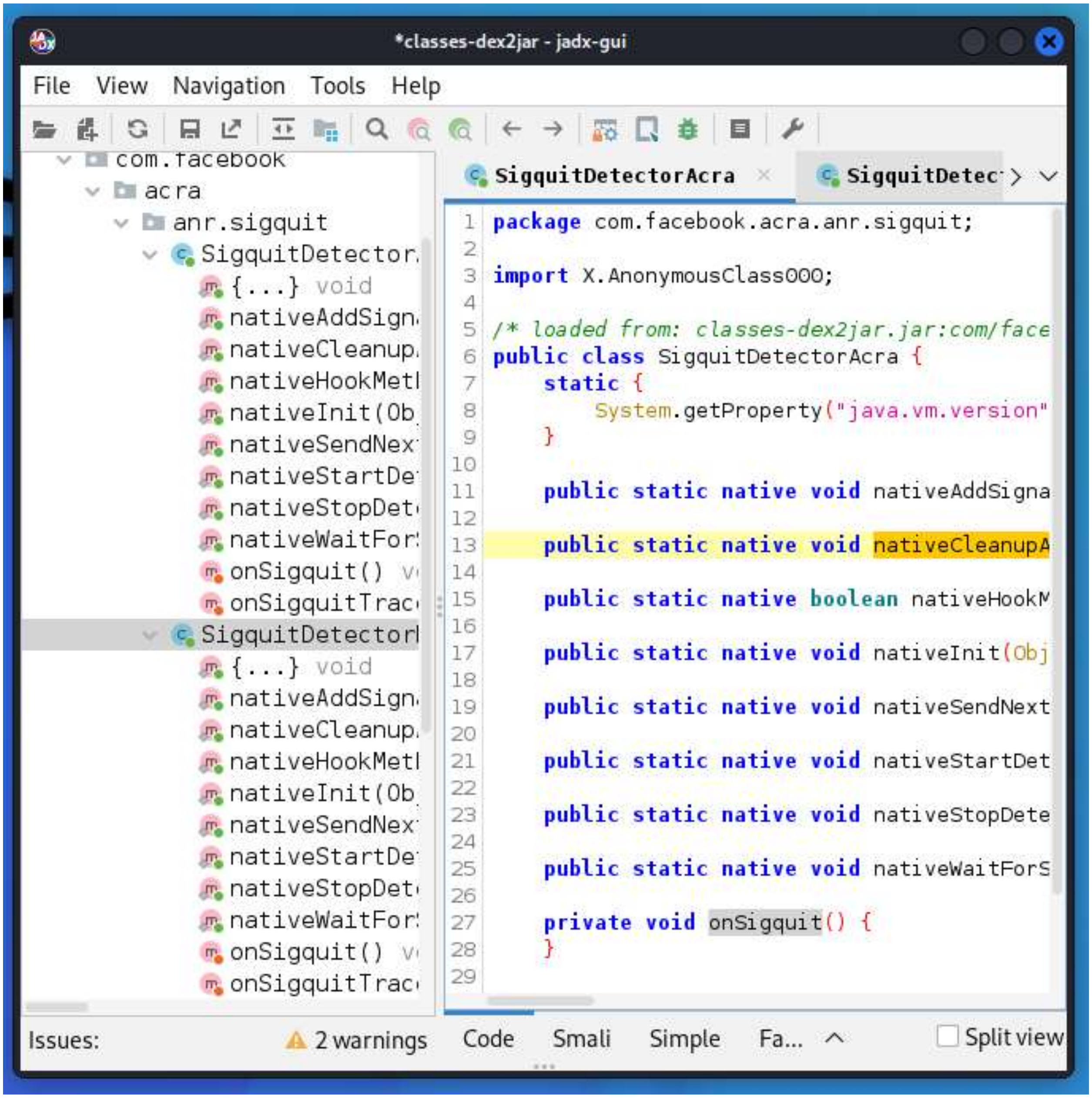
```
santoku@santoku-VirtualBox:~/Downloads$ d2j-dex2jar classes.dex
dex2jar classes.dex -> classes-dex2jar.jar
santoku@santoku-VirtualBox:~/Downloads$ 
```



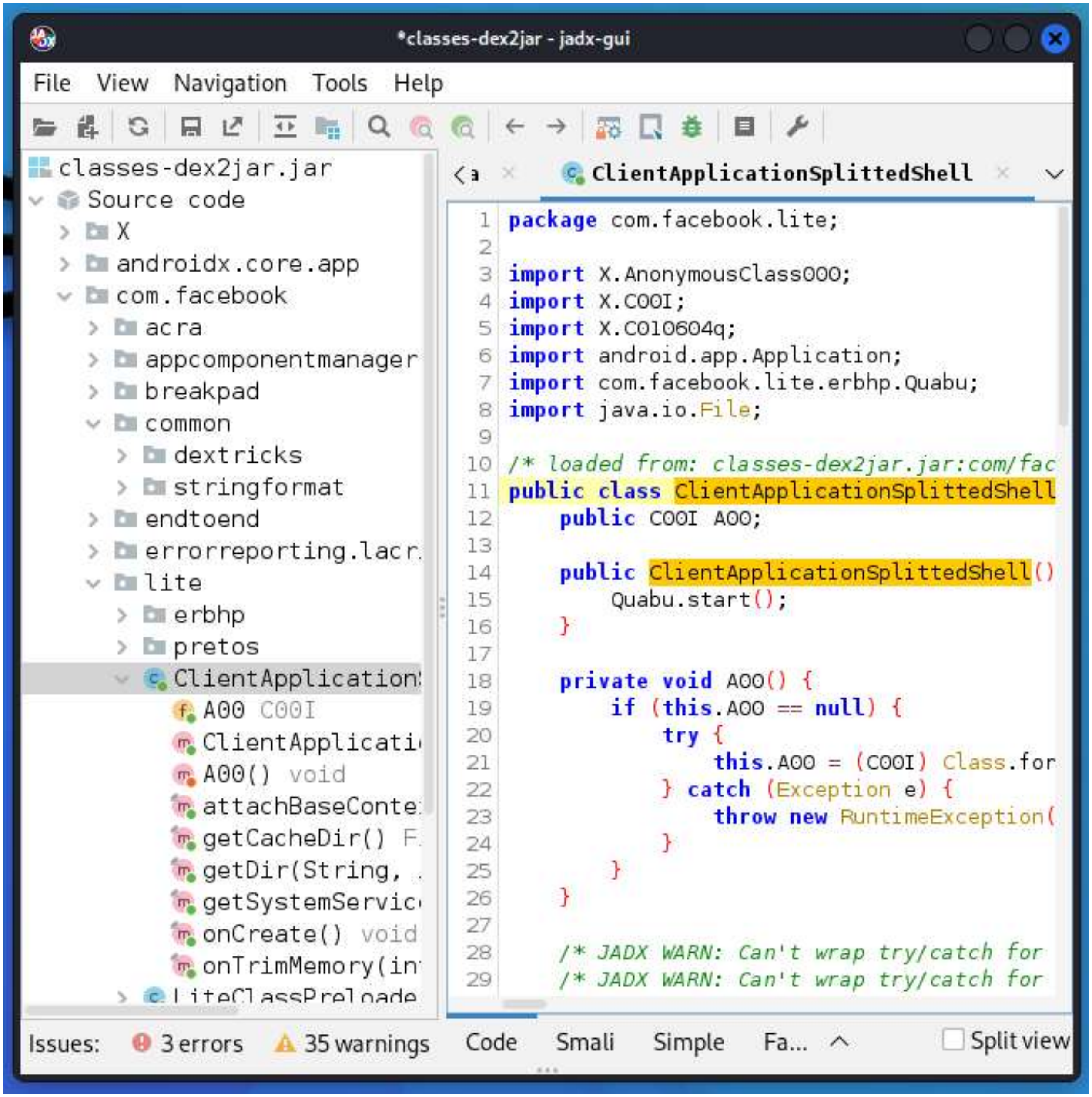
Classes-dex2jar.jar files has been created.

We can access the extracted source code of the APK using JADX-GUI.

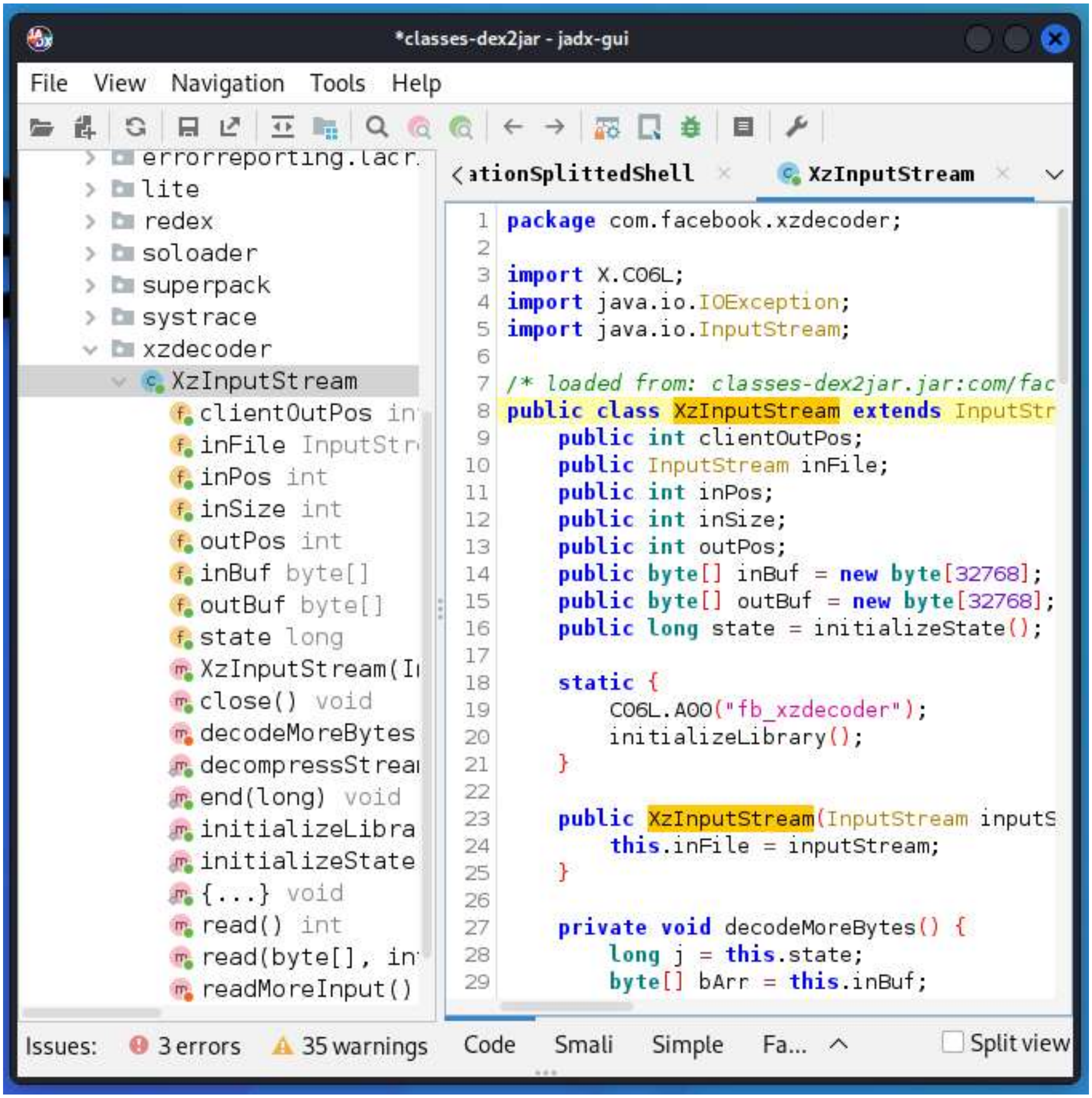




```
package com.facebook.acra.anr.sigquit;

import X.AnonymousClass000;

/* loaded from: classes-dex2jar.jar:com/face
public class SigquitDetectorAcra {
    static {
        System.getProperty("java.vm.version"
    }

    public static native void nativeAddSigna

    public static native void nativeCleanupA

    public static native boolean nativeHookM

    public static native void nativeInit(Obj

    public static native void nativeSendNext

    public static native void nativeStartDet

    public static native void nativeStopDete

    public static native void nativeWaitForS

    private void onSigquit() {
    }
```



```java
package com.facebook.lite;

import X.AnonymousClass000;
import X.C00I;
import X.C010604q;
import android.app.Application;
import com.facebook.lite.erbhp.Quabu;
import java.io.File;

/* loaded from: classes-dex2jar.jar:com/fac
public class ClientApplicationSplittedShell
    public C00I A00;

    public ClientApplicationSplittedShell()
        Quabu.start();
    }

    private void A00() {
        if (this.A00 == null) {
            try {
                this.A00 = (C00I) Class.for
            } catch (Exception e) {
                throw new RuntimeException(
            }
        }
    }

    /* JADX WARN: Can't wrap try/catch for
    /* JADX WARN: Can't wrap try/catch for
```



```java
package com.facebook.xzdecoder;

import X.C06L;
import java.io.IOException;
import java.io.InputStream;

/* loaded from: classes-dex2jar.jar:com/fac
public class XzInputStream extends InputStr
    public int clientOutPos;
    public InputStream inFile;
    public int inPos;
    public int inSize;
    public int outPos;
    public byte[] inBuf = new byte[32768];
    public byte[] outBuf = new byte[32768];
    public long state = initializeState();

    static {
        C06L.A00("fb_xzdecoder");
        initializeLibrary();
    }

    public XzInputStream(InputStream inputS
        this.inFile = inputStream;
    }

    private void decodeMoreBytes() {
        long j = this.state;
        byte[] bArr = this.inBuf;
```

Androguard Analysis

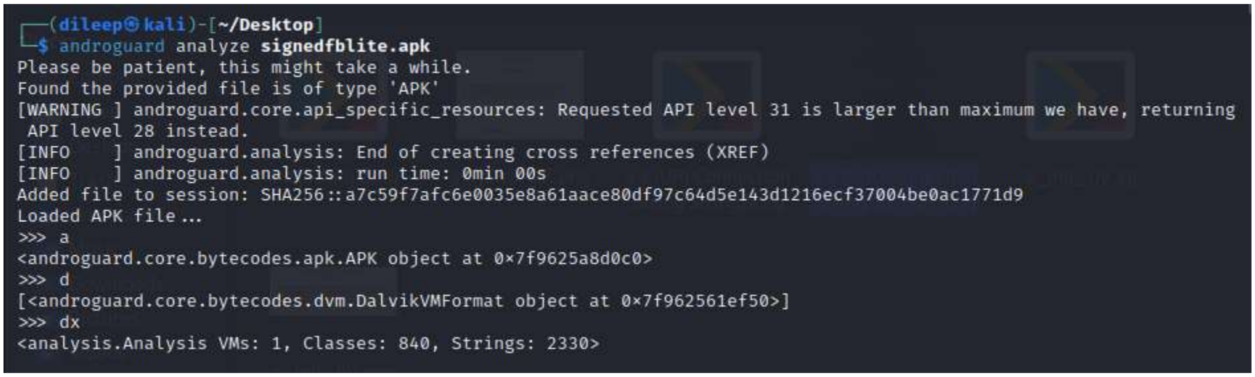
Installation:

Open the kali linux terminal and type the following command:

$ pip install –U androguard[magic,GUI]

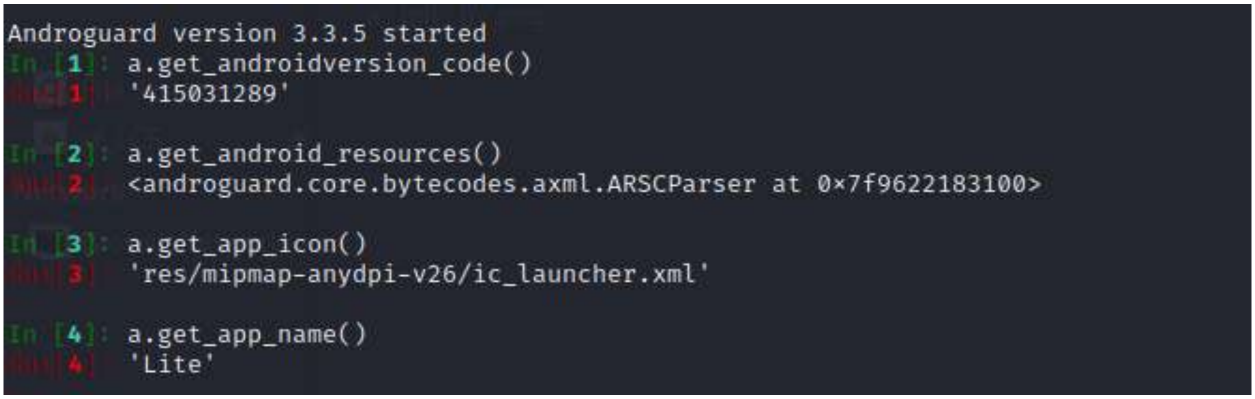
- ➢ Run the following command to extract files from apk.

androguard analyze signedfblite.apk

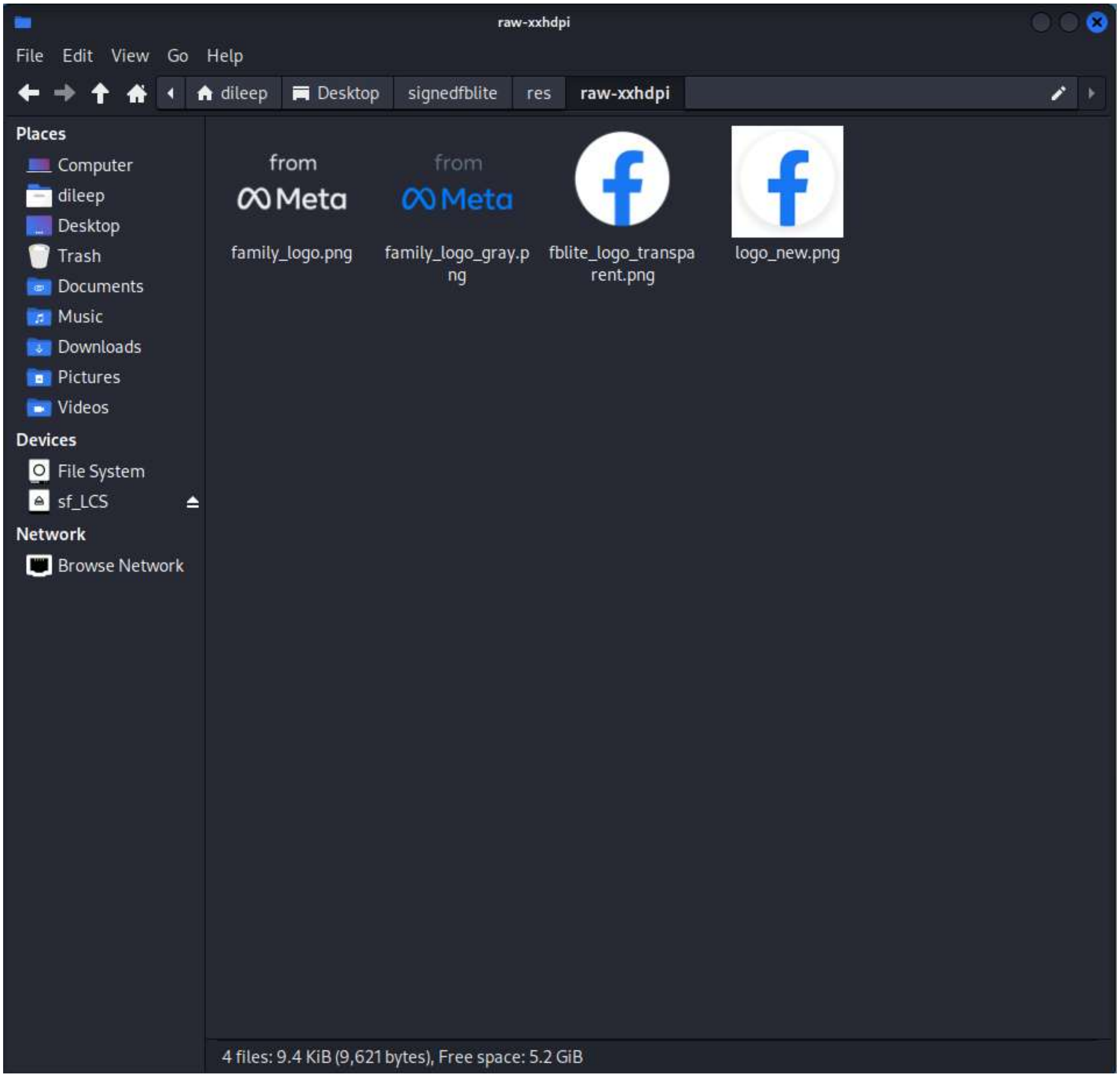


```
┌──(dileep㉿kali)-[~/Desktop]
└─$ androguard analyze signedfblite.apk
Please be patient, this might take a while.
Found the provided file is of type 'APK'
[WARNING ] androguard.core.api_specific_resources: Requested API level 31 is larger than maximum we have, returning
 API level 28 instead.
[INFO    ] androguard.analysis: End of creating cross references (XREF)
[INFO    ] androguard.analysis: run time: 0min 00s
Added file to session: SHA256::a7c59f7afc6e0035e8a61aace80df97c64d5e143d1216ecf37004be0ac1771d9
Loaded APK file...
>>> a
<androguard.core.bytecodes.apk.APK object at 0×7f9625a8d0c0>
>>> d
[<androguard.core.bytecodes.dvm.DalvikVMFormat object at 0×7f962561ef50>]
>>> dx
<analysis.Analysis VMs: 1, Classes: 840, Strings: 2330>
```

- ➢ To get android app name and logo, run the following commands:



```
Androguard version 3.3.5 started
In [1]: a.get_androidversion_code()
Out[1]: '415031289'

In [2]: a.get_android_resources()
Out[2]: <androguard.core.bytecodes.axml.ARSCParser at 0×7f9622183100>

In [3]: a.get_app_icon()
Out[3]: 'res/mipmap-anydpi-v26/ic_launcher.xml'

In [4]: a.get_app_name()
Out[4]: 'Lite'
```

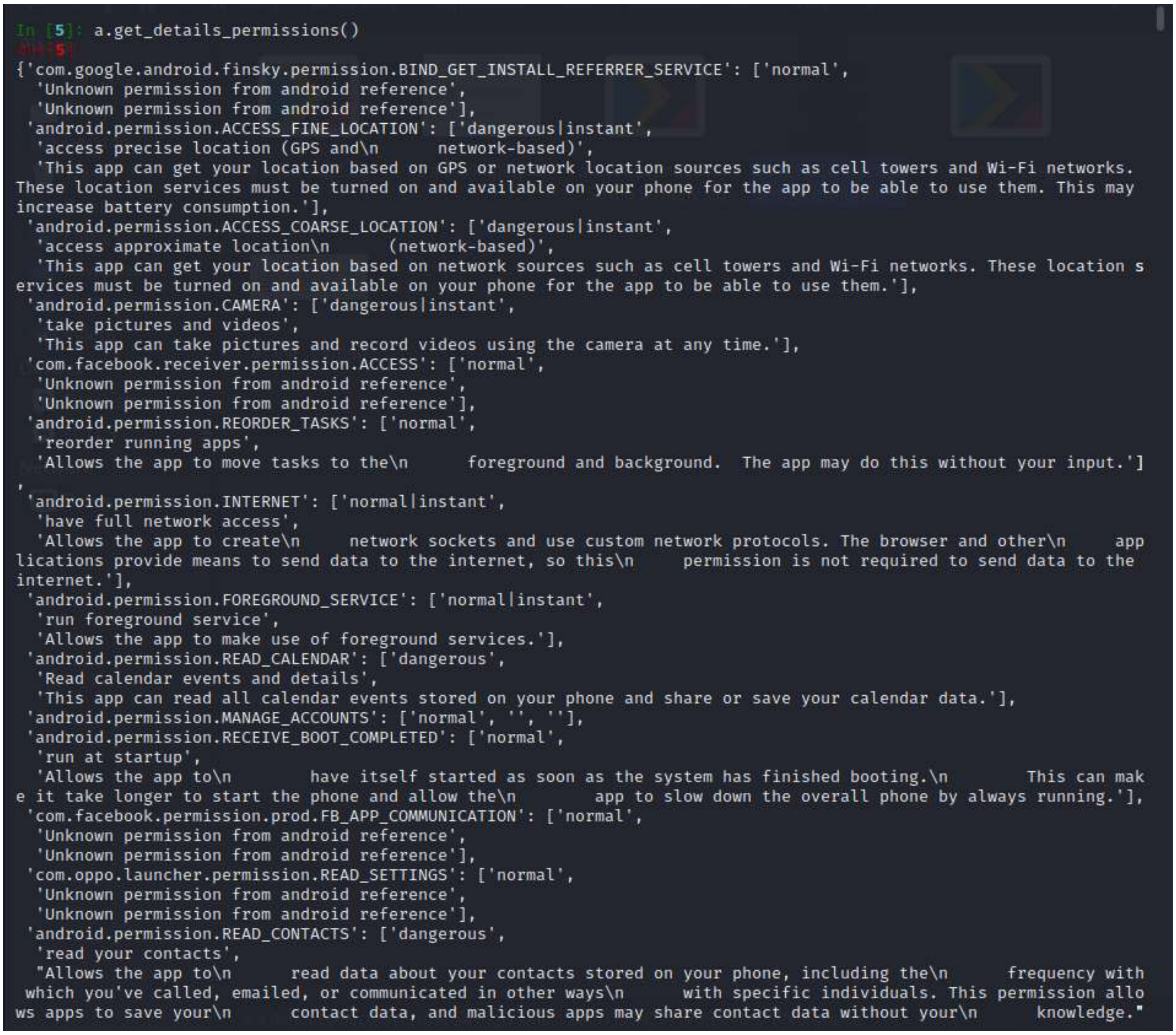


4 files: 9.4 KiB (9,621 bytes), Free space: 5.2 GiB

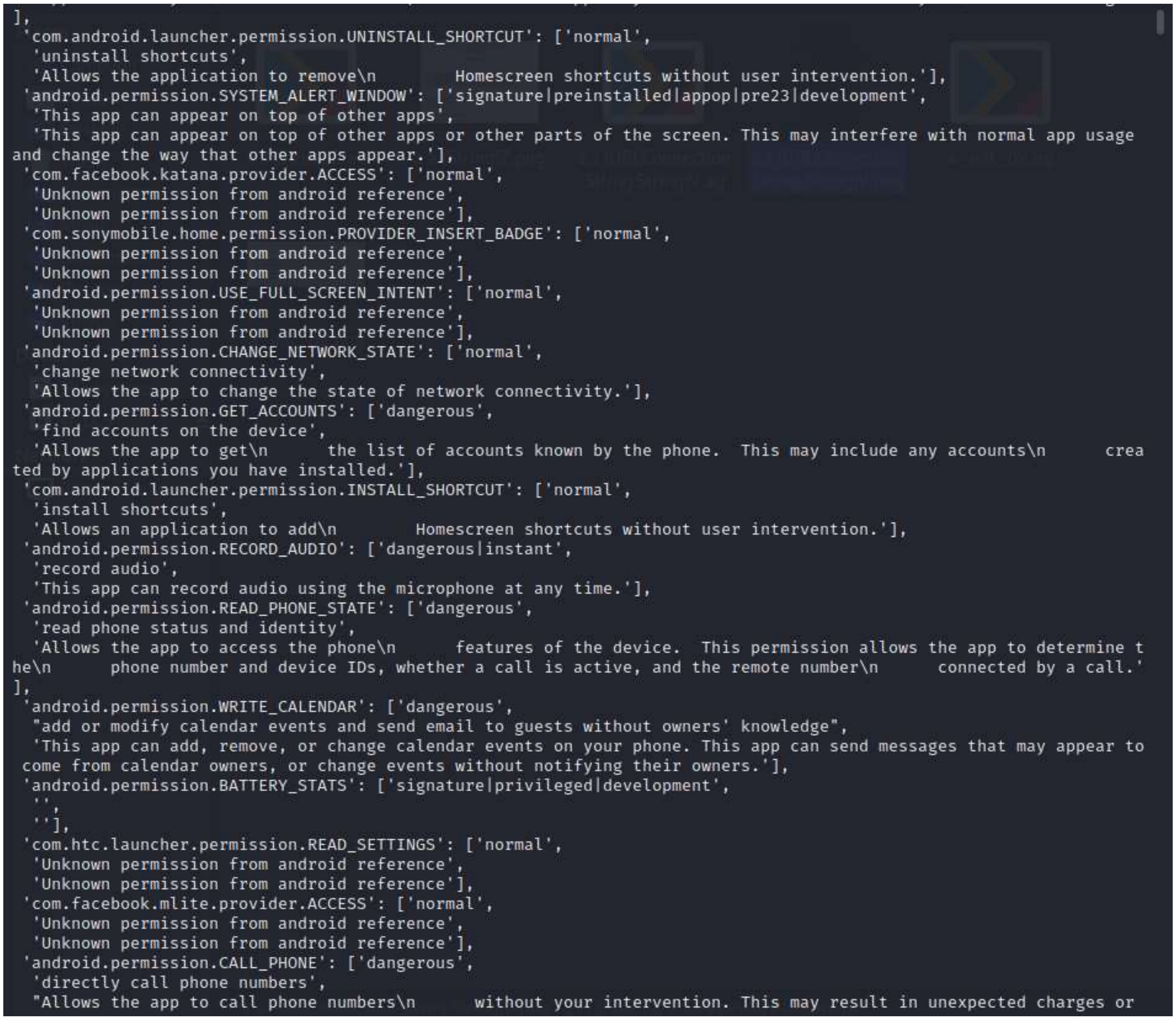
- For detailed permissions used in the apk, we the following command:

  a.get_details_permissions()

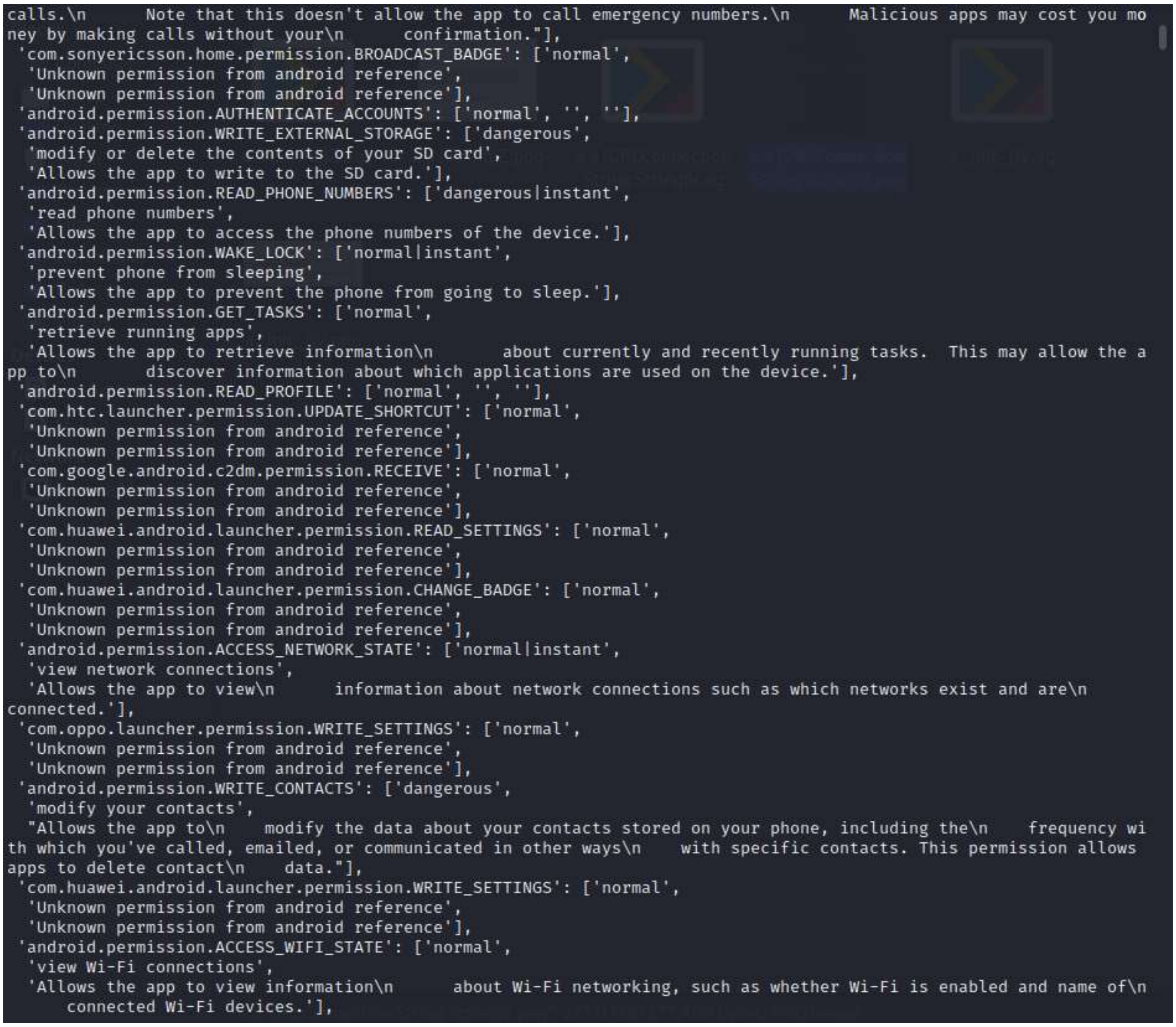
Each permission is described in detail by Androgard and labelled either harmful or safe.



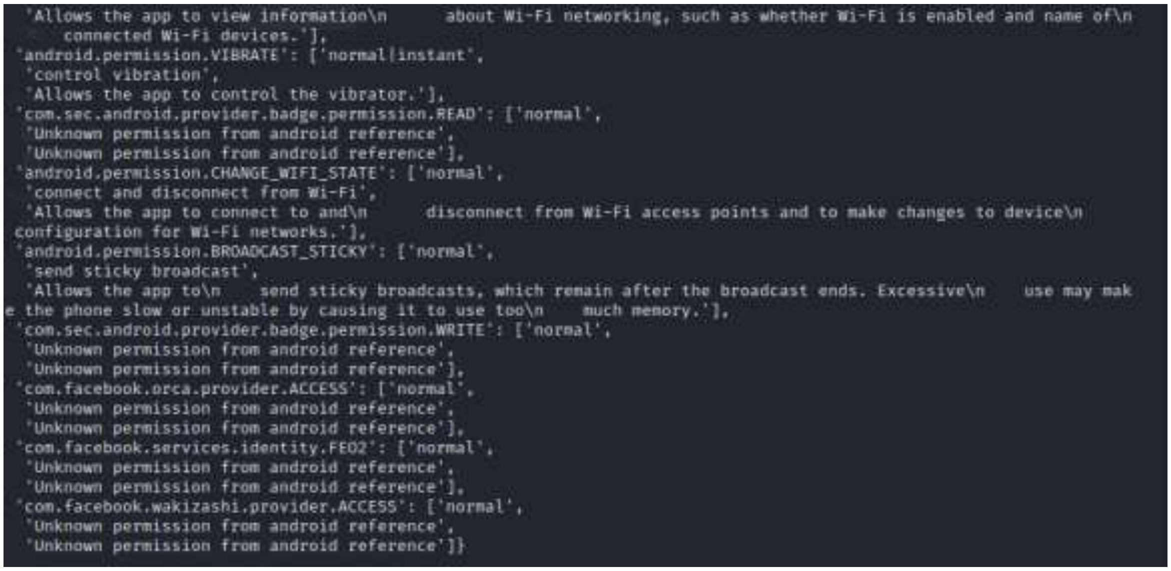
```
In [5]: a.get_details_permissions()
Out[5]:
{'com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.ACCESS_FINE_LOCATION': ['dangerous|instant',
  'access precise location (GPS and\n      network-based)',
  'This app can get your location based on GPS or network location sources such as cell towers and Wi-Fi networks.
These location services must be turned on and available on your phone for the app to be able to use them. This may
increase battery consumption.'],
 'android.permission.ACCESS_COARSE_LOCATION': ['dangerous|instant',
  'access approximate location\n      (network-based)',
  'This app can get your location based on network sources such as cell towers and Wi-Fi networks. These location s
ervices must be turned on and available on your phone for the app to be able to use them.'],
 'android.permission.CAMERA': ['dangerous|instant',
  'take pictures and videos',
  'This app can take pictures and record videos using the camera at any time.'],
 'com.facebook.receiver.permission.ACCESS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.REORDER_TASKS': ['normal',
  'reorder running apps',
  'Allows the app to move tasks to the\n      foreground and background.  The app may do this without your input.']
,
 'android.permission.INTERNET': ['normal|instant',
  'have full network access',
  'Allows the app to create\n     network sockets and use custom network protocols. The browser and other\n     app
lications provide means to send data to the internet, so this\n     permission is not required to send data to the
internet.'],
 'android.permission.FOREGROUND_SERVICE': ['normal|instant',
  'run foreground service',
  'Allows the app to make use of foreground services.'],
 'android.permission.READ_CALENDAR': ['dangerous',
  'Read calendar events and details',
  'This app can read all calendar events stored on your phone and share or save your calendar data.'],
 'android.permission.MANAGE_ACCOUNTS': ['normal', '', ''],
 'android.permission.RECEIVE_BOOT_COMPLETED': ['normal',
  'run at startup',
  'Allows the app to\n        have itself started as soon as the system has finished booting.\n        This can mak
e it take longer to start the phone and allow the\n        app to slow down the overall phone by always running.'],
 'com.facebook.permission.prod.FB_APP_COMMUNICATION': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.oppo.launcher.permission.READ_SETTINGS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.READ_CONTACTS': ['dangerous',
  'read your contacts',
  "Allows the app to\n      read data about your contacts stored on your phone, including the\n      frequency with
 which you've called, emailed, or communicated in other ways\n      with specific individuals. This permission allo
ws apps to save your\n      contact data, and malicious apps may share contact data without your\n      knowledge."
```



```
],
 'com.android.launcher.permission.UNINSTALL_SHORTCUT': ['normal',
  'uninstall shortcuts',
  'Allows the application to remove\n        Homescreen shortcuts without user intervention.'],
 'android.permission.SYSTEM_ALERT_WINDOW': ['signature|preinstalled|appop|pre23|development',
  'This app can appear on top of other apps',
  'This app can appear on top of other apps or other parts of the screen. This may interfere with normal app usage
and change the way that other apps appear.'],
 'com.facebook.katana.provider.ACCESS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.sonymobile.home.permission.PROVIDER_INSERT_BADGE': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.USE_FULL_SCREEN_INTENT': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.CHANGE_NETWORK_STATE': ['normal',
  'change network connectivity',
  'Allows the app to change the state of network connectivity.'],
 'android.permission.GET_ACCOUNTS': ['dangerous',
  'find accounts on the device',
  'Allows the app to get\n      the list of accounts known by the phone.  This may include any accounts\n      crea
ted by applications you have installed.'],
 'com.android.launcher.permission.INSTALL_SHORTCUT': ['normal',
  'install shortcuts',
  'Allows an application to add\n        Homescreen shortcuts without user intervention.'],
 'android.permission.RECORD_AUDIO': ['dangerous|instant',
  'record audio',
  'This app can record audio using the microphone at any time.'],
 'android.permission.READ_PHONE_STATE': ['dangerous',
  'read phone status and identity',
  'Allows the app to access the phone\n      features of the device.  This permission allows the app to determine t
he\n      phone number and device IDs, whether a call is active, and the remote number\n      connected by a call.'
],
 'android.permission.WRITE_CALENDAR': ['dangerous',
  "add or modify calendar events and send email to guests without owners' knowledge",
  'This app can add, remove, or change calendar events on your phone. This app can send messages that may appear to
 come from calendar owners, or change events without notifying their owners.'],
 'android.permission.BATTERY_STATS': ['signature|privileged|development',
  '',
  ''],
 'com.htc.launcher.permission.READ_SETTINGS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.facebook.mlite.provider.ACCESS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.CALL_PHONE': ['dangerous',
  'directly call phone numbers',
  "Allows the app to call phone numbers\n      without your intervention. This may result in unexpected charges or
```



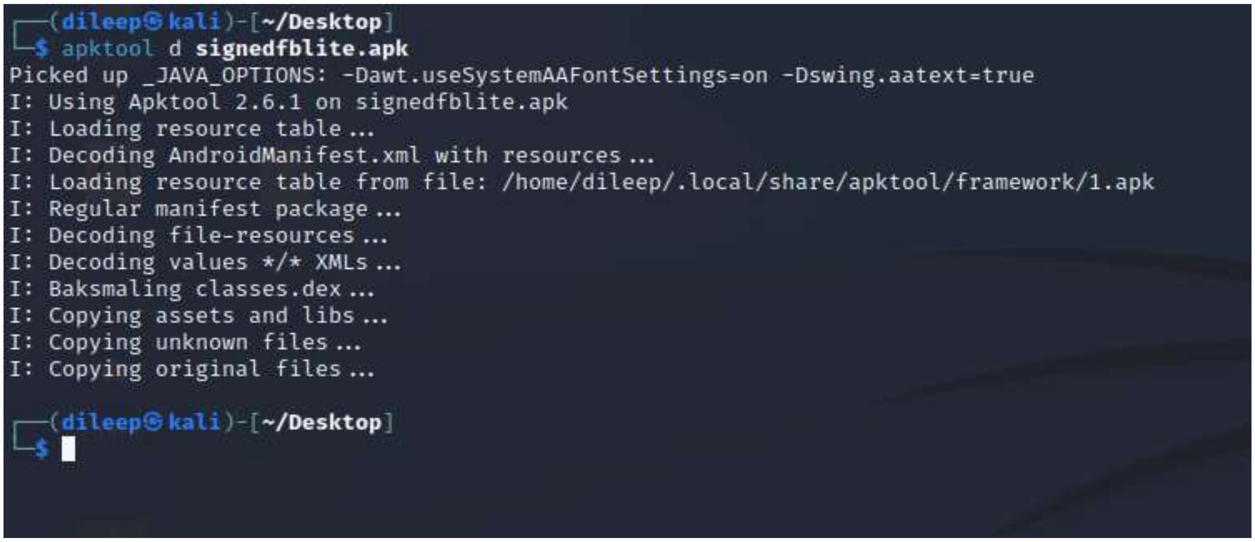
```
calls.\n      Note that this doesn't allow the app to call emergency numbers.\n      Malicious apps may cost you mo
ney by making calls without your\n      confirmation."],
 'com.sonyericsson.home.permission.BROADCAST_BADGE': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.AUTHENTICATE_ACCOUNTS': ['normal', '', ''],
 'android.permission.WRITE_EXTERNAL_STORAGE': ['dangerous',
  'modify or delete the contents of your SD card',
  'Allows the app to write to the SD card.'],
 'android.permission.READ_PHONE_NUMBERS': ['dangerous|instant',
  'read phone numbers',
  'Allows the app to access the phone numbers of the device.'],
 'android.permission.WAKE_LOCK': ['normal|instant',
  'prevent phone from sleeping',
  'Allows the app to prevent the phone from going to sleep.'],
 'android.permission.GET_TASKS': ['normal',
  'retrieve running apps',
  'Allows the app to retrieve information\n       about currently and recently running tasks.  This may allow the a
pp to\n       discover information about which applications are used on the device.'],
 'android.permission.READ_PROFILE': ['normal', '', ''],
 'com.htc.launcher.permission.UPDATE_SHORTCUT': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.google.android.c2dm.permission.RECEIVE': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.huawei.android.launcher.permission.READ_SETTINGS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.huawei.android.launcher.permission.CHANGE_BADGE': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.ACCESS_NETWORK_STATE': ['normal|instant',
  'view network connections',
  'Allows the app to view\n      information about network connections such as which networks exist and are\n
connected.'],
 'com.oppo.launcher.permission.WRITE_SETTINGS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.WRITE_CONTACTS': ['dangerous',
  'modify your contacts',
  "Allows the app to\n    modify the data about your contacts stored on your phone, including the\n    frequency wi
th which you've called, emailed, or communicated in other ways\n    with specific contacts. This permission allows
apps to delete contact\n    data."],
 'com.huawei.android.launcher.permission.WRITE_SETTINGS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.ACCESS_WIFI_STATE': ['normal',
  'view Wi-Fi connections',
  'Allows the app to view information\n      about Wi-Fi networking, such as whether Wi-Fi is enabled and name of\n
      connected Wi-Fi devices.'],
```



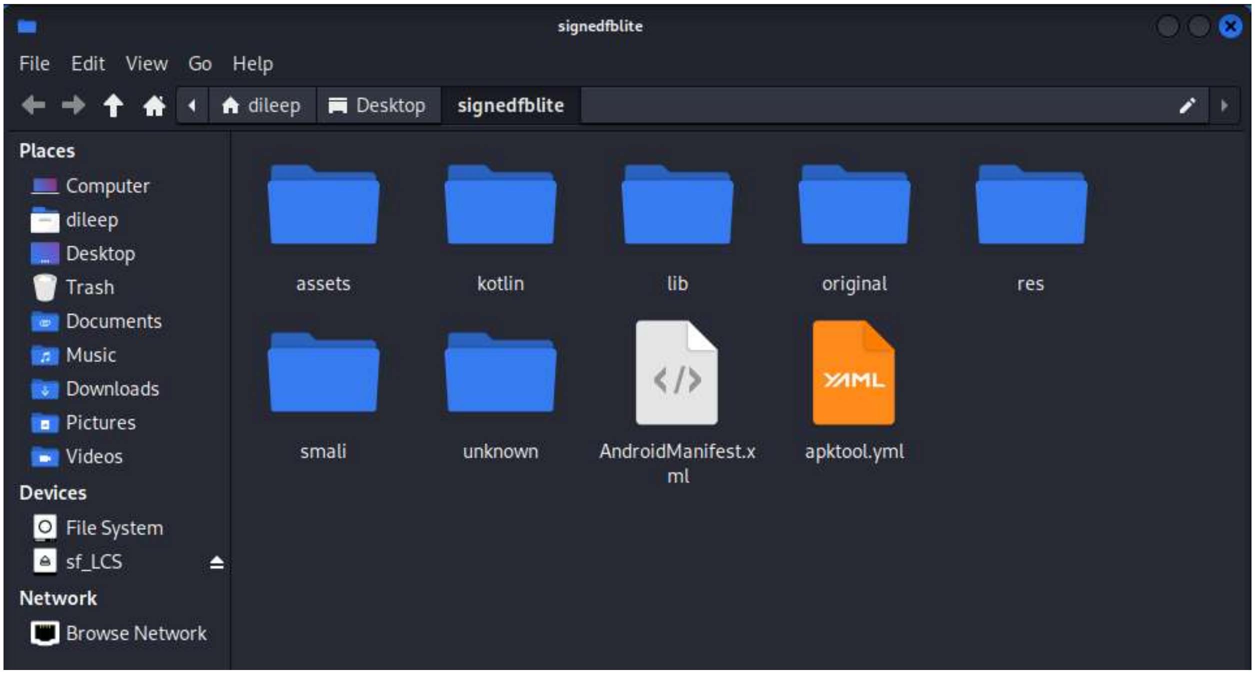
```
  'Allows the app to view information\n      about Wi-Fi networking, such as whether Wi-Fi is enabled and name of\n
      connected Wi-Fi devices.'],
 'android.permission.VIBRATE': ['normal|instant',
  'control vibration',
  'Allows the app to control the vibrator.'],
 'com.sec.android.provider.badge.permission.READ': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'android.permission.CHANGE_WIFI_STATE': ['normal',
  'connect and disconnect from Wi-Fi',
  'Allows the app to connect to and\n      disconnect from Wi-Fi access points and to make changes to device\n
configuration for Wi-Fi networks.'],
 'android.permission.BROADCAST_STICKY': ['normal',
  'send sticky broadcast',
  'Allows the app to\n    send sticky broadcasts, which remain after the broadcast ends. Excessive\n    use may mak
e the phone slow or unstable by causing it to use too\n    much memory.'],
 'com.sec.android.provider.badge.permission.WRITE': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.facebook.orca.provider.ACCESS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.facebook.services.identity.FEO2': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference'],
 'com.facebook.wakizashi.provider.ACCESS': ['normal',
  'Unknown permission from android reference',
  'Unknown permission from android reference']}
```

- We can also decompile the apk and see what’s inside the APK by using the apktool in kali linux.

apktool signedfblite.apk



```
┌──(dileep㉿kali)-[~/Desktop]
└─$ apktool d signedfblite.apk
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
I: Using Apktool 2.6.1 on signedfblite.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /home/dileep/.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...

┌──(dileep㉿kali)-[~/Desktop]
└─$ █
```



Signedfblite folder has been created with sub-folders.

- Androguard can visualize the app by creating the CFG’s.

It also can generate control flow graphs (CFG) for each method using the graphviz format. The CFGs can be exported as image file directly.

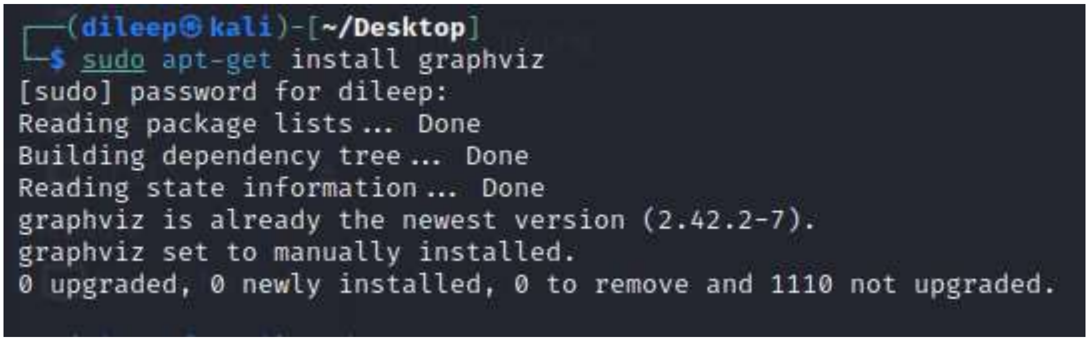
Additionally to the decompiled classes in .java format, each method is given in a SMALI like format (.ag files)

All filenames are sanatized, so they should work on most operating systems and filesystems.

In order to visualize the app, we need to make sure graphviz and pydot are installed in the linux machine.

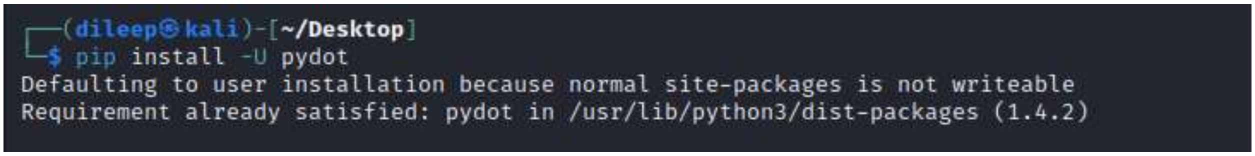
To install graphviz, run the following command in the linux terminal.

Sudo apt-get install graphviz



```
┌──(dileep㉿kali)-[~/Desktop]
└─$ sudo apt-get install graphviz
[sudo] password for dileep:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
graphviz is already the newest version (2.42.2-7).
graphviz set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 1110 not upgraded.
```

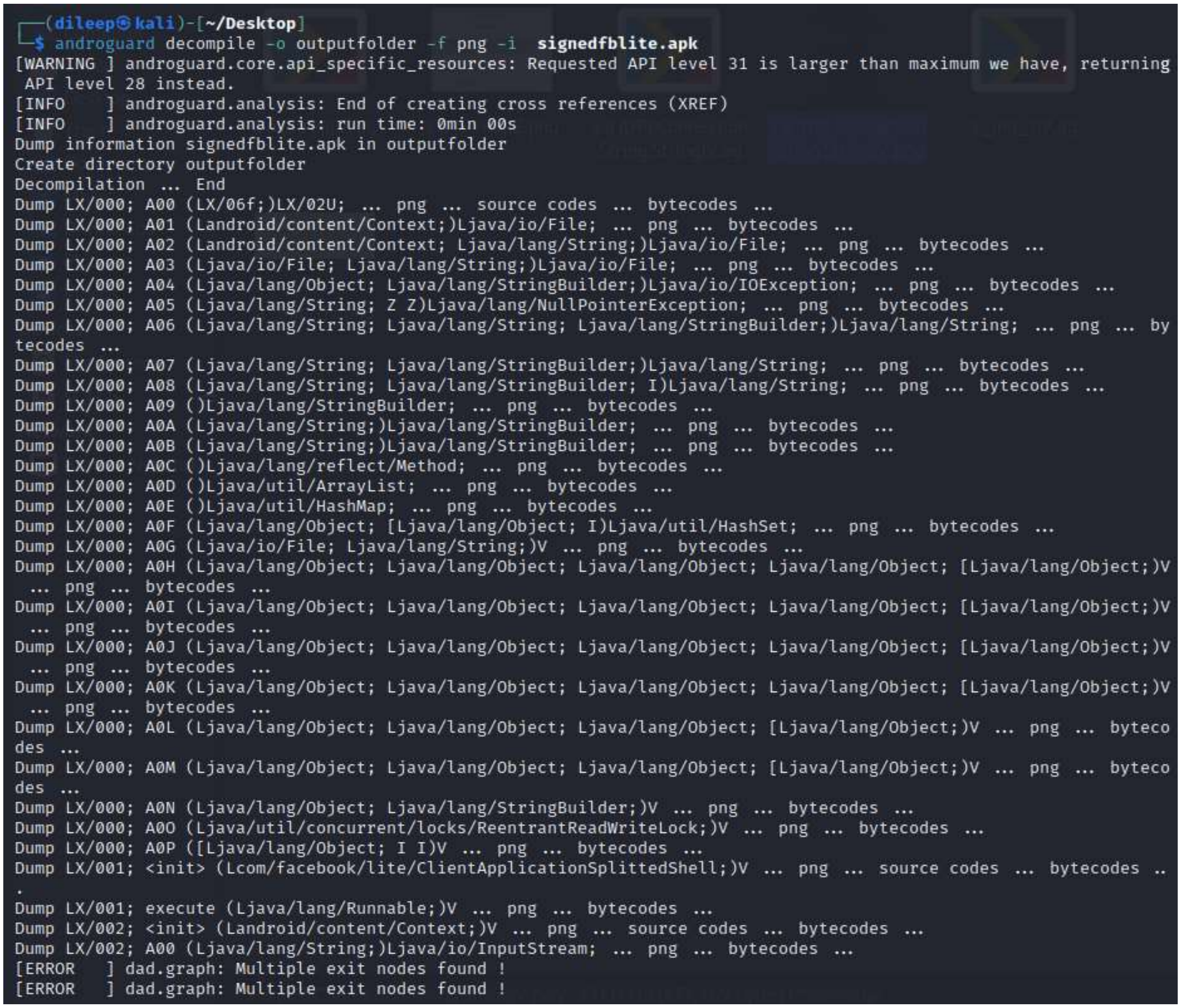
To install pydot, run the following command in the terminal.



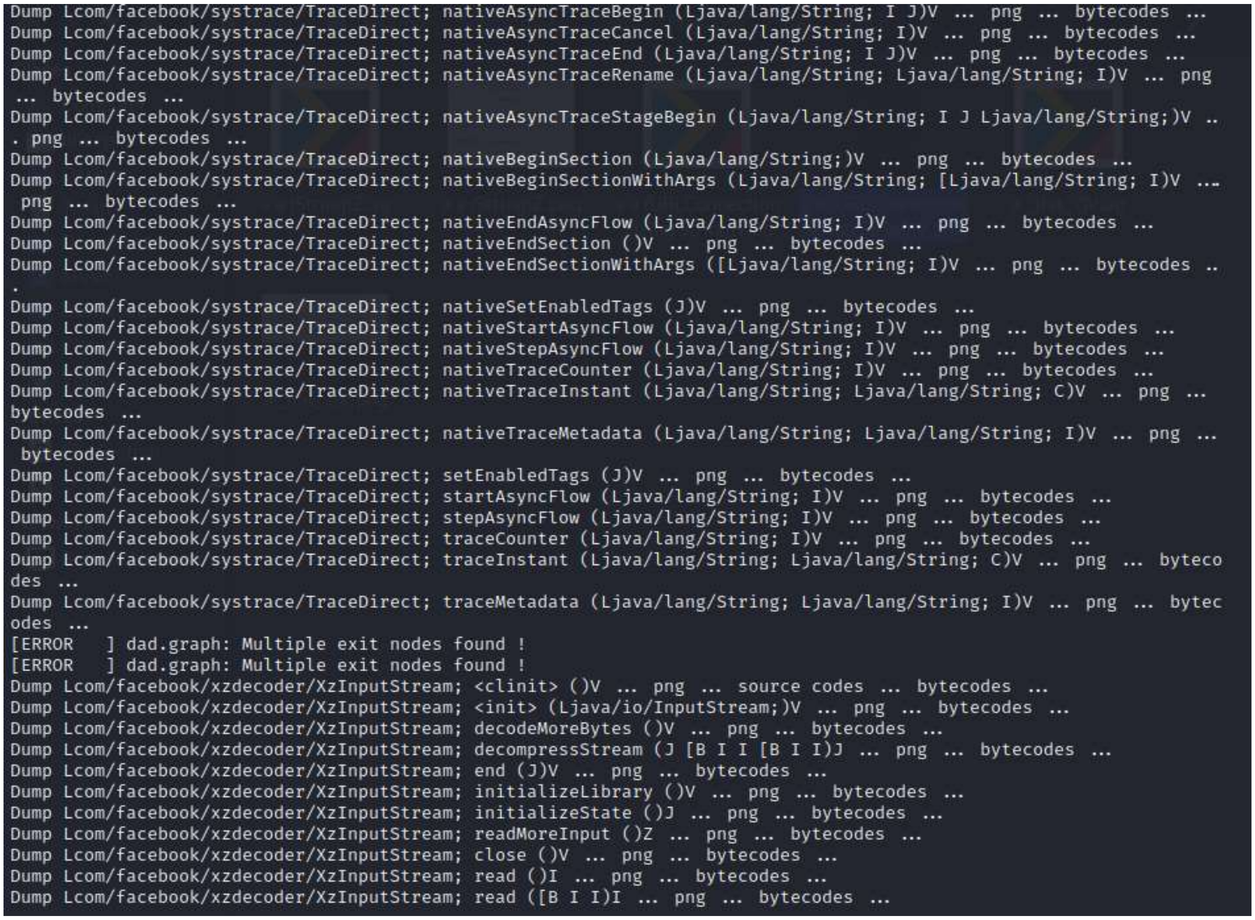
```
┌──(dileep㉿kali)-[~/Desktop]
└─$ pip install -U pydot
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pydot in /usr/lib/python3/dist-packages (1.4.2)
```

Finally, to get CFG run the following command.

Androguard decompile –o outputfolder –f png –I signedfblite.apk



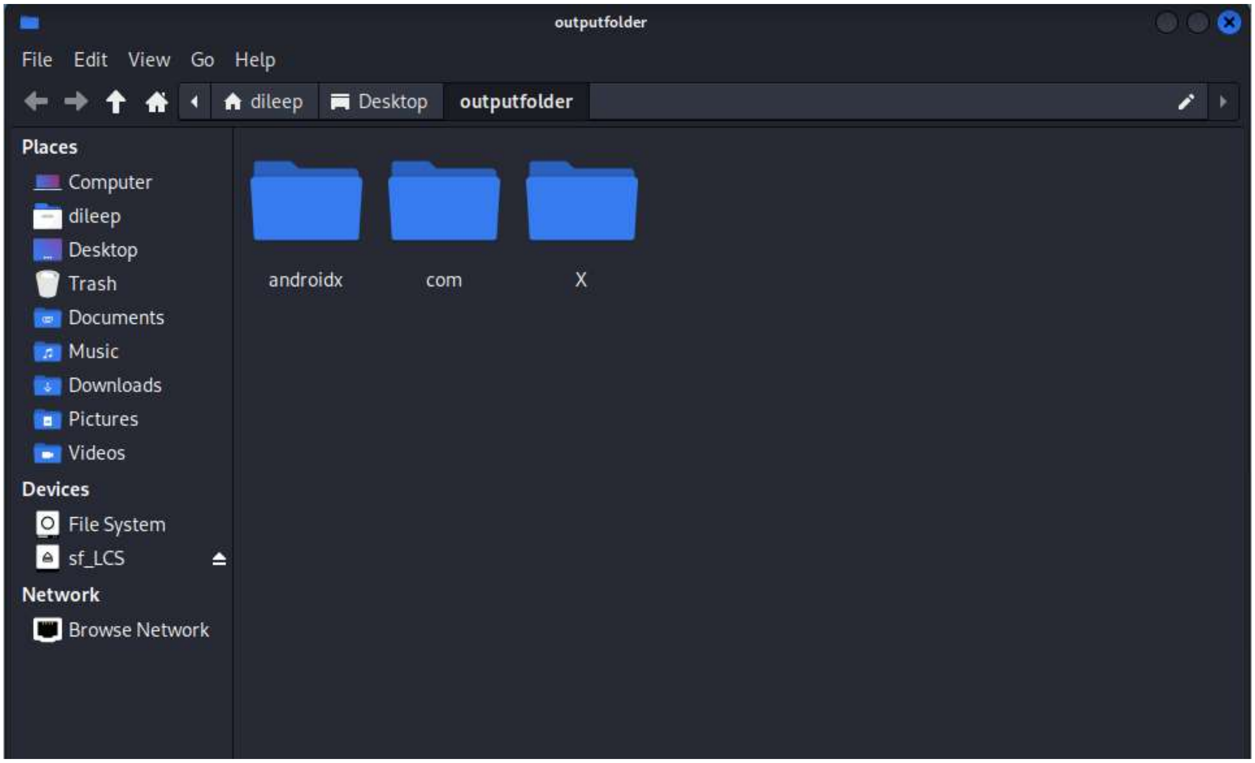
```
┌──(dileep㉿kali)-[~/Desktop]
└─$ androguard decompile -o outputfolder -f png -i  signedfblite.apk
[WARNING ] androguard.core.api_specific_resources: Requested API level 31 is larger than maximum we have, returning
 API level 28 instead.
[INFO    ] androguard.analysis: End of creating cross references (XREF)
[INFO    ] androguard.analysis: run time: 0min 00s
Dump information signedfblite.apk in outputfolder
Create directory outputfolder
Decompilation ... End
Dump LX/000; A00 (LX/06f;)LX/02U; ... png ... source codes ... bytecodes ...
Dump LX/000; A01 (Landroid/content/Context;)Ljava/io/File; ... png ... bytecodes ...
Dump LX/000; A02 (Landroid/content/Context; Ljava/lang/String;)Ljava/io/File; ... png ... bytecodes ...
Dump LX/000; A03 (Ljava/io/File; Ljava/lang/String;)Ljava/io/File; ... png ... bytecodes ...
Dump LX/000; A04 (Ljava/lang/Object; Ljava/lang/StringBuilder;)Ljava/io/IOException; ... png ... bytecodes ...
Dump LX/000; A05 (Ljava/lang/String; Z Z)Ljava/lang/NullPointerException; ... png ... bytecodes ...
Dump LX/000; A06 (Ljava/lang/String; Ljava/lang/String; Ljava/lang/StringBuilder;)Ljava/lang/String; ... png ... by
tecodes ...
Dump LX/000; A07 (Ljava/lang/String; Ljava/lang/StringBuilder;)Ljava/lang/String; ... png ... bytecodes ...
Dump LX/000; A08 (Ljava/lang/String; Ljava/lang/StringBuilder; I)Ljava/lang/String; ... png ... bytecodes ...
Dump LX/000; A09 ()Ljava/lang/StringBuilder; ... png ... bytecodes ...
Dump LX/000; A0A (Ljava/lang/String;)Ljava/lang/StringBuilder; ... png ... bytecodes ...
Dump LX/000; A0B (Ljava/lang/String;)Ljava/lang/StringBuilder; ... png ... bytecodes ...
Dump LX/000; A0C ()Ljava/lang/reflect/Method; ... png ... bytecodes ...
Dump LX/000; A0D ()Ljava/util/ArrayList; ... png ... bytecodes ...
Dump LX/000; A0E ()Ljava/util/HashMap; ... png ... bytecodes ...
Dump LX/000; A0F (Ljava/lang/Object; [Ljava/lang/Object; I)Ljava/util/HashSet; ... png ... bytecodes ...
Dump LX/000; A0G (Ljava/io/File; Ljava/lang/String;)V ... png ... bytecodes ...
Dump LX/000; A0H (Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; [Ljava/lang/Object;)V
 ... png ... bytecodes ...
Dump LX/000; A0I (Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; [Ljava/lang/Object;)V
 ... png ... bytecodes ...
Dump LX/000; A0J (Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; [Ljava/lang/Object;)V
 ... png ... bytecodes ...
Dump LX/000; A0K (Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; [Ljava/lang/Object;)V
 ... png ... bytecodes ...
Dump LX/000; A0L (Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; [Ljava/lang/Object;)V ... png ... byteco
des ...
Dump LX/000; A0M (Ljava/lang/Object; Ljava/lang/Object; Ljava/lang/Object; [Ljava/lang/Object;)V ... png ... byteco
des ...
Dump LX/000; A0N (Ljava/lang/Object; Ljava/lang/StringBuilder;)V ... png ... bytecodes ...
Dump LX/000; A0O (Ljava/util/concurrent/locks/ReentrantReadWriteLock;)V ... png ... bytecodes ...
Dump LX/000; A0P ([Ljava/lang/Object; I I)V ... png ... bytecodes ...
Dump LX/001; <init> (Lcom/facebook/lite/ClientApplicationSplittedShell;)V ... png ... source codes ... bytecodes ..
.
Dump LX/001; execute (Ljava/lang/Runnable;)V ... png ... bytecodes ...
Dump LX/002; <init> (Landroid/content/Context;)V ... png ... source codes ... bytecodes ...
Dump LX/002; A00 (Ljava/lang/String;)Ljava/io/InputStream; ... png ... bytecodes ...
[ERROR   ] dad.graph: Multiple exit nodes found !
[ERROR   ] dad.graph: Multiple exit nodes found !
```



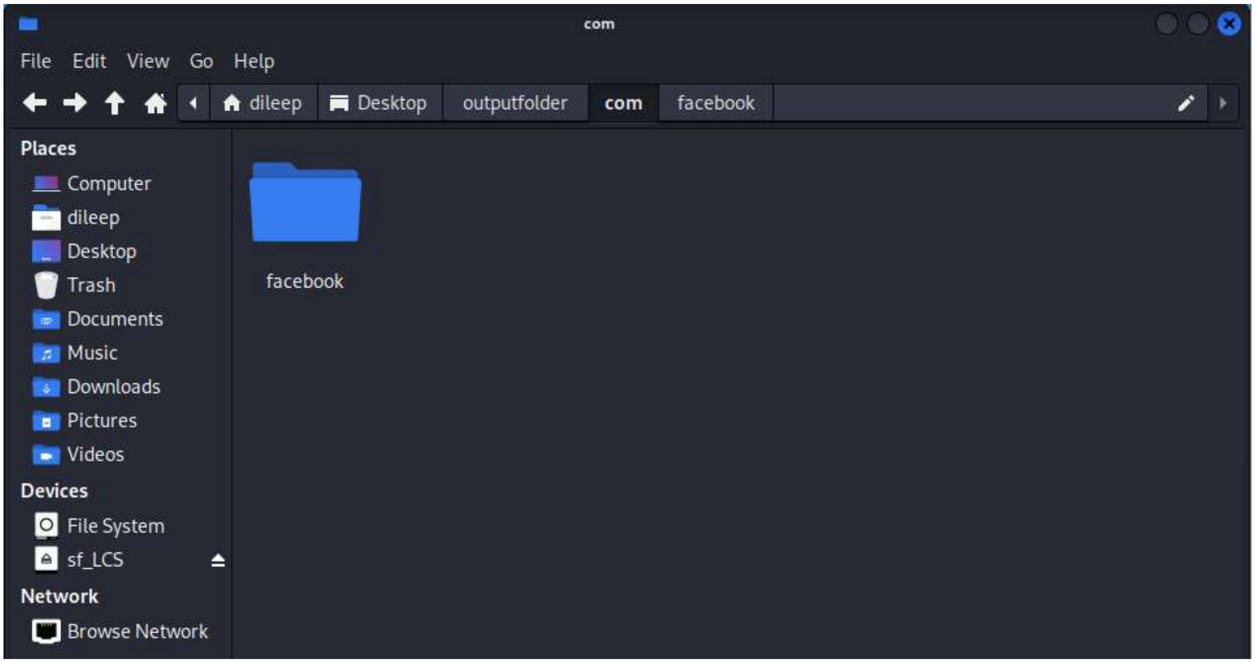
```
Dump Lcom/facebook/systrace/TraceDirect; nativeAsyncTraceBegin (Ljava/lang/String; I J)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeAsyncTraceCancel (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeAsyncTraceEnd (Ljava/lang/String; I J)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeAsyncTraceRename (Ljava/lang/String; Ljava/lang/String; I)V ... png
 ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeAsyncTraceStageBegin (Ljava/lang/String; I J Ljava/lang/String;)V ..
. png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeBeginSection (Ljava/lang/String;)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeBeginSectionWithArgs (Ljava/lang/String; [Ljava/lang/String; I)V ...
 png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeEndAsyncFlow (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeEndSection ()V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeEndSectionWithArgs ([Ljava/lang/String; I)V ... png ... bytecodes ..
.
Dump Lcom/facebook/systrace/TraceDirect; nativeSetEnabledTags (J)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeStartAsyncFlow (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeStepAsyncFlow (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeTraceCounter (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeTraceInstant (Ljava/lang/String; Ljava/lang/String; C)V ... png ...
bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; nativeTraceMetadata (Ljava/lang/String; Ljava/lang/String; I)V ... png ...
 bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; setEnabledTags (J)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; startAsyncFlow (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; stepAsyncFlow (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; traceCounter (Ljava/lang/String; I)V ... png ... bytecodes ...
Dump Lcom/facebook/systrace/TraceDirect; traceInstant (Ljava/lang/String; Ljava/lang/String; C)V ... png ... byteco
des ...
Dump Lcom/facebook/systrace/TraceDirect; traceMetadata (Ljava/lang/String; Ljava/lang/String; I)V ... png ... bytec
odes ...
[ERROR   ] dad.graph: Multiple exit nodes found !
[ERROR   ] dad.graph: Multiple exit nodes found !
Dump Lcom/facebook/xzdecoder/XzInputStream; <clinit> ()V ... png ... source codes ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; <init> (Ljava/io/InputStream;)V ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; decodeMoreBytes ()V ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; decompressStream (J [B I I [B I I)J ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; end (J)V ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; initializeLibrary ()V ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; initializeState ()J ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; readMoreInput ()Z ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; close ()V ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; read ()I ... png ... bytecodes ...
Dump Lcom/facebook/xzdecoder/XzInputStream; read ([B I I)I ... png ... bytecodes ...
```

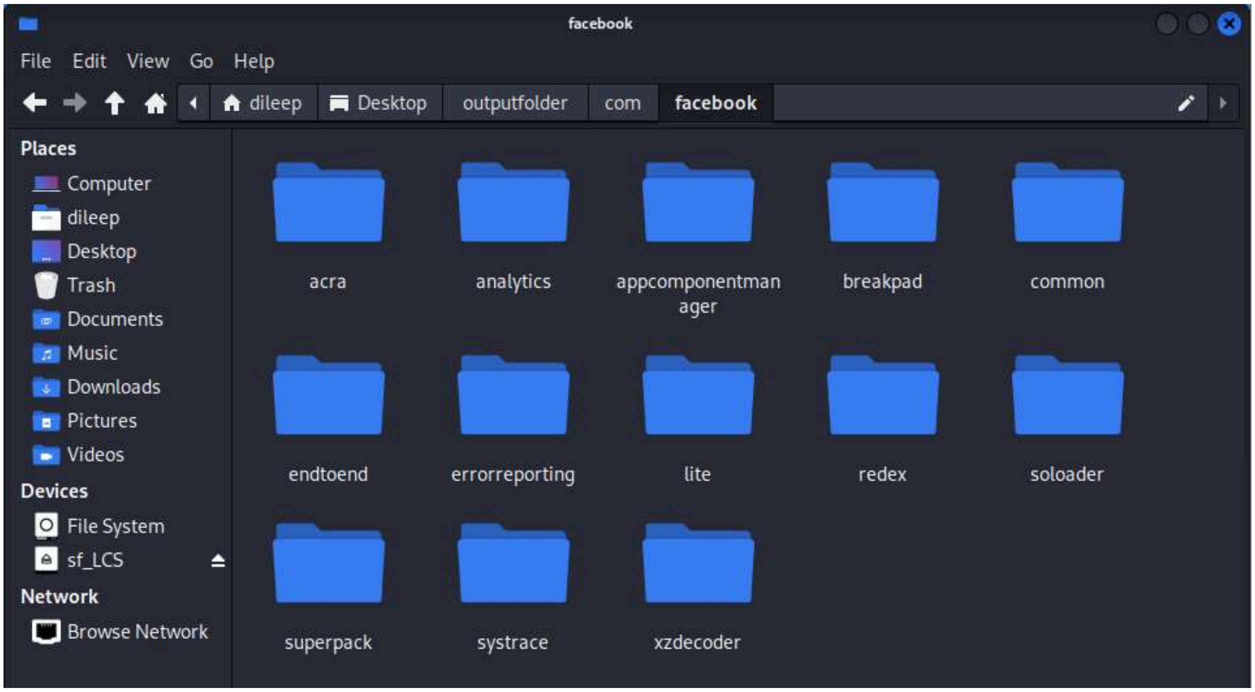
This will decompile the app someapp.apk into the folder outputfolder and limit the processing to all methods, where the classname starts with com.elite..

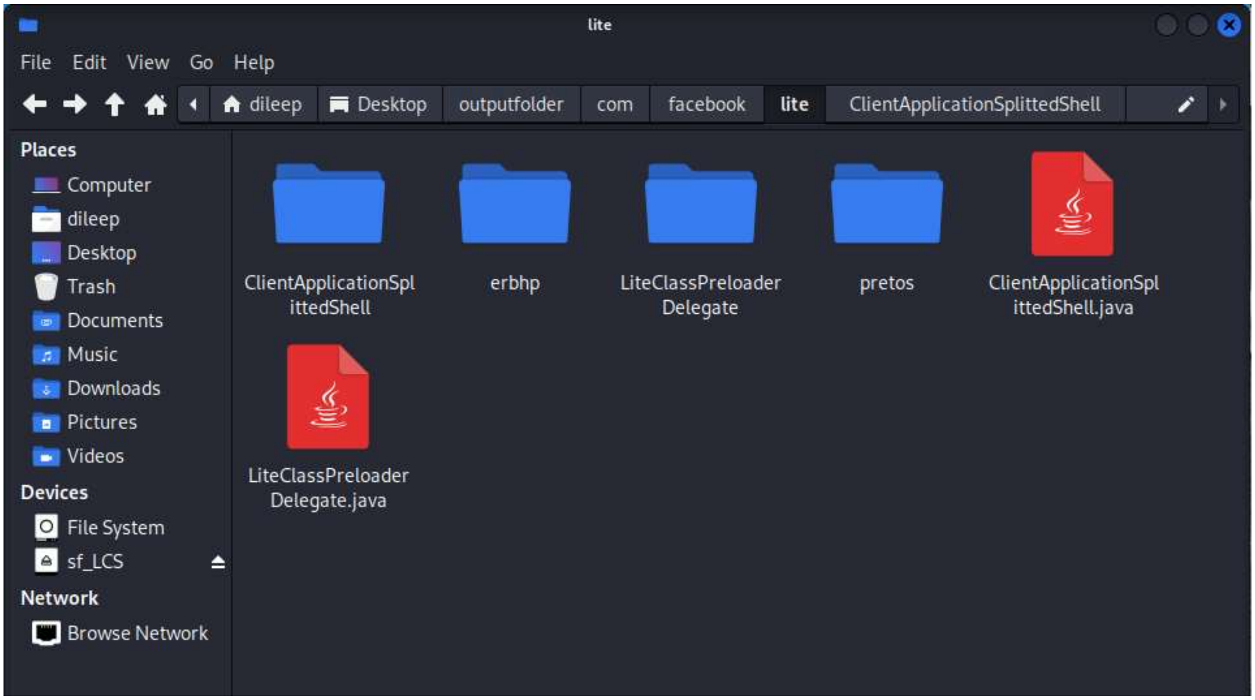
After the decompiling the app, the following folder are created.

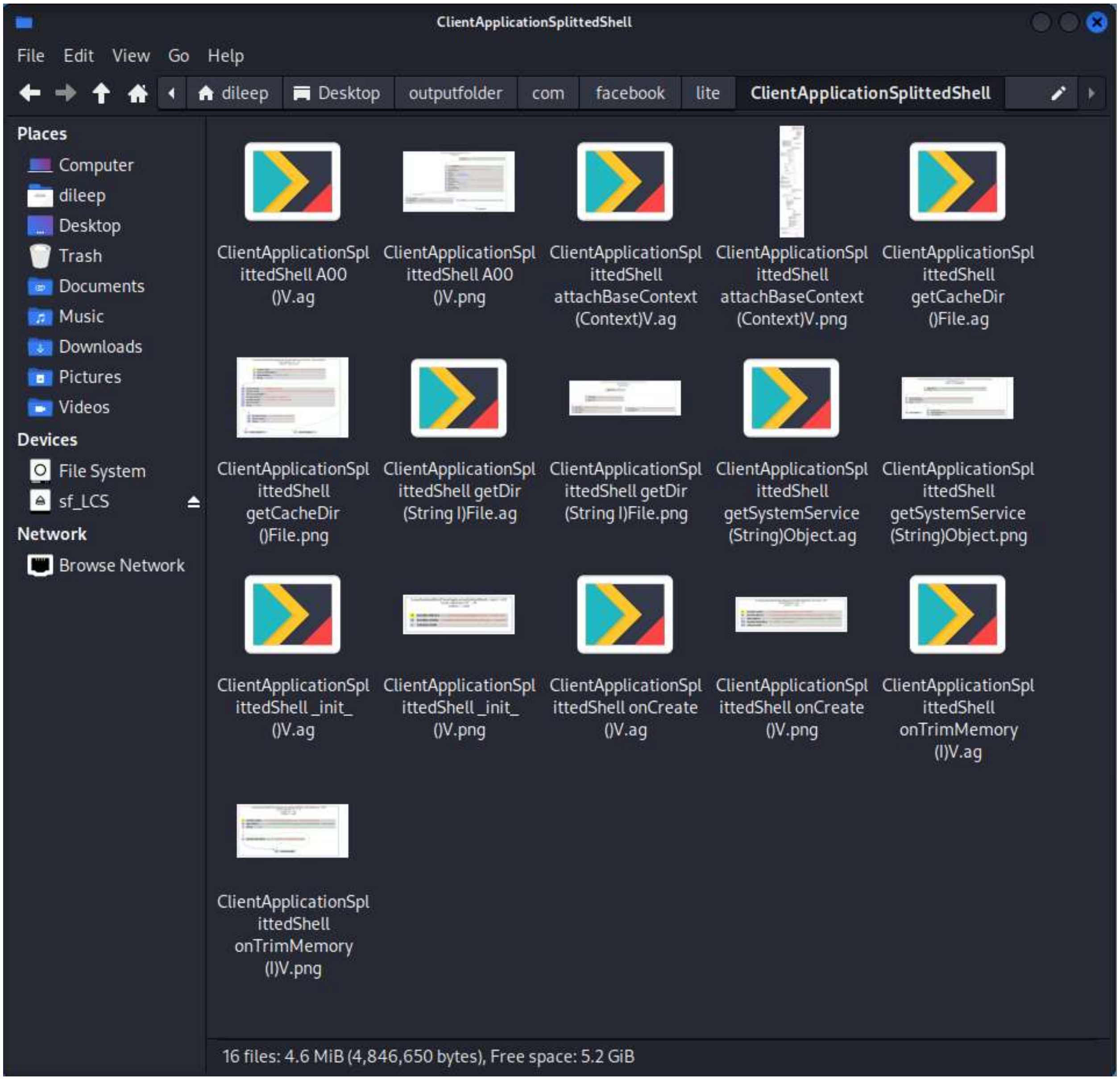


We have the following folders inside the com folder.




facebook
File Edit View Go Help
dileep Desktop outputfolder com facebook
Places
Computer
dileep
Desktop
Trash
Documents
Music
Downloads
Pictures
Videos
Devices
File System
sf_LCS
Network
Browse Network
acra
analytics
appcomponentmanager
breakpad
common
endtoend
errorreporting
lite
redex
soloader
superpack
systrace
xzdecoder


lite
File Edit View Go Help
dileep Desktop outputfolder com facebook lite ClientApplicationSplittedShell
Places
Computer
dileep
Desktop
Trash
Documents
Music
Downloads
Pictures
Videos
Devices
File System
sf_LCS
Network
Browse Network
ClientApplicationSplittedShell
erbhp
LiteClassPreloaderDelegate
pretos
ClientApplicationSplittedShell.java
LiteClassPreloaderDelegate.java



ClientApplicationSplittedShell A00 ()V.ag

ClientApplicationSplittedShell A00 ()V.png

ClientApplicationSplittedShell attachBaseContext (Context)V.ag

ClientApplicationSplittedShell attachBaseContext (Context)V.png

ClientApplicationSplittedShell getCacheDir ()File.ag

ClientApplicationSplittedShell getCacheDir ()File.png

ClientApplicationSplittedShell getDir (String I)File.ag

ClientApplicationSplittedShell getDir (String I)File.png

ClientApplicationSplittedShell getSystemService (String)Object.ag

ClientApplicationSplittedShell getSystemService (String)Object.png

ClientApplicationSplittedShell _init_ ()V.ag

ClientApplicationSplittedShell _init_ ()V.png

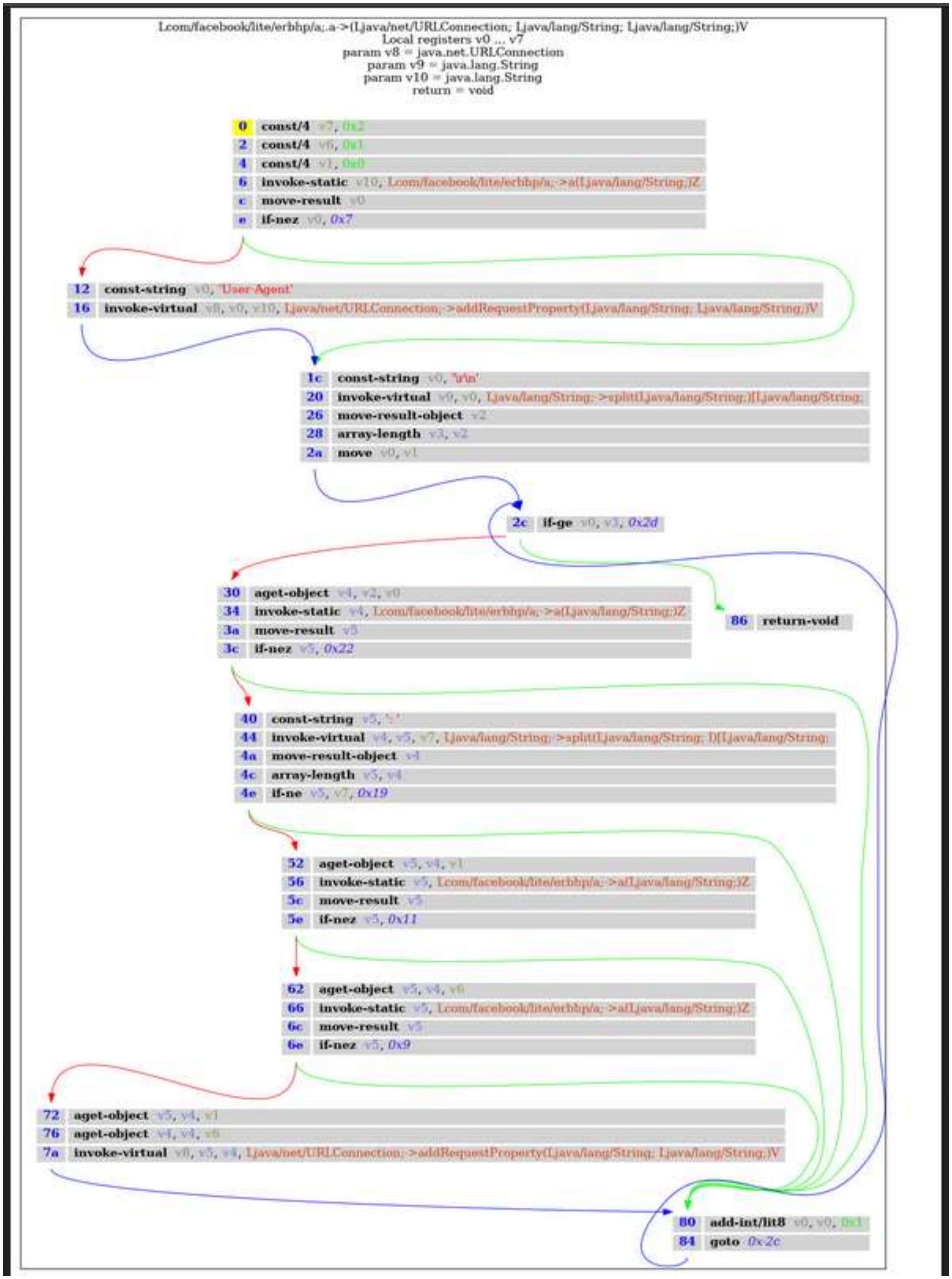
ClientApplicationSplittedShell onCreate ()V.ag

ClientApplicationSplittedShell onCreate ()V.png

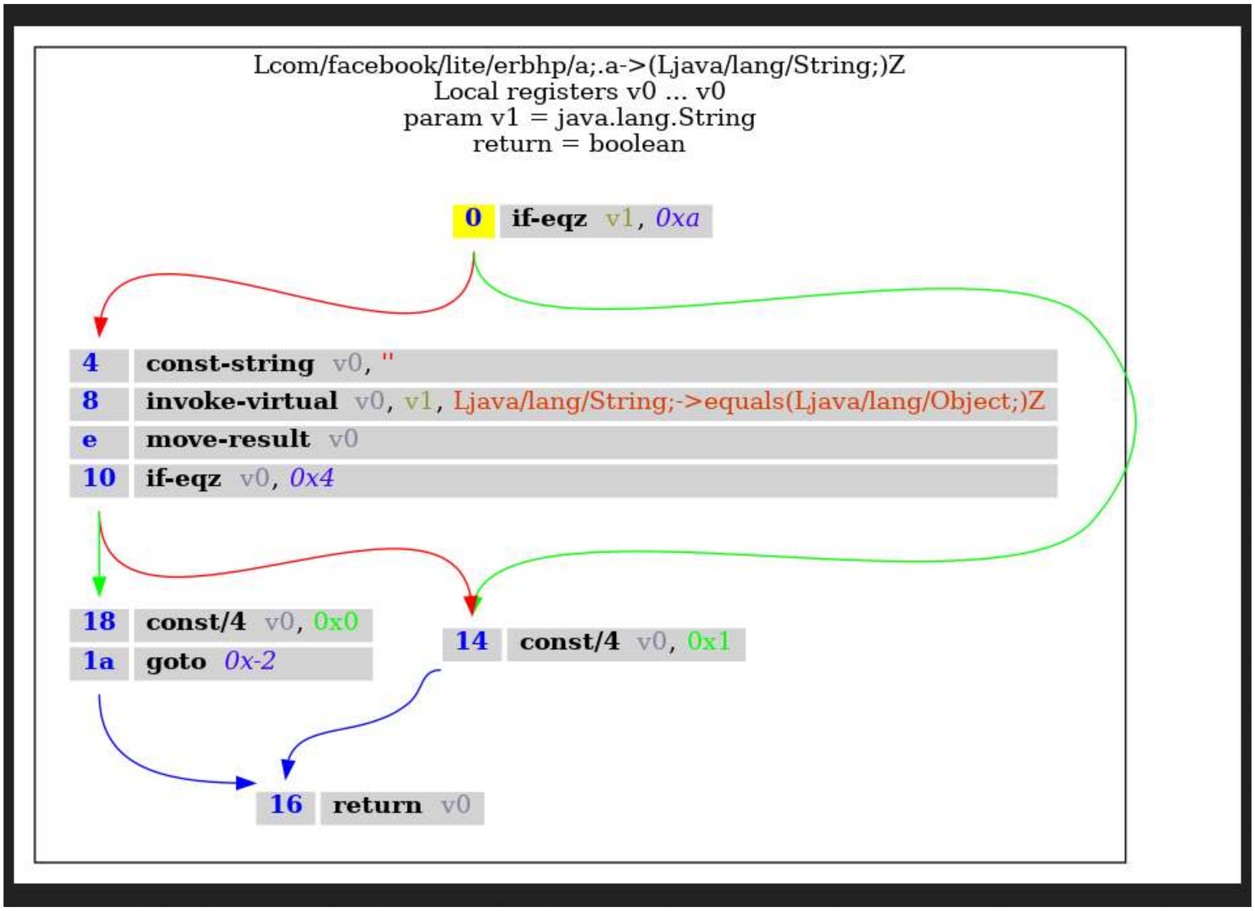
ClientApplicationSplittedShell onTrimMemory (I)V.ag

ClientApplicationSplittedShell onTrimMemory (I)V.png

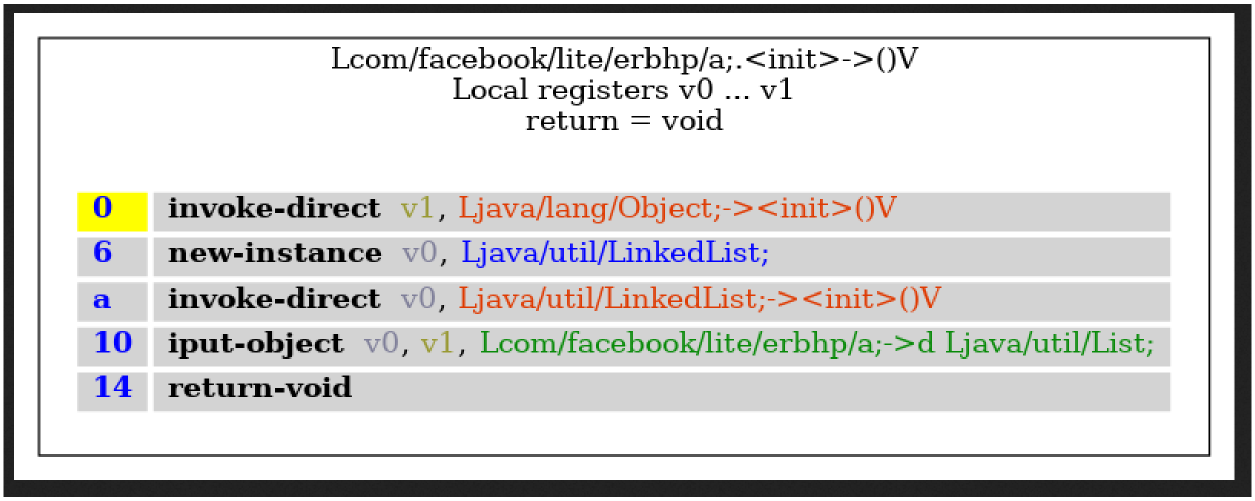
16 files: 4.6 MiB (4,846,650 bytes), Free space: 5.2 GiB



```
Lcom/facebook/lite/erbhp/a;.a->(Ljava/net/URLConnection; Ljava/lang/String; Ljava/lang/String;)V
Local registers v0 ... v7
param v8 = java.net.URLConnection
param v9 = java.lang.String
param v10 = java.lang.String
return = void

0  const/4 v7, 0x2
2  const/4 v6, 0x1
4  const/4 v1, 0x0
6  invoke-static v10, Lcom/facebook/lite/erbhp/a;->a(Ljava/lang/String;)Z
c  move-result v0
e  if-nez v0, 0x7

12 const-string v0, 'User-Agent'
16 invoke-virtual v8, v0, v10, Ljava/net/URLConnection;->addRequestProperty(Ljava/lang/String; Ljava/lang/String;)V

1c const-string v0, '\r\n'
20 invoke-virtual v9, v0, Ljava/lang/String;->split(Ljava/lang/String;)[Ljava/lang/String;
26 move-result-object v2
28 array-length v3, v2
2a move v0, v1

2c if-ge v0, v3, 0x2d

30 aget-object v4, v2, v0
34 invoke-static v4, Lcom/facebook/lite/erbhp/a;->a(Ljava/lang/String;)Z
3a move-result v5
3c if-nez v5, 0x22

86 return-void

40 const-string v5, ': '
44 invoke-virtual v4, v5, v7, Ljava/lang/String;->split(Ljava/lang/String; I)[Ljava/lang/String;
4a move-result-object v4
4c array-length v5, v4
4e if-ne v5, v7, 0x19

52 aget-object v5, v4, v1
56 invoke-static v5, Lcom/facebook/lite/erbhp/a;->a(Ljava/lang/String;)Z
5c move-result v5
5e if-nez v5, 0x11

62 aget-object v5, v4, v6
66 invoke-static v5, Lcom/facebook/lite/erbhp/a;->a(Ljava/lang/String;)Z
6c move-result v5
6e if-nez v5, 0x9

72 aget-object v5, v4, v1
76 aget-object v4, v4, v6
7a invoke-virtual v8, v5, v4, Ljava/net/URLConnection;->addRequestProperty(Ljava/lang/String; Ljava/lang/String;)V

80 add-int/lit8 v0, v0, 0x1
84 goto 0x-2c
```



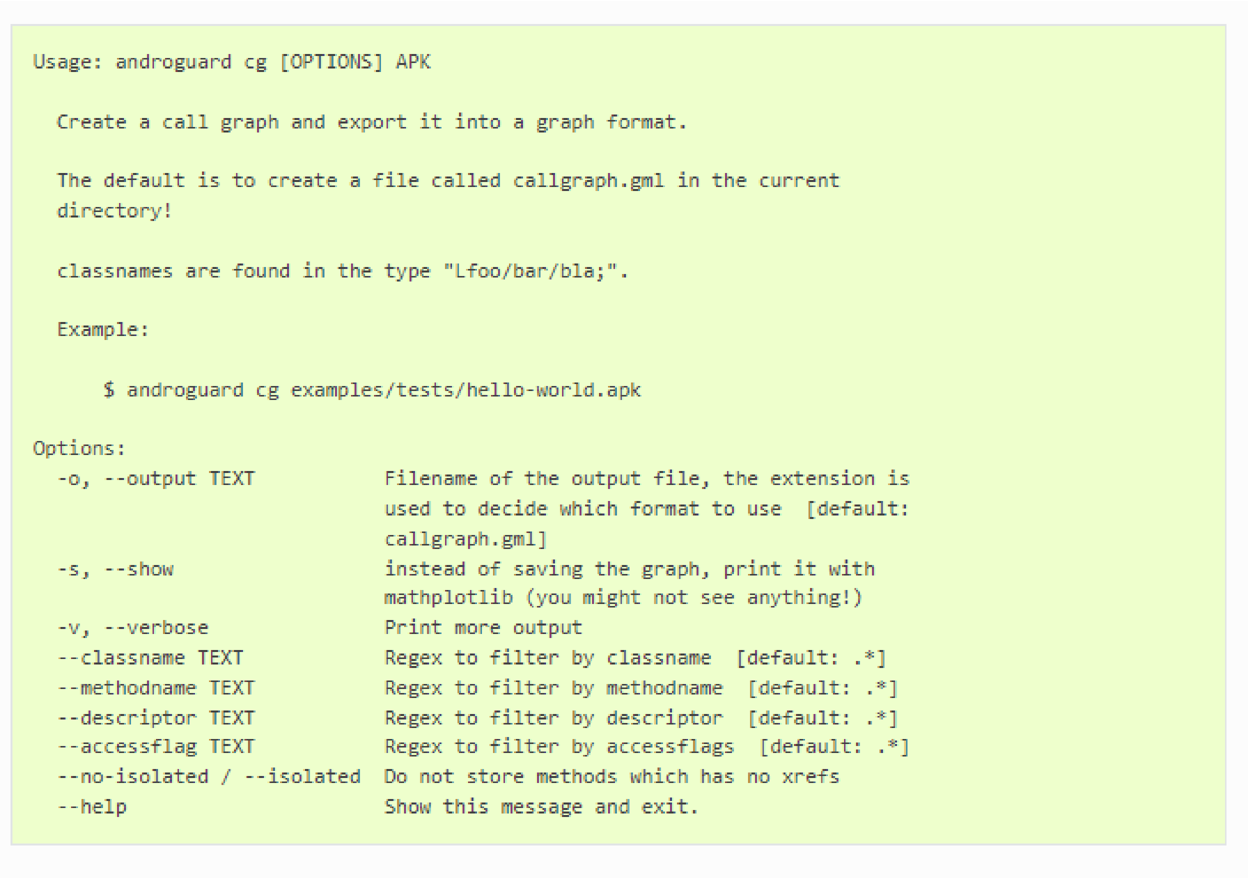
```
Lcom/facebook/lite/erbhp/a;.a->(Ljava/lang/String;)Z
Local registers v0 ... v0
param v1 = java.lang.String
return = boolean

0   if-eqz v1, 0xa

4   const-string v0, ''
8   invoke-virtual v0, v1, Ljava/lang/String;->equals(Ljava/lang/Object;)Z
e   move-result v0
10  if-eqz v0, 0x4

18  const/4 v0, 0x0
1a  goto 0x-2

14  const/4 v0, 0x1

16  return v0
```



```
Lcom/facebook/lite/erbhp/a;.<init>->()V
Local registers v0 ... v1
return = void

0   invoke-direct v1, Ljava/lang/Object;-><init>()V
6   new-instance v0, Ljava/util/LinkedList;
a   invoke-direct v0, Ljava/util/LinkedList;-><init>()V
10  iput-object v0, v1, Lcom/facebook/lite/erbhp/a;->d Ljava/util/List;
14  return-void
```

- Generating call graph

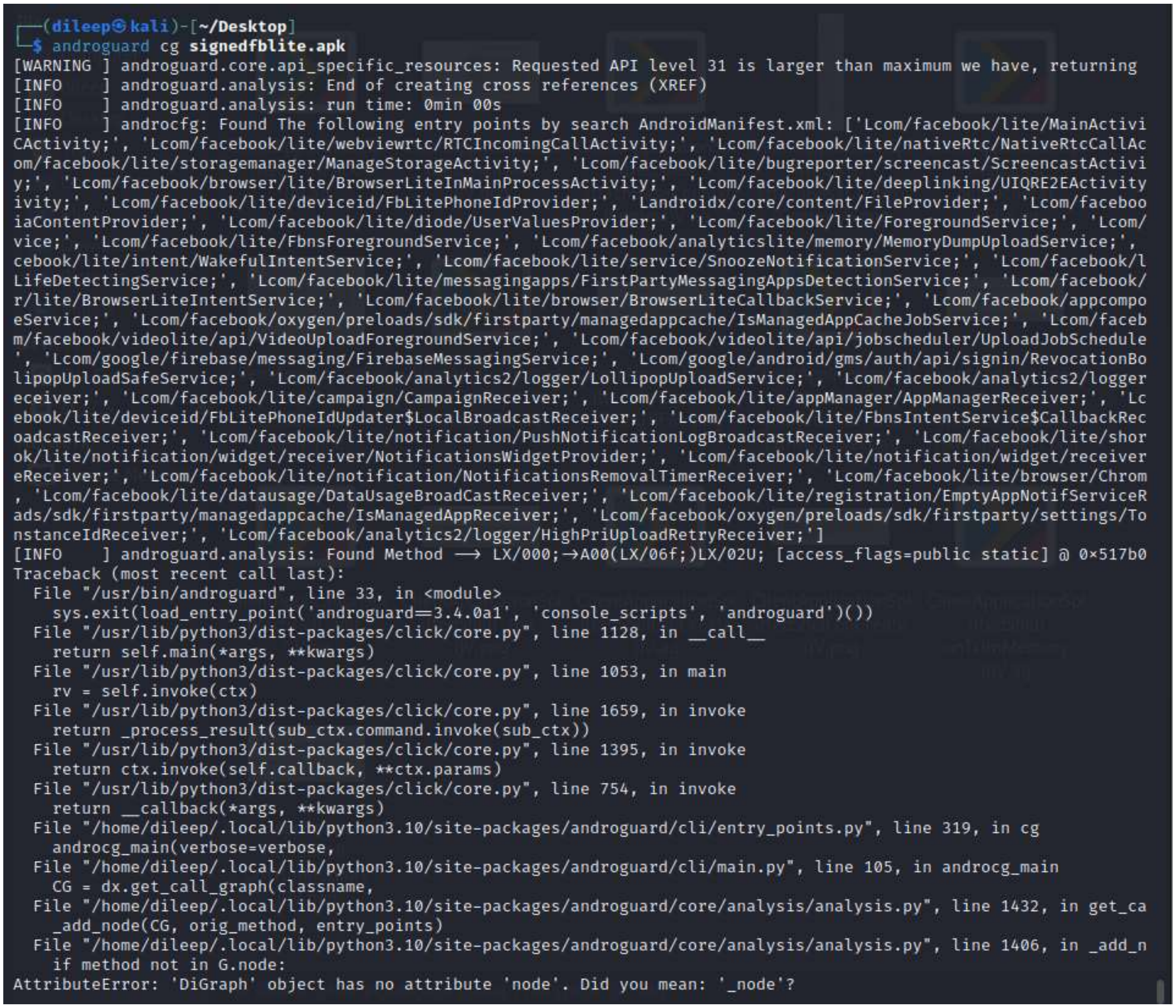
Androguard has the ability to generate call graphs.



```
Usage: androguard cg [OPTIONS] APK

  Create a call graph and export it into a graph format.

  The default is to create a file called callgraph.gml in the current
  directory!

  classnames are found in the type "Lfoo/bar/bla;".

  Example:

      $ androguard cg examples/tests/hello-world.apk

Options:
  -o, --output TEXT           Filename of the output file, the extension is
                              used to decide which format to use  [default:
                              callgraph.gml]
  -s, --show                  instead of saving the graph, print it with
                              mathplotlib (you might not see anything!)
  -v, --verbose               Print more output
  --classname TEXT            Regex to filter by classname  [default: .*]
  --methodname TEXT           Regex to filter by methodname  [default: .*]
  --descriptor TEXT           Regex to filter by descriptor  [default: .*]
  --accessflag TEXT           Regex to filter by accessflags  [default: .*]
  --no-isolated / --isolated  Do not store methods which has no xrefs
  --help                      Show this message and exit.
```

To create call graphs (methods as nodes and edges as calls to methods), we run the following command.

Androguard cg signedfblite.apk

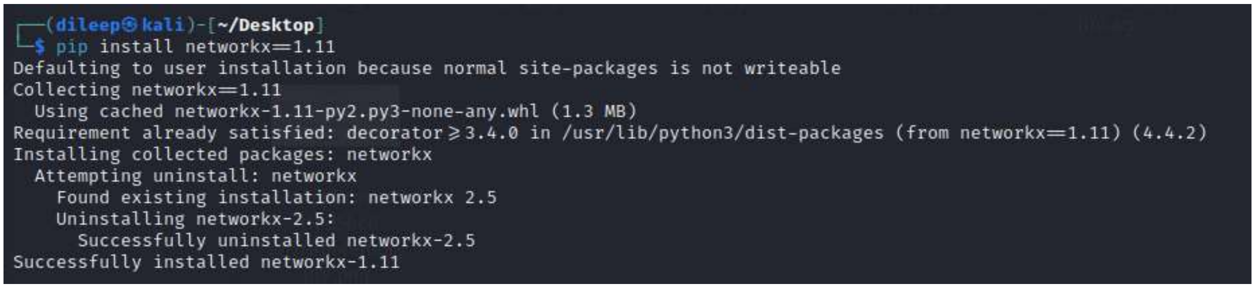


```
┌──(dileep㉿kali)-[~/Desktop]
└─$ androguard cg signedfblite.apk
[WARNING ] androguard.core.api_specific_resources: Requested API level 31 is larger than maximum we have, returning
[INFO    ] androguard.analysis: End of creating cross references (XREF)
[INFO    ] androguard.analysis: run time: 0min 00s
[INFO    ] androcfg: Found The following entry points by search AndroidManifest.xml: ['Lcom/facebook/lite/MainActivi
CActivity;', 'Lcom/facebook/lite/webviewrtc/RTCIncomingCallActivity;', 'Lcom/facebook/lite/nativeRtc/NativeRtcCallAc
om/facebook/lite/storagemanager/ManageStorageActivity;', 'Lcom/facebook/lite/bugreporter/screencast/ScreencastActivi
y;', 'Lcom/facebook/browser/lite/BrowserLiteInMainProcessActivity;', 'Lcom/facebook/lite/deeplinking/UIQRE2EActivity
ivity;', 'Lcom/facebook/lite/deviceid/FbLitePhoneIdProvider;', 'Landroidx/core/content/FileProvider;', 'Lcom/faceboo
iaContentProvider;', 'Lcom/facebook/lite/diode/UserValuesProvider;', 'Lcom/facebook/lite/ForegroundService;', 'Lcom/
vice;', 'Lcom/facebook/lite/FbnsForegroundService;', 'Lcom/facebook/analyticslite/memory/MemoryDumpUploadService;',
cebook/lite/intent/WakefulIntentService;', 'Lcom/facebook/lite/service/SnoozeNotificationService;', 'Lcom/facebook/l
LifeDetectingService;', 'Lcom/facebook/lite/messagingapps/FirstPartyMessagingAppsDetectionService;', 'Lcom/facebook/
r/lite/BrowserLiteIntentService;', 'Lcom/facebook/lite/browser/BrowserLiteCallbackService;', 'Lcom/facebook/appcompo
eService;', 'Lcom/facebook/oxygen/preloads/sdk/firstparty/managedappcache/IsManagedAppCacheJobService;', 'Lcom/faceb
m/facebook/videolite/api/VideoUploadForegroundService;', 'Lcom/facebook/videolite/api/jobscheduler/UploadJobSchedule
', 'Lcom/google/firebase/messaging/FirebaseMessagingService;', 'Lcom/google/android/gms/auth/api/signin/RevocationBo
lipopUploadSafeService;', 'Lcom/facebook/analytics2/logger/LollipopUploadService;', 'Lcom/facebook/analytics2/logger
eceiver;', 'Lcom/facebook/lite/campaign/CampaignReceiver;', 'Lcom/facebook/lite/appManager/AppManagerReceiver;', 'Lc
ebook/lite/deviceid/FbLitePhoneIdUpdater$LocalBroadcastReceiver;', 'Lcom/facebook/lite/FbnsIntentService$CallbackRec
oadcastReceiver;', 'Lcom/facebook/lite/notification/PushNotificationLogBroadcastReceiver;', 'Lcom/facebook/lite/shor
ok/lite/notification/widget/receiver/NotificationsWidgetProvider;', 'Lcom/facebook/lite/notification/widget/receiver
eReceiver;', 'Lcom/facebook/lite/notification/NotificationsRemovalTimerReceiver;', 'Lcom/facebook/lite/browser/Chrom
, 'Lcom/facebook/lite/datausage/DataUsageBroadCastReceiver;', 'Lcom/facebook/lite/registration/EmptyAppNotifServiceR
ads/sdk/firstparty/managedappcache/IsManagedAppReceiver;', 'Lcom/facebook/oxygen/preloads/sdk/firstparty/settings/To
nstanceIdReceiver;', 'Lcom/facebook/analytics2/logger/HighPriUploadRetryReceiver;']
[INFO    ] androguard.analysis: Found Method ⟶ LX/000;→A00(LX/06f;)LX/02U; [access_flags=public static] @ 0×517b0
Traceback (most recent call last):
  File "/usr/bin/androguard", line 33, in <module>
    sys.exit(load_entry_point('androguard==3.4.0a1', 'console_scripts', 'androguard')())
  File "/usr/lib/python3/dist-packages/click/core.py", line 1128, in __call__
    return self.main(*args, **kwargs)
  File "/usr/lib/python3/dist-packages/click/core.py", line 1053, in main
    rv = self.invoke(ctx)
  File "/usr/lib/python3/dist-packages/click/core.py", line 1659, in invoke
    return _process_result(sub_ctx.command.invoke(sub_ctx))
  File "/usr/lib/python3/dist-packages/click/core.py", line 1395, in invoke
    return ctx.invoke(self.callback, **ctx.params)
  File "/usr/lib/python3/dist-packages/click/core.py", line 754, in invoke
    return __callback(*args, **kwargs)
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/cli/entry_points.py", line 319, in cg
    androcg_main(verbose=verbose,
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/cli/main.py", line 105, in androcg_main
    CG = dx.get_call_graph(classname,
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/core/analysis/analysis.py", line 1432, in get_ca
    _add_node(CG, orig_method, entry_points)
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/core/analysis/analysis.py", line 1406, in _add_n
    if method not in G.node:
AttributeError: 'DiGraph' object has no attribute 'node'. Did you mean: '_node'?
```

While trying to generate the call graph, got the error saying,

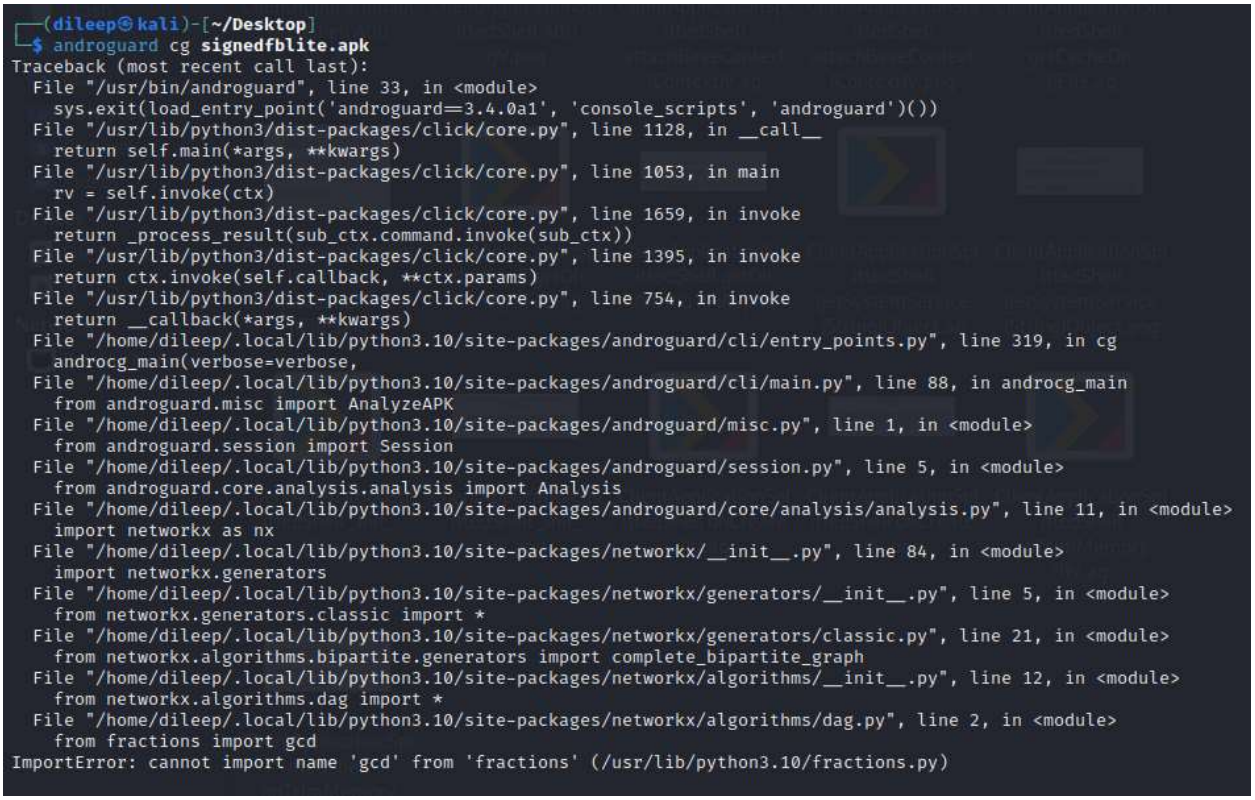
AttributeError: ‘Digraph’ object has no attribute ‘node’. Did you mean: ‘_node’?

Tried to fix the error by changing the networkx version.



```
┌──(dileep㉿kali)-[~/Desktop]
└─$ pip install networkx==1.11
Defaulting to user installation because normal site-packages is not writeable
Collecting networkx==1.11
  Using cached networkx-1.11-py2.py3-none-any.whl (1.3 MB)
Requirement already satisfied: decorator>=3.4.0 in /usr/lib/python3/dist-packages (from networkx==1.11) (4.4.2)
Installing collected packages: networkx
  Attempting uninstall: networkx
    Found existing installation: networkx 2.5
    Uninstalling networkx-2.5:
      Successfully uninstalled networkx-2.5
Successfully installed networkx-1.11
```

After that tried generating call graph again, but I didn’t work this time around.



```
┌──(dileep㉿kali)-[~/Desktop]
└─$ androguard cg signedfblite.apk
Traceback (most recent call last):
  File "/usr/bin/androguard", line 33, in <module>
    sys.exit(load_entry_point('androguard==3.4.0a1', 'console_scripts', 'androguard')())
  File "/usr/lib/python3/dist-packages/click/core.py", line 1128, in __call__
    return self.main(*args, **kwargs)
  File "/usr/lib/python3/dist-packages/click/core.py", line 1053, in main
    rv = self.invoke(ctx)
  File "/usr/lib/python3/dist-packages/click/core.py", line 1659, in invoke
    return _process_result(sub_ctx.command.invoke(sub_ctx))
  File "/usr/lib/python3/dist-packages/click/core.py", line 1395, in invoke
    return ctx.invoke(self.callback, **ctx.params)
  File "/usr/lib/python3/dist-packages/click/core.py", line 754, in invoke
    return __callback(*args, **kwargs)
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/cli/entry_points.py", line 319, in cg
    androcg_main(verbose=verbose,
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/cli/main.py", line 88, in androcg_main
    from androguard.misc import AnalyzeAPK
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/misc.py", line 1, in <module>
    from androguard.session import Session
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/session.py", line 5, in <module>
    from androguard.core.analysis.analysis import Analysis
  File "/home/dileep/.local/lib/python3.10/site-packages/androguard/core/analysis/analysis.py", line 11, in <module>
    import networkx as nx
  File "/home/dileep/.local/lib/python3.10/site-packages/networkx/__init__.py", line 84, in <module>
    import networkx.generators
  File "/home/dileep/.local/lib/python3.10/site-packages/networkx/generators/__init__.py", line 5, in <module>
    from networkx.generators.classic import *
  File "/home/dileep/.local/lib/python3.10/site-packages/networkx/generators/classic.py", line 21, in <module>
    from networkx.algorithms.bipartite.generators import complete_bipartite_graph
  File "/home/dileep/.local/lib/python3.10/site-packages/networkx/algorithms/__init__.py", line 12, in <module>
    from networkx.algorithms.dag import *
  File "/home/dileep/.local/lib/python3.10/site-packages/networkx/algorithms/dag.py", line 2, in <module>
    from fractions import gcd
ImportError: cannot import name 'gcd' from 'fractions' (/usr/lib/python3.10/fractions.py)
```

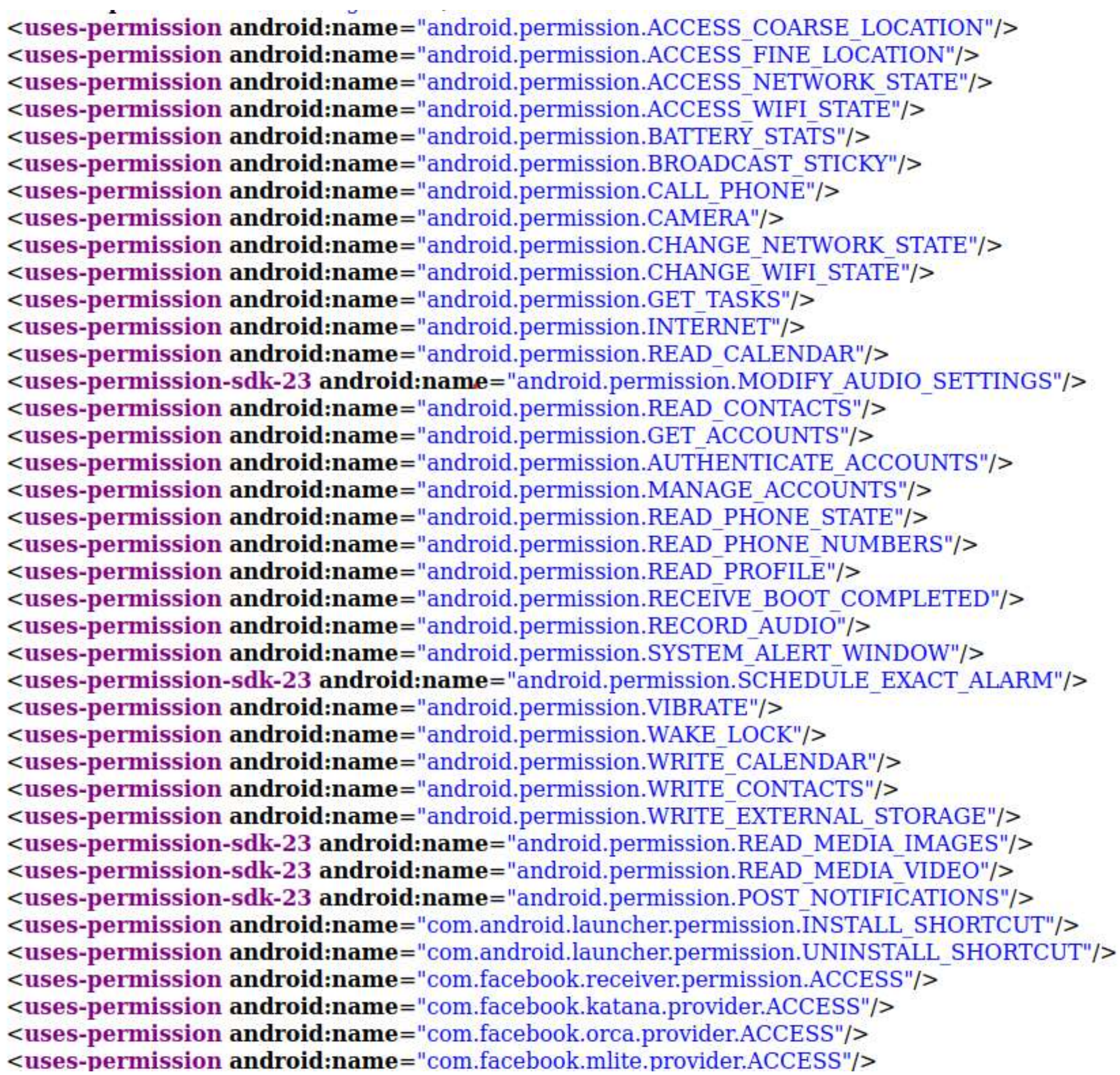
Another error has occurred saying:

ImportError: cannot import name ‘gcd’ from ‘fractions’.

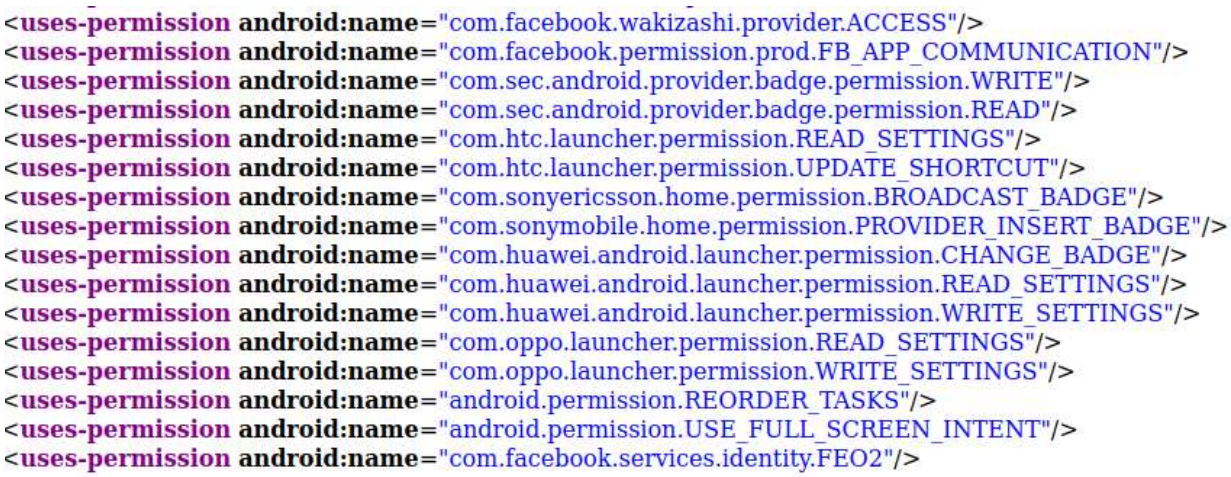
Tried hard but couldn’t fix that error. So, I stopped generating call graph.

Code Analysis

The most important place to start an investigation is the AndroidManifest.xml file. This file is always packed with APK's, and it is where permissions are declared, among other things. For this analysis, I am using Jadx, an open-source tool for decompiling Java Code.



```
<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_NETWORK_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.BATTERY_STATS"/>
<uses-permission android:name="android.permission.BROADCAST_STICKY"/>
<uses-permission android:name="android.permission.CALL_PHONE"/>
<uses-permission android:name="android.permission.CAMERA"/>
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
<uses-permission android:name="android.permission.GET_TASKS"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.READ_CALENDAR"/>
<uses-permission-sdk-23 android:name="android.permission.MODIFY_AUDIO_SETTINGS"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.GET_ACCOUNTS"/>
<uses-permission android:name="android.permission.AUTHENTICATE_ACCOUNTS"/>
<uses-permission android:name="android.permission.MANAGE_ACCOUNTS"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_PHONE_NUMBERS"/>
<uses-permission android:name="android.permission.READ_PROFILE"/>
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW"/>
<uses-permission-sdk-23 android:name="android.permission.SCHEDULE_EXACT_ALARM"/>
<uses-permission android:name="android.permission.VIBRATE"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="android.permission.WRITE_CALENDAR"/>
<uses-permission android:name="android.permission.WRITE_CONTACTS"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission-sdk-23 android:name="android.permission.READ_MEDIA_IMAGES"/>
<uses-permission-sdk-23 android:name="android.permission.READ_MEDIA_VIDEO"/>
<uses-permission-sdk-23 android:name="android.permission.POST_NOTIFICATIONS"/>
<uses-permission android:name="com.android.launcher.permission.INSTALL_SHORTCUT"/>
<uses-permission android:name="com.android.launcher.permission.UNINSTALL_SHORTCUT"/>
<uses-permission android:name="com.facebook.receiver.permission.ACCESS"/>
<uses-permission android:name="com.facebook.katana.provider.ACCESS"/>
<uses-permission android:name="com.facebook.orca.provider.ACCESS"/>
<uses-permission android:name="com.facebook.mlite.provider.ACCESS"/>
```



```
<uses-permission android:name="com.facebook.wakizashi.provider.ACCESS"/>
<uses-permission android:name="com.facebook.permission.prod.FB_APP_COMMUNICATION"/>
<uses-permission android:name="com.sec.android.provider.badge.permission.WRITE"/>
<uses-permission android:name="com.sec.android.provider.badge.permission.READ"/>
<uses-permission android:name="com.htc.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.htc.launcher.permission.UPDATE_SHORTCUT"/>
<uses-permission android:name="com.sonyericsson.home.permission.BROADCAST_BADGE"/>
<uses-permission android:name="com.sonymobile.home.permission.PROVIDER_INSERT_BADGE"/>
<uses-permission android:name="com.huawei.android.launcher.permission.CHANGE_BADGE"/>
<uses-permission android:name="com.huawei.android.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.huawei.android.launcher.permission.WRITE_SETTINGS"/>
<uses-permission android:name="com.oppo.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.oppo.launcher.permission.WRITE_SETTINGS"/>
<uses-permission android:name="android.permission.REORDER_TASKS"/>
<uses-permission android:name="android.permission.USE_FULL_SCREEN_INTENT"/>
<uses-permission android:name="com.facebook.services.identity.FEO2"/>
```

By investigating the permissions, there are some dangerous permissions.

Potentially dangerous permissions used by the app.

WRITE_EXTERNAL_STORAGE

READ_CALENDER

READ_PHONE_STATE

CALL_PHONE

READ_CONTACTS

RECORD_AUTO

WRITE_CALENDER

CAMERA

WRITE_CONTACTS

ACCESS_COARSE_LOCATION

READ_PHONE_NUMBERS

GET_ACCOUNTS

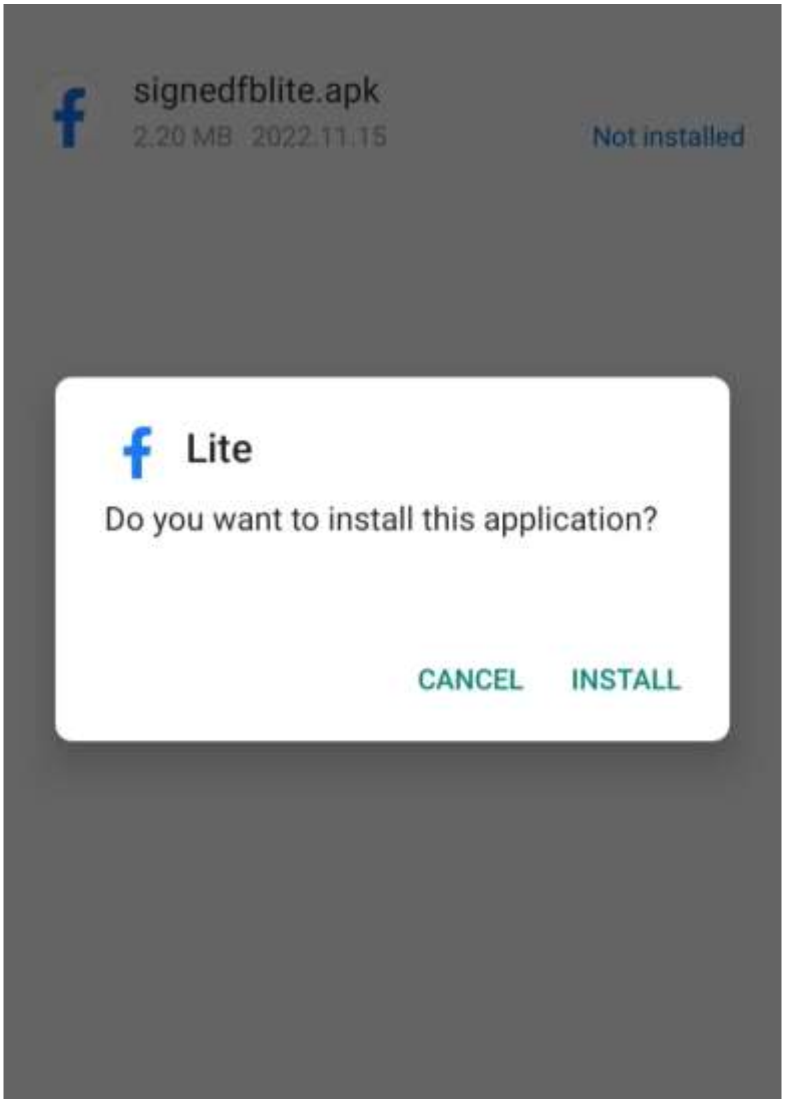
After investigating the source code, the app is not looking suspicious despite some potential dangerous permissions used by the app.

Dynamic Analysis

signedfblite.apk

2.20 MB 2022.11.15

Not installed



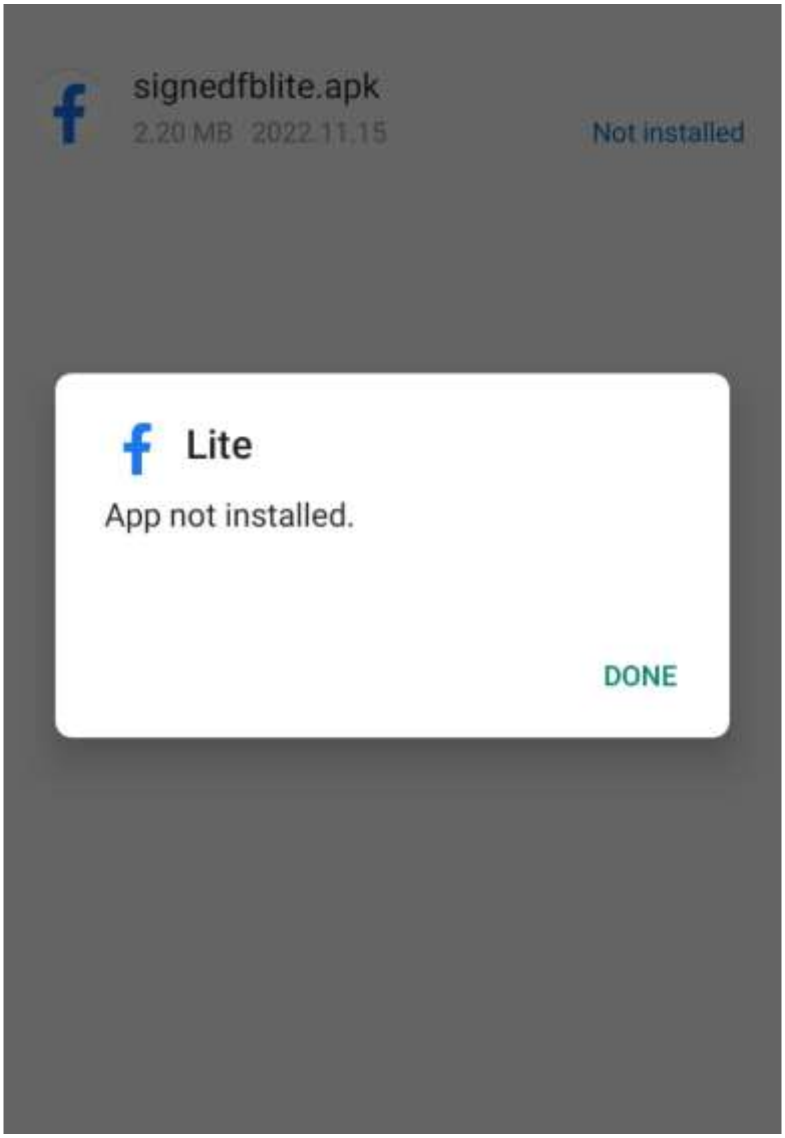
signedfblite.apk

2.20 MB 2022.11.15

Not installed

Lite

Do you want to install this application?



signedfblite.apk
2.20 MB 2022.11.15
Not installed

Lite

App not installed.

In dynamic analysis, we are not even able to install the given apk.

Secret Code:

There’s no secret code in the given apk file. We have reviewed the apk file carefully but we couldn’t find any secret code in any of the file.

Final Verdict:

Inconclusive. The code doesn’t contain a lot of highly-suspicious activity and most most of it seems to be dead code, and dynamic analysis isn’t possible as we are not able to install the app. Overall, I wouldn’t trust this app in the slightest.

Conclusion:

I started my malware analysis by learning about Windows malware, so I was surprised to find that Android malware analysis is much easier. You have the advantage of reading decompiled Java code instead of looking at x86/x64 Intel disassembly with a debugger and disassembler. This is much more readable. Additionally, the required AndroidManifest.xml file present in all Android apps provides a great guide to where the analysis should begin. I also personally find it a bit faster and easier to manage an Android emulator than running a Windows VM for dynamic analysis.