星舆实验室

ICVSec 汽车信息安全

汽车信息安全践行者

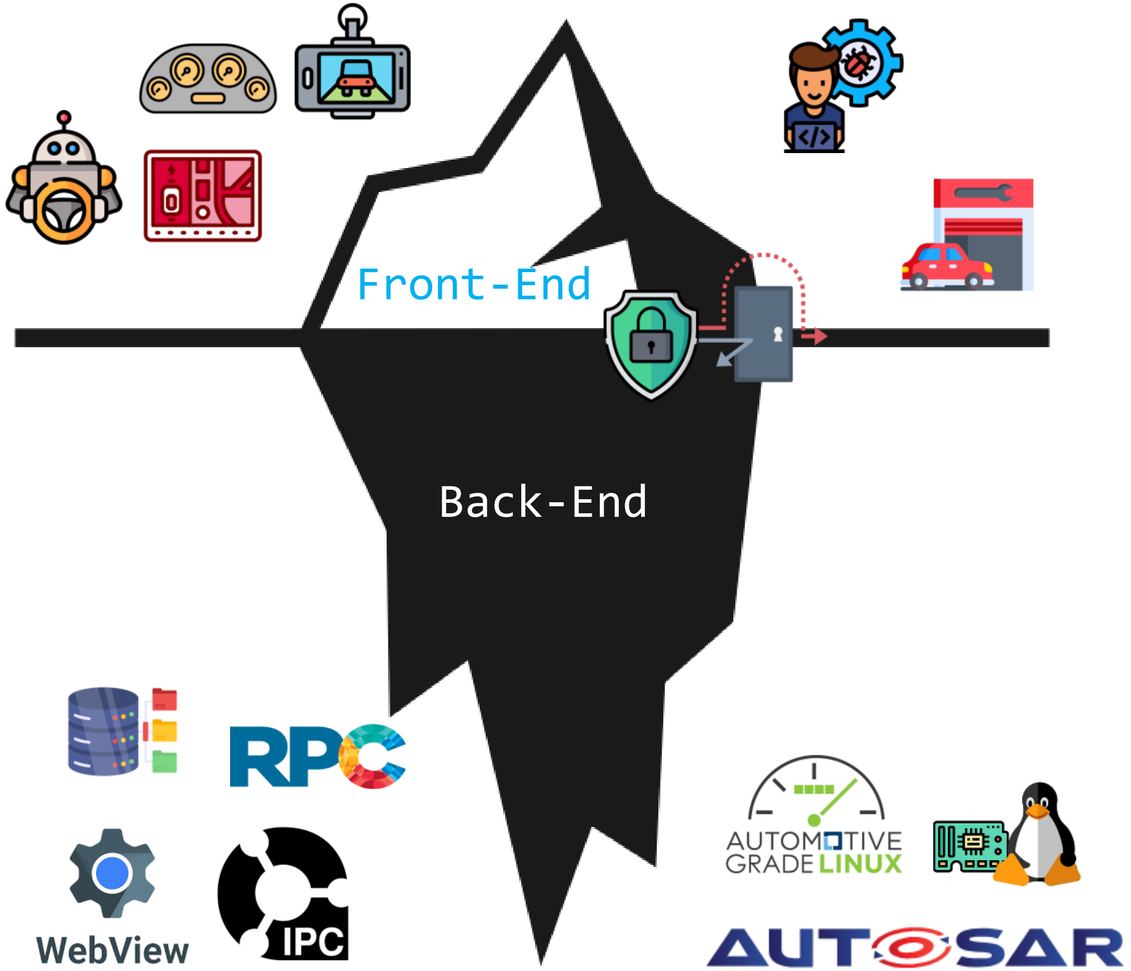
Unlocking Automotive Secrets

Strategies and Tool for accessing hidden services

FF-BY-ONE 2024

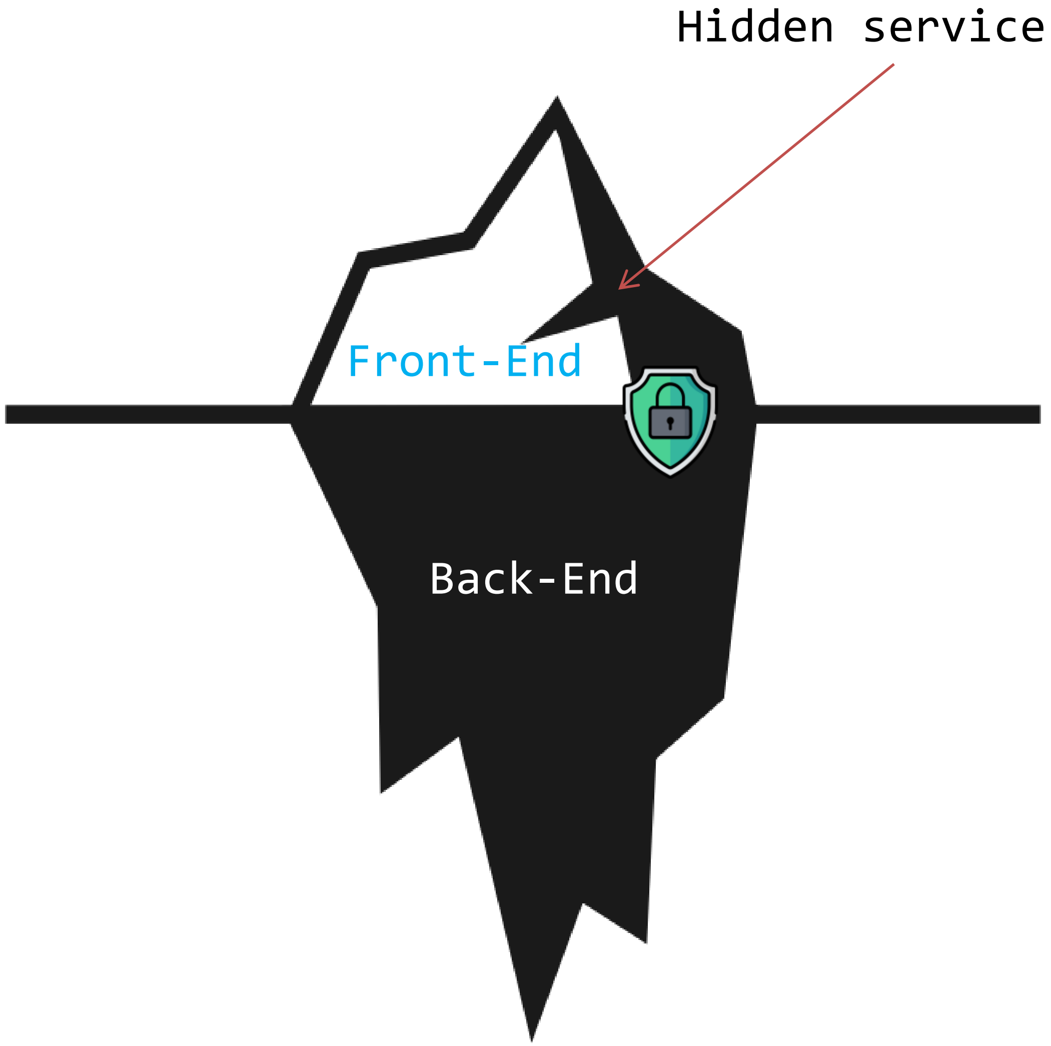
@Delikely

What’s hidden service



Hidden services

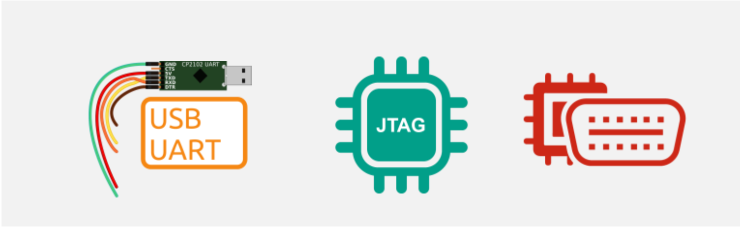
Entrance



Desktop Environment

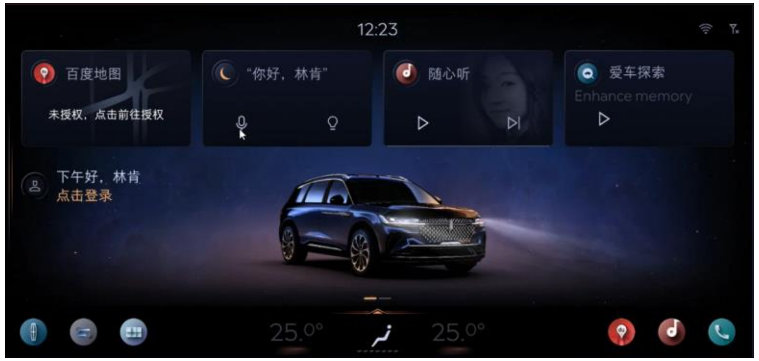
Debug Interface

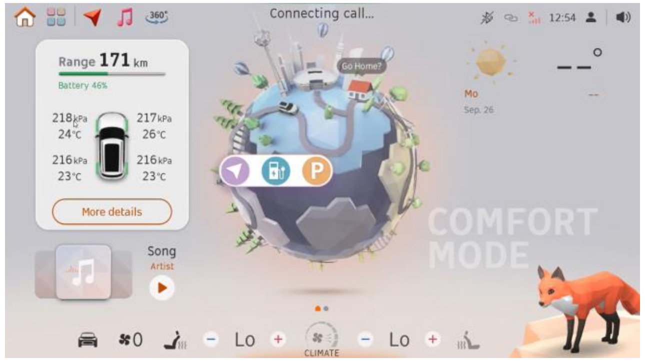
Hardware Interface

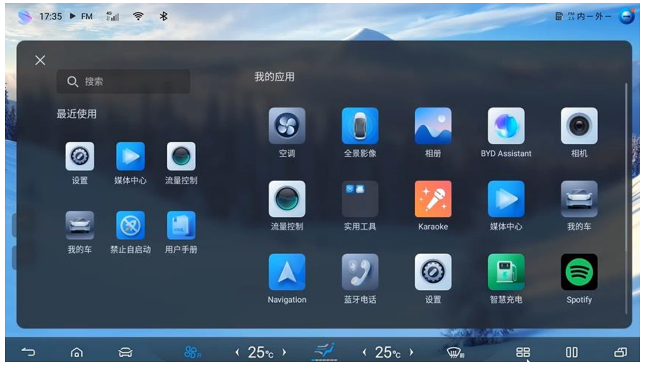


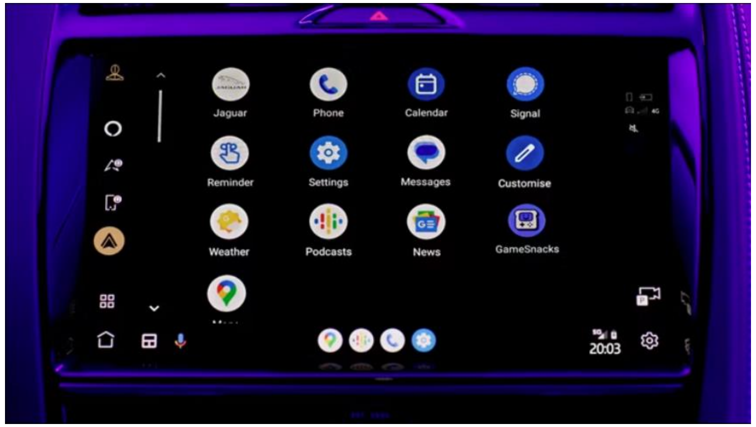
Hidden services

Car Launcher



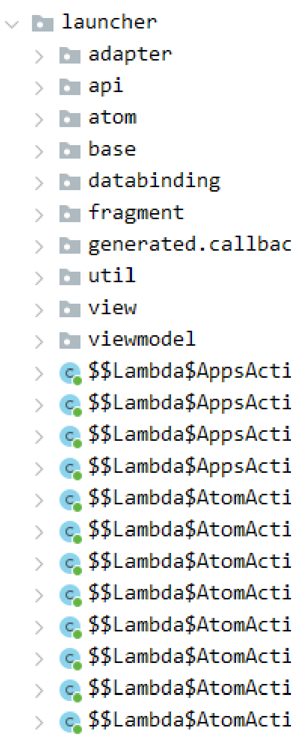




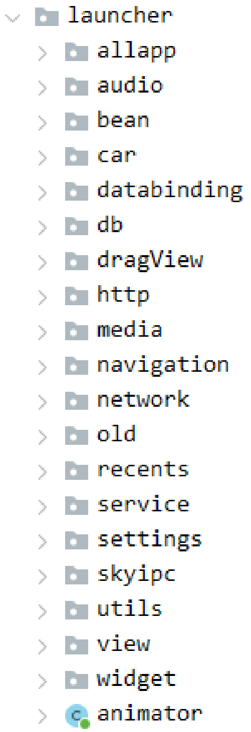


Hidden services

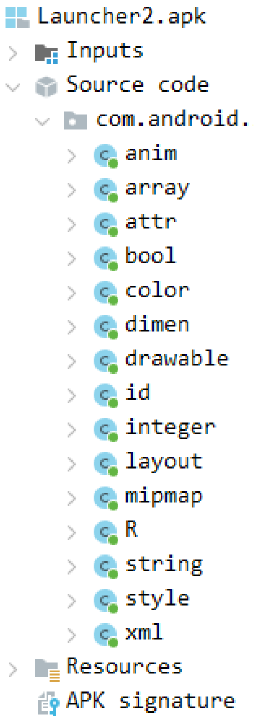
Car Launcher



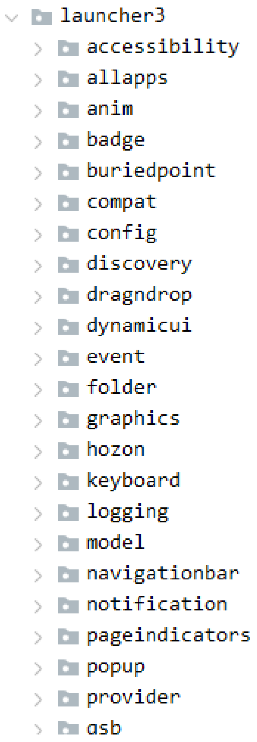
```
∨ launcher
  > adapter
  > api
  > atom
  > base
  > databinding
  > fragment
  > generated.callbac
  > util
  > view
  > viewmodel
  > $$Lambda$AppsActi
  > $$Lambda$AppsActi
  > $$Lambda$AppsActi
  > $$Lambda$AppsActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
  > $$Lambda$AtomActi
```



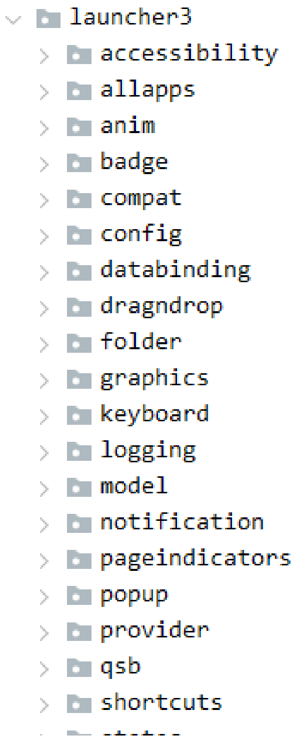
```
∨ launcher
  > allapp
  > audio
  > bean
  > car
  > databinding
  > db
  > dragView
  > http
  > media
  > navigation
  > network
  > old
  > recents
  > service
  > settings
  > skyipc
  > utils
  > view
  > widget
  > animator
```



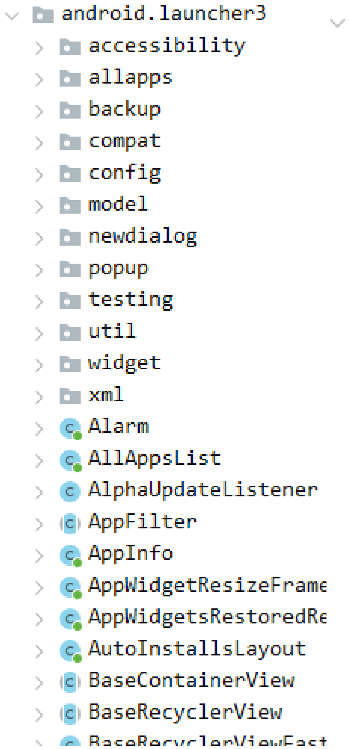
```
Launcher2.apk
> Inputs
∨ Source code
  ∨ com.android.
    > anim
    > array
    > attr
    > bool
    > color
    > dimen
    > drawable
    > id
    > integer
    > layout
    > mipmap
    > R
    > string
    > style
    > xml
> Resources
  APK signature
```



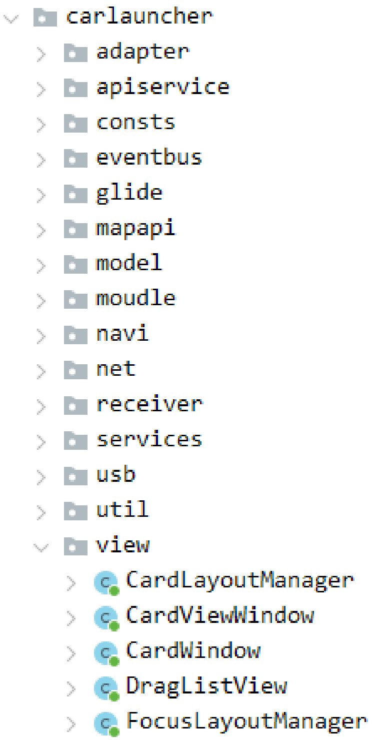
```
∨ launcher3
  > accessibility
  > allapps
  > anim
  > badge
  > buriedpoint
  > compat
  > config
  > discovery
  > dragndrop
  > dynamicui
  > event
  > folder
  > graphics
  > hozon
  > keyboard
  > logging
  > model
  > navigationbar
  > notification
  > pageindicators
  > popup
  > provider
  > qsb
```



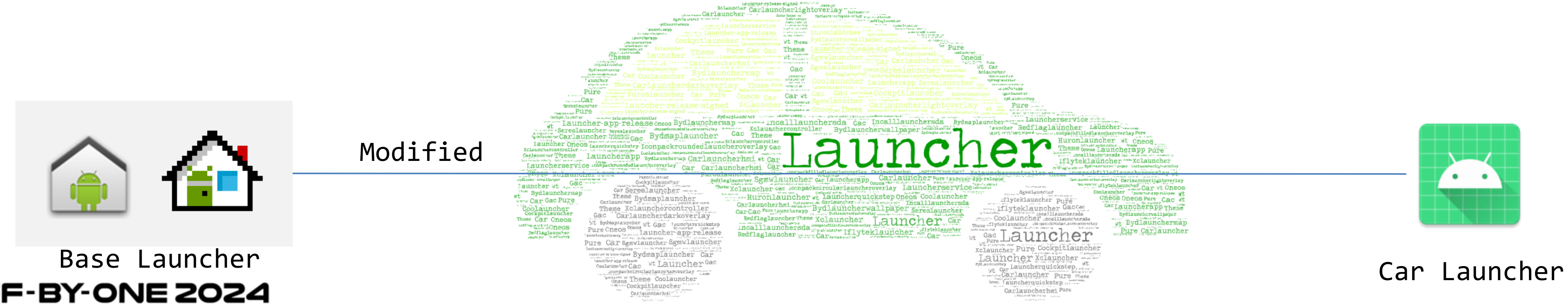
```
∨ launcher3
  > accessibility
  > allapps
  > anim
  > badge
  > compat
  > config
  > databinding
  > dragndrop
  > folder
  > graphics
  > keyboard
  > logging
  > model
  > notification
  > pageindicators
  > popup
  > provider
  > qsb
  > shortcuts
```



```
∨ android.launcher3
  > accessibility
  > allapps
  > backup
  > compat
  > config
  > model
  > newdialog
  > popup
  > testing
  > util
  > widget
  > xml
  > Alarm
  > AllAppsList
  > AlphaUpdateListener
  > AppFilter
  > AppInfo
  > AppWidgetResizeFrame
  > AppWidgetsRestoredRe
  > AutoInstallsLayout
  > BaseContainerView
  > BaseRecyclerView
  > BaseRecyclerViewFast
```

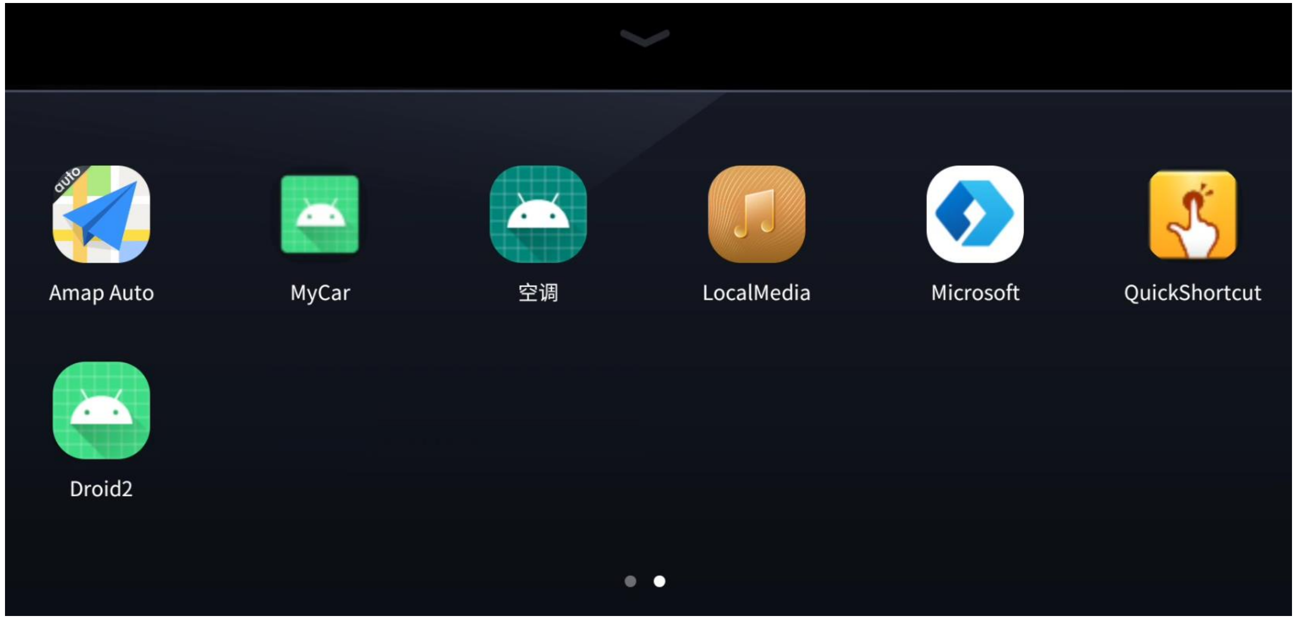


```
∨ carlauncher
  > adapter
  > apiservice
  > consts
  > eventbus
  > glide
  > mapapi
  > model
  > moudle
  > navi
  > net
  > receiver
  > services
  > usb
  > util
  ∨ view
    > CardLayoutManager
    > CardViewWindow
    > CardWindow
    > DragListView
    > FocusLayoutManager
```

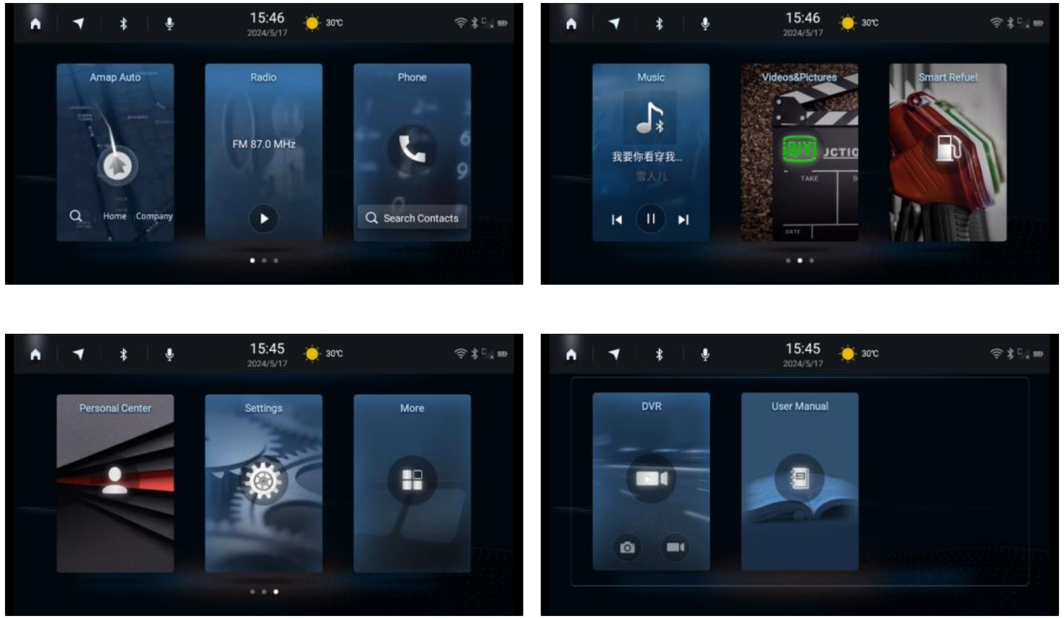


Hidden services

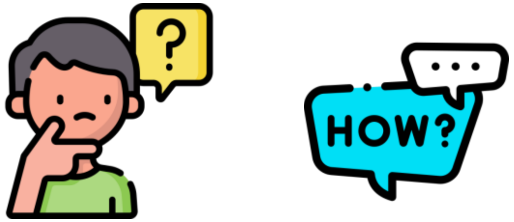
Launcher Features



Third-party apps show on launcher



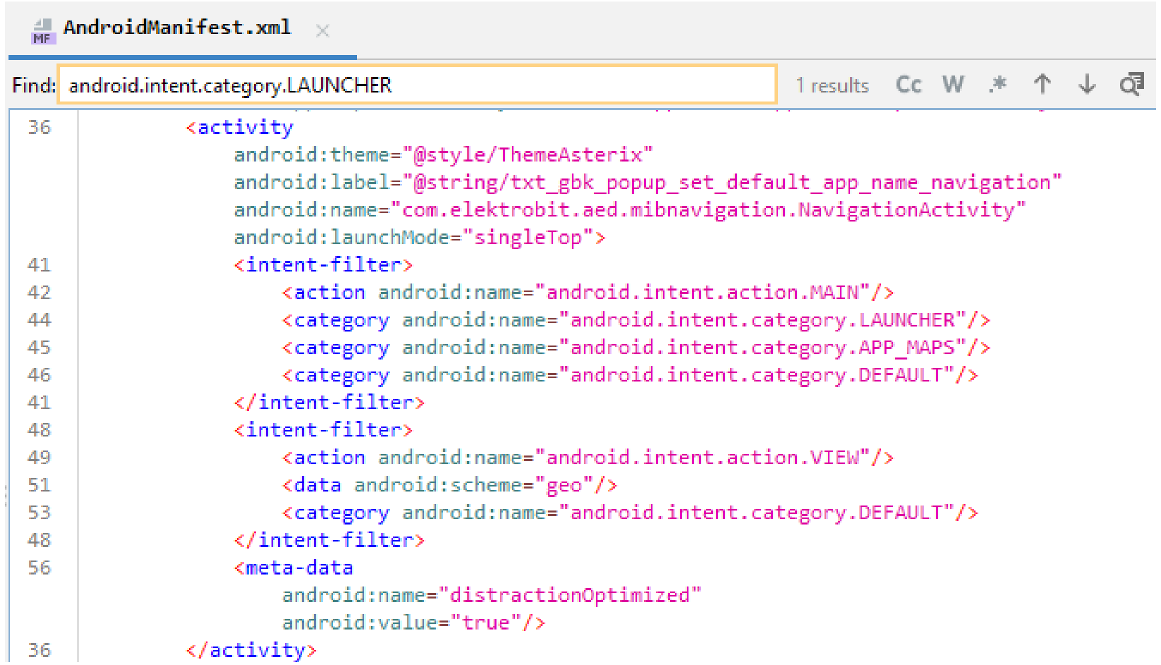
apps are not show on launcher



3rd party/Some build-in apps are hidden

Hidden services

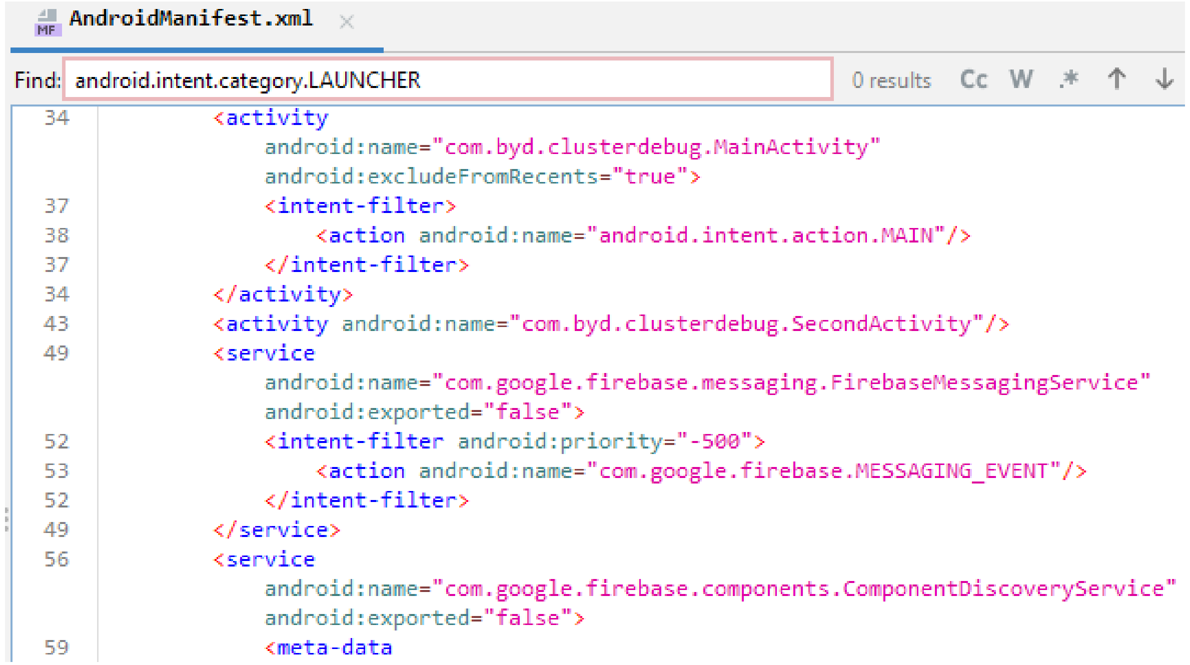
Hidden App



Find: android.intent.category.LAUNCHER

1 results

```
        <activity
            android:theme="@style/ThemeAsterix"
            android:label="@string/txt_gbk_popup_set_default_app_name_navigation"
            android:name="com.elektrobit.aed.mibnavigation.NavigationActivity"
            android:launchMode="singleTop">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
                <category android:name="android.intent.category.APP_MAPS"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
            <intent-filter>
                <action android:name="android.intent.action.VIEW"/>
                <data android:scheme="geo"/>
                <category android:name="android.intent.category.DEFAULT"/>
            </intent-filter>
            <meta-data
                android:name="distractionOptimized"
                android:value="true"/>
        </activity>
```



Find: android.intent.category.LAUNCHER

0 results

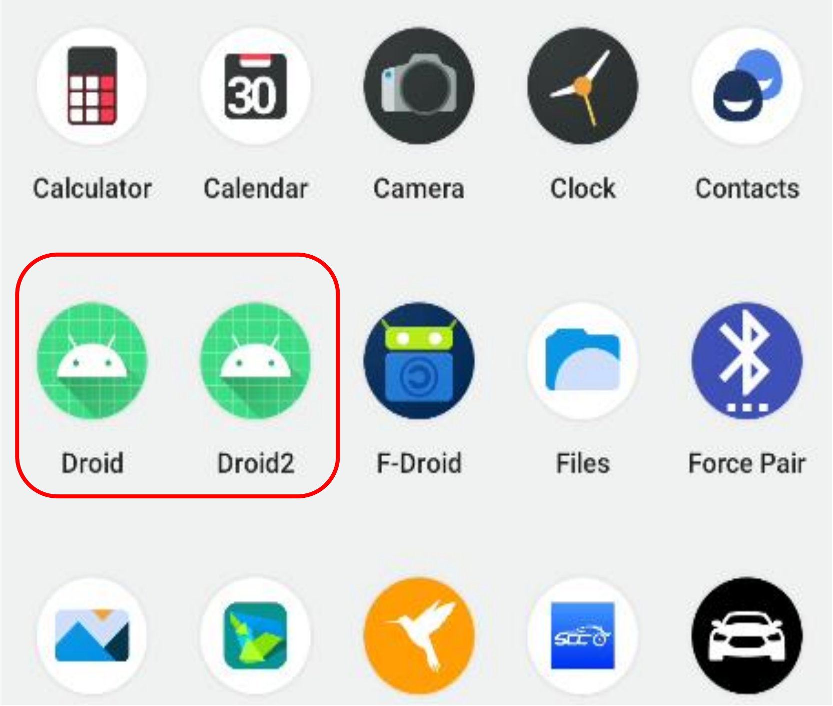
```
        <activity
            android:name="com.byd.clusterdebug.MainActivity"
            android:excludeFromRecents="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
            </intent-filter>
        </activity>
        <activity android:name="com.byd.clusterdebug.SecondActivity"/>
        <service
            android:name="com.google.firebase.messaging.FirebaseMessagingService"
            android:exported="false">
            <intent-filter android:priority="-500">
                <action android:name="com.google.firebase.MESSAGING_EVENT"/>
            </intent-filter>
        </service>
        <service
            android:name="com.google.firebase.components.ComponentDiscoveryService"
            android:exported="false">
            <meta-data
```

```
<category android:name="android.intent.category.LAUNCHER" />
```

```
/**
* Should be displayed in the top-level launcher.
*/
@SdkConstant(SdkConstantType.INTENT_CATEGORY)
public static final String CATEGORY_LAUNCHER = "android.intent.category.LAUNCHER";
```

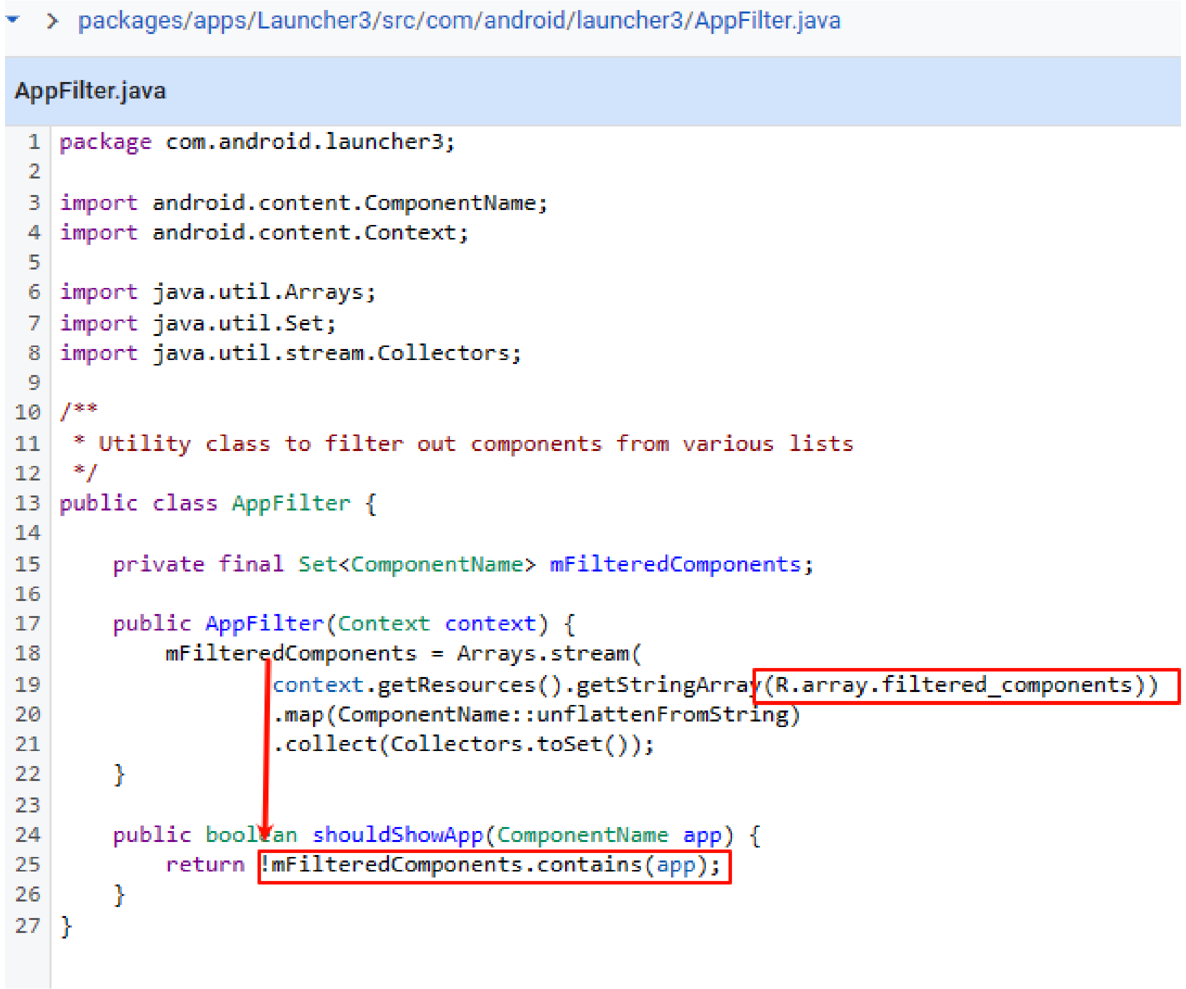
Hidden App

```
<activity
    android:name=".MainActivity"
    android:exported="true"
    android:label="@string/app_name"
    android:theme="@style/Theme.Droid.NoActionBar">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>

<activity
    android:name=".SecondActivity"
    android:exported="true"
    android:label="Droid2"
    android:theme="@style/Theme.Droid.NoActionBar">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
```



Hidden services

Filtered App

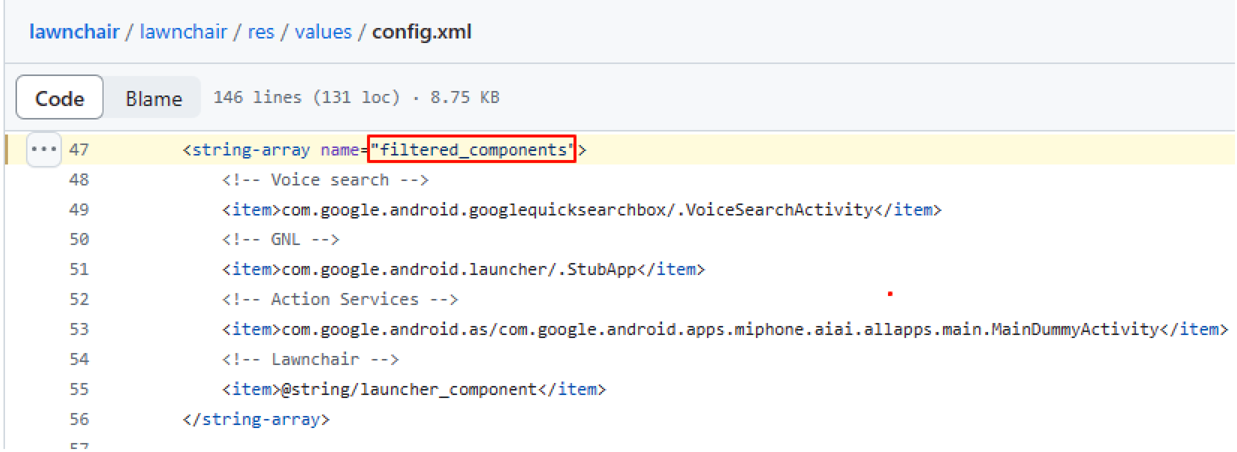


packages/apps/Launcher3/src/com/android/launcher3/AppFilter.java

AppFilter.java

```java
package com.android.launcher3;

import android.content.ComponentName;
import android.content.Context;

import java.util.Arrays;
import java.util.Set;
import java.util.stream.Collectors;

/**
 * Utility class to filter out components from various lists
 */
public class AppFilter {

    private final Set<ComponentName> mFilteredComponents;

    public AppFilter(Context context) {
        mFilteredComponents = Arrays.stream(
                context.getResources().getStringArray(R.array.filtered_components))
                .map(ComponentName::unflattenFromString)
                .collect(Collectors.toSet());
    }

    public boolean shouldShowApp(ComponentName app) {
        return !mFilteredComponents.contains(app);
    }
}
```

R.array.filtered_components An array resource defined in an config.xml,used to filter apps that are hidden from the launcher.



lawnchair / lawnchair / res / values / config.xml

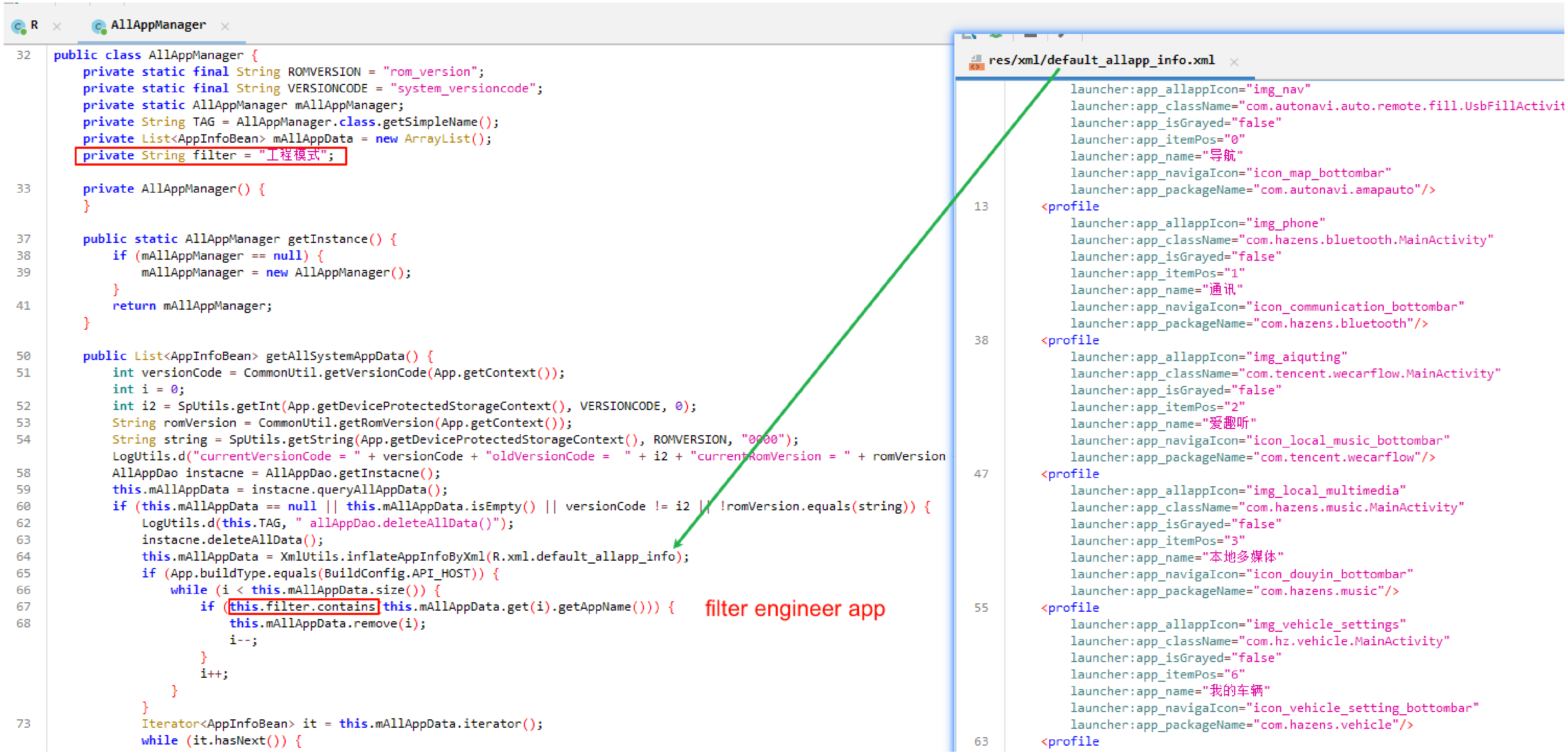
146 lines (131 loc) · 8.75 KB

```xml
    <string-array name="filtered_components">
        <!-- Voice search -->
        <item>com.google.android.googlequicksearchbox/.VoiceSearchActivity</item>
        <!-- GNL -->
        <item>com.google.android.launcher/.StubApp</item>
        <!-- Action Services -->
        <item>com.google.android.as/com.google.android.apps.miphone.aiai.allapps.main.MainDummyActivity</item>
        <!-- Lawnchair -->
        <item>@string/launcher_component</item>
    </string-array>
```

Hidden services

Filtered App

Another way to hide apps



```java
public class AllAppManager {
    private static final String ROMVERSION = "rom_version";
    private static final String VERSIONCODE = "system_versioncode";
    private static AllAppManager mAllAppManager;
    private String TAG = AllAppManager.class.getSimpleName();
    private List<AppInfoBean> mAllAppData = new ArrayList();
    private String filter = "工程模式";

    private AllAppManager() {
    }

    public static AllAppManager getInstance() {
        if (mAllAppManager == null) {
            mAllAppManager = new AllAppManager();
        }
        return mAllAppManager;
    }

    public List<AppInfoBean> getAllSystemAppData() {
        int versionCode = CommonUtil.getVersionCode(App.getContext());
        int i = 0;
        int i2 = SpUtils.getInt(App.getDeviceProtectedStorageContext(), VERSIONCODE, 0);
        String romVersion = CommonUtil.getRomVersion(App.getContext());
        String string = SpUtils.getString(App.getDeviceProtectedStorageContext(), ROMVERSION, "0000");
        LogUtils.d("currentVersionCode = " + versionCode + "oldVersionCode =  " + i2 + "currentRomVersion = " + romVersion
        AllAppDao instacne = AllAppDao.getInstacne();
        this.mAllAppData = instacne.queryAllAppData();
        if (this.mAllAppData == null || this.mAllAppData.isEmpty() || versionCode != i2 || !romVersion.equals(string)) {
            LogUtils.d(this.TAG, " allAppDao.deleteAllData()");
            instacne.deleteAllData();
            this.mAllAppData = XmlUtils.inflateAppInfoByXml(R.xml.default_allapp_info);
            if (App.buildType.equals(BuildConfig.API_HOST)) {
                while (i < this.mAllAppData.size()) {
                    if (this.filter.contains(this.mAllAppData.get(i).getAppName())) {
                        this.mAllAppData.remove(i);
                        i--;
                    }
                    i++;
                }
            }
            Iterator<AppInfoBean> it = this.mAllAppData.iterator();
            while (it.hasNext()) {
```

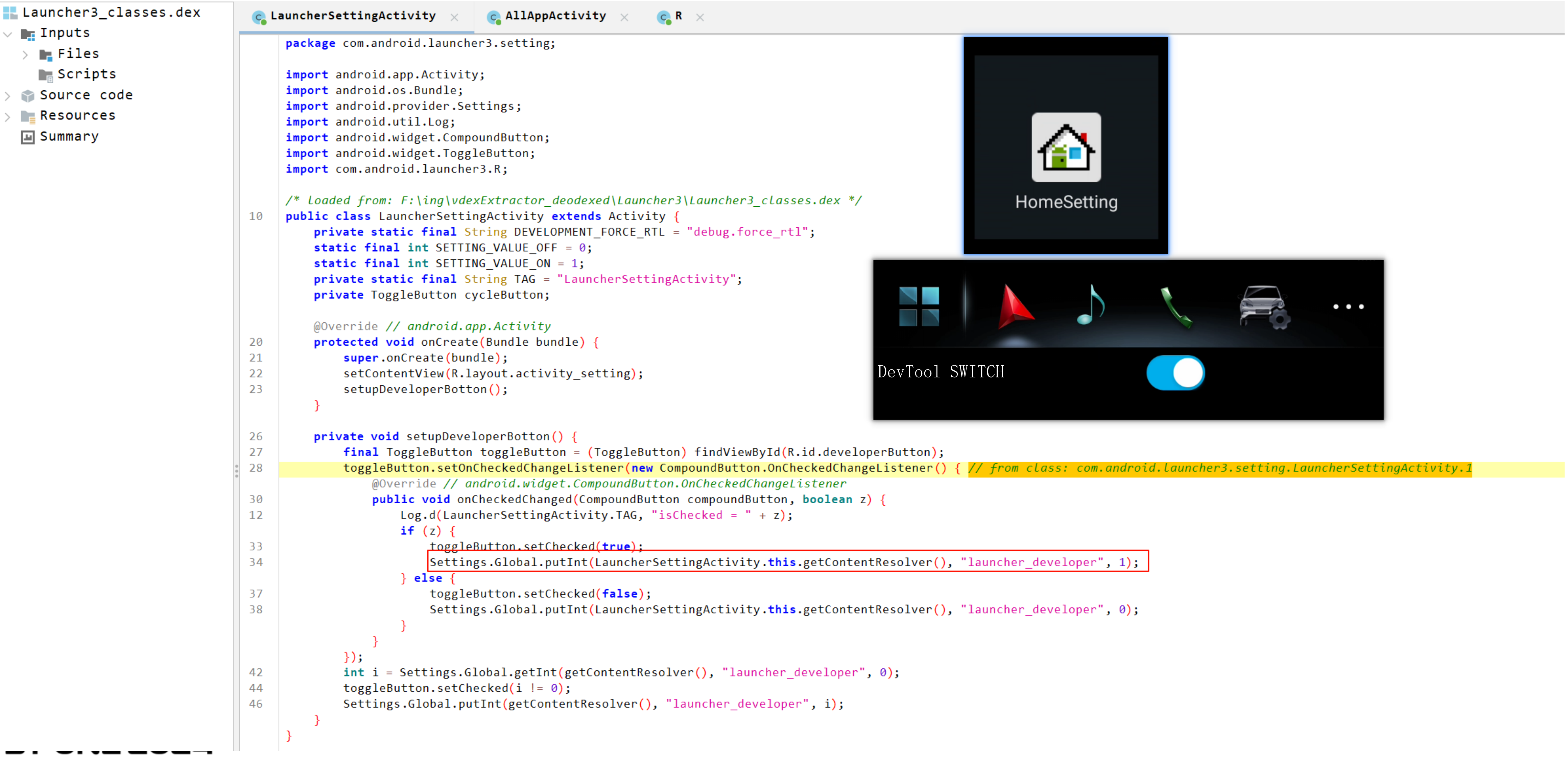
filter engineer app

```xml
            launcher:app_allappIcon="img_nav"
            launcher:app_className="com.autonavi.auto.remote.fill.UsbFillActivit
            launcher:app_isGrayed="false"
            launcher:app_itemPos="0"
            launcher:app_name="导航"
            launcher:app_navigaIcon="icon_map_bottombar"
            launcher:app_packageName="com.autonavi.amapauto"/>
        <profile
            launcher:app_allappIcon="img_phone"
            launcher:app_className="com.hazens.bluetooth.MainActivity"
            launcher:app_isGrayed="false"
            launcher:app_itemPos="1"
            launcher:app_name="通讯"
            launcher:app_navigaIcon="icon_communication_bottombar"
            launcher:app_packageName="com.hazens.bluetooth"/>
        <profile
            launcher:app_allappIcon="img_aiquting"
            launcher:app_className="com.tencent.wecarflow.MainActivity"
            launcher:app_isGrayed="false"
            launcher:app_itemPos="2"
            launcher:app_name="爱趣听"
            launcher:app_navigaIcon="icon_local_music_bottombar"
            launcher:app_packageName="com.tencent.wecarflow"/>
        <profile
            launcher:app_allappIcon="img_local_multimedia"
            launcher:app_className="com.hazens.music.MainActivity"
            launcher:app_isGrayed="false"
            launcher:app_itemPos="3"
            launcher:app_name="本地多媒体"
            launcher:app_navigaIcon="icon_douyin_bottombar"
            launcher:app_packageName="com.hazens.music"/>
        <profile
            launcher:app_allappIcon="img_vehicle_settings"
            launcher:app_className="com.hz.vehicle.MainActivity"
            launcher:app_isGrayed="false"
            launcher:app_itemPos="6"
            launcher:app_name="我的车辆"
            launcher:app_navigaIcon="icon_vehicle_setting_bottombar"
            launcher:app_packageName="com.hazens.vehicle"/>
        <profile
```

Hidden services

Hidden App

settings launcher_developer=1 make hidden App available



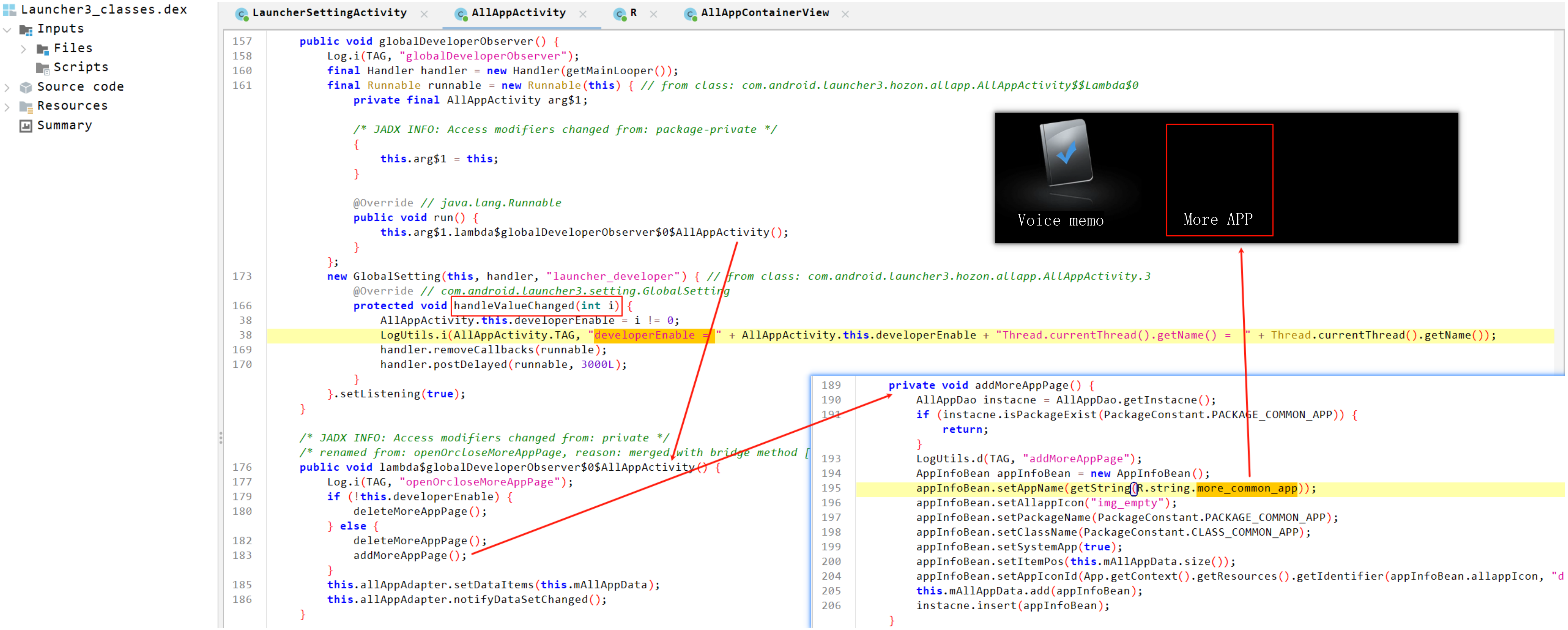
DevTool SWITCH

```
package com.android.launcher3.setting;

import android.app.Activity;
import android.os.Bundle;
import android.provider.Settings;
import android.util.Log;
import android.widget.CompoundButton;
import android.widget.ToggleButton;
import com.android.launcher3.R;

/* loaded from: F:\ing\vdexExtractor_deodexed\Launcher3\Launcher3_classes.dex */
public class LauncherSettingActivity extends Activity {
    private static final String DEVELOPMENT_FORCE_RTL = "debug.force_rtl";
    static final int SETTING_VALUE_OFF = 0;
    static final int SETTING_VALUE_ON = 1;
    private static final String TAG = "LauncherSettingActivity";
    private ToggleButton cycleButton;

    @Override // android.app.Activity
    protected void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.activity_setting);
        setupDeveloperBotton();
    }

    private void setupDeveloperBotton() {
        final ToggleButton toggleButton = (ToggleButton) findViewById(R.id.developerButton);
        toggleButton.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() { // from class: com.android.launcher3.setting.LauncherSettingActivity.1
            @Override // android.widget.CompoundButton.OnCheckedChangeListener
            public void onCheckedChanged(CompoundButton compoundButton, boolean z) {
                Log.d(LauncherSettingActivity.TAG, "isChecked = " + z);
                if (z) {
                    toggleButton.setChecked(true);
                    Settings.Global.putInt(LauncherSettingActivity.this.getContentResolver(), "launcher_developer", 1);
                } else {
                    toggleButton.setChecked(false);
                    Settings.Global.putInt(LauncherSettingActivity.this.getContentResolver(), "launcher_developer", 0);
                }
            }
        });
        int i = Settings.Global.getInt(getContentResolver(), "launcher_developer", 0);
        toggleButton.setChecked(i != 0);
        Settings.Global.putInt(getContentResolver(), "launcher_developer", i);
    }
}
```

Hidden services

Hidden App

handleValueChanged listens for changes in settings and calls **addMoreAppPage** to add desktop applications.



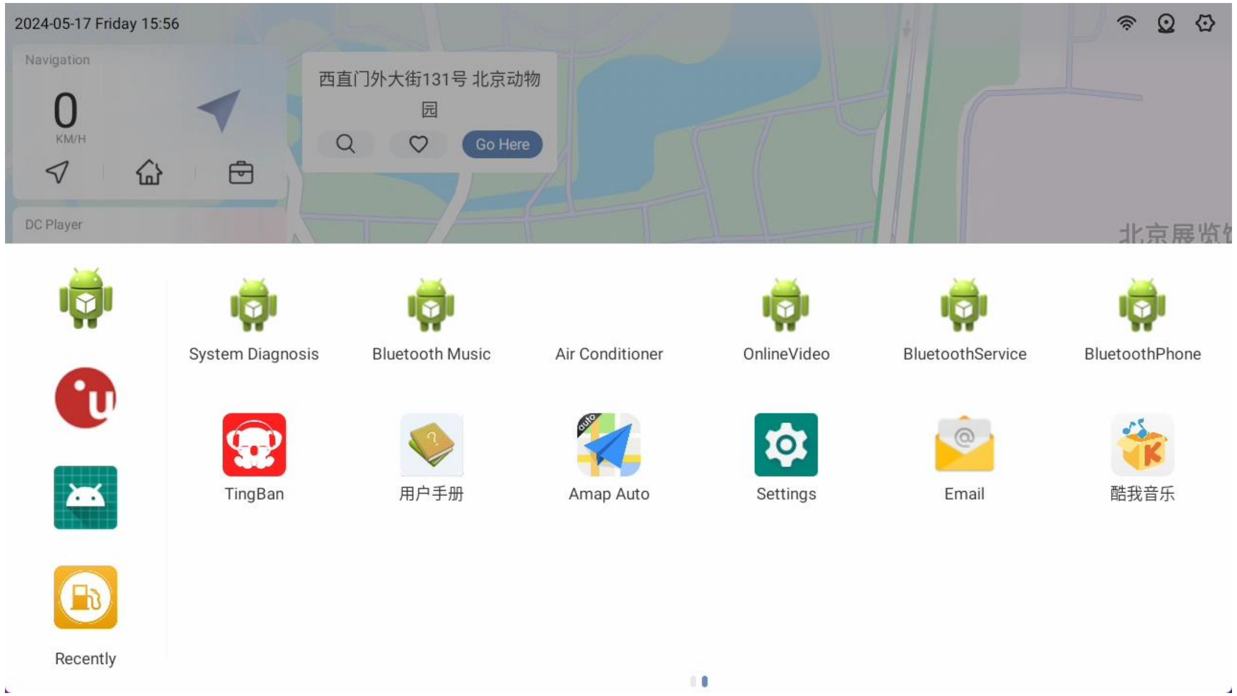
```
    public void globalDeveloperObserver() {
        Log.i(TAG, "globalDeveloperObserver");
        final Handler handler = new Handler(getMainLooper());
        final Runnable runnable = new Runnable(this) { // from class: com.android.launcher3.hozon.allapp.AllAppActivity$$Lambda$0
            private final AllAppActivity arg$1;

            /* JADX INFO: Access modifiers changed from: package-private */
            {
                this.arg$1 = this;
            }

            @Override // java.lang.Runnable
            public void run() {
                this.arg$1.lambda$globalDeveloperObserver$0$AllAppActivity();
            }
        };
        new GlobalSetting(this, handler, "launcher_developer") { // from class: com.android.launcher3.hozon.allapp.AllAppActivity.3
            @Override // com.android.launcher3.setting.GlobalSetting
            protected void handleValueChanged(int i) {
                AllAppActivity.this.developerEnable = i != 0;
                LogUtils.i(AllAppActivity.TAG, "developerEnable = " + AllAppActivity.this.developerEnable + "Thread.currentThread().getName() =  " + Thread.currentThread().getName());
                handler.removeCallbacks(runnable);
                handler.postDelayed(runnable, 3000L);
            }
        }.setListening(true);
    }

    /* JADX INFO: Access modifiers changed from: private */
    /* renamed from: openOrcloseMoreAppPage, reason: merged with bridge method [
    public void lambda$globalDeveloperObserver$0$AllAppActivity() {
        Log.i(TAG, "openOrcloseMoreAppPage");
        if (!this.developerEnable) {
            deleteMoreAppPage();
        } else {
            deleteMoreAppPage();
            addMoreAppPage();
        }
        this.allAppAdapter.setDataItems(this.mAllAppData);
        this.allAppAdapter.notifyDataSetChanged();
    }
```

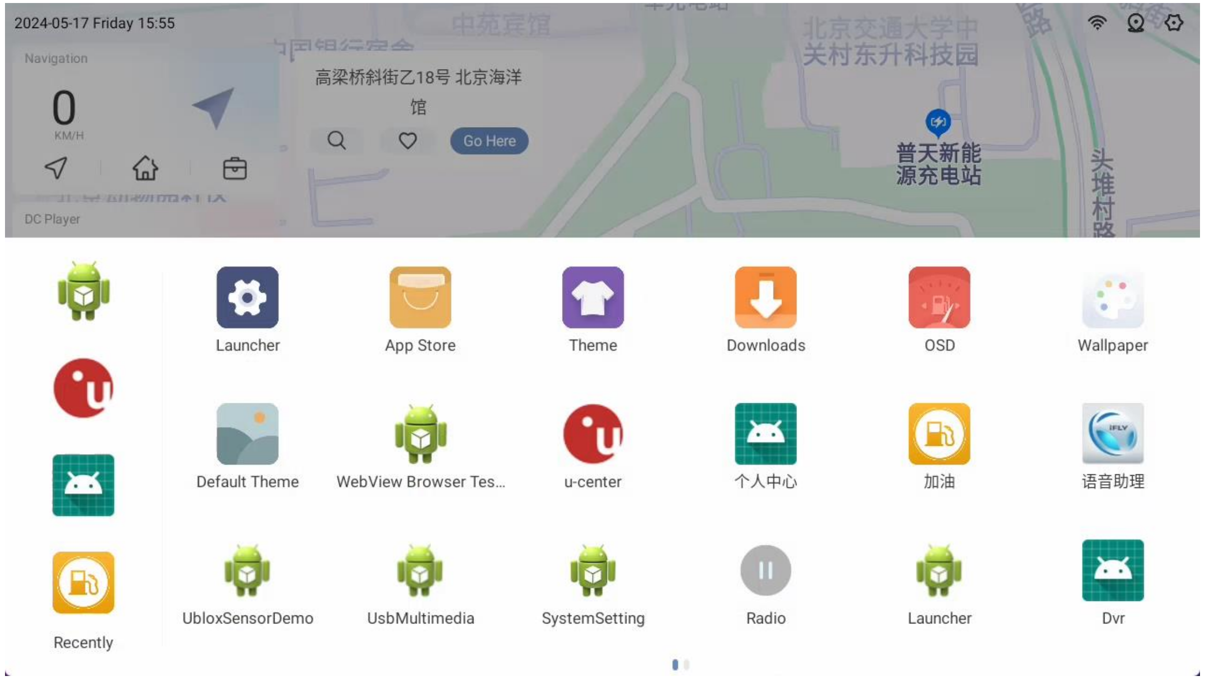
```
    private void addMoreAppPage() {
        AllAppDao instacne = AllAppDao.getInstacne();
        if (instacne.isPackageExist(PackageConstant.PACKAGE_COMMON_APP)) {
            return;
        }
        LogUtils.d(TAG, "addMoreAppPage");
        AppInfoBean appInfoBean = new AppInfoBean();
        appInfoBean.setAppName(getString(R.string.more_common_app));
        appInfoBean.setAllappIcon("img_empty");
        appInfoBean.setPackageName(PackageConstant.PACKAGE_COMMON_APP);
        appInfoBean.setClassName(PackageConstant.CLASS_COMMON_APP);
        appInfoBean.setSystemApp(true);
        appInfoBean.setItemPos(this.mAllAppData.size());
        appInfoBean.setAppIconId(App.getContext().getResources().getIdentifier(appInfoBean.allappIcon, "d
        this.mAllAppData.add(appInfoBean);
        instacne.insert(appInfoBean);
    }
```

Hidden services

Show all apps

Use third-party launcher show all apps



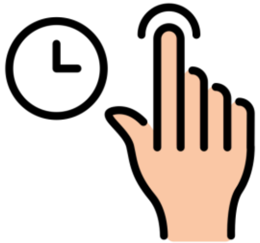


Key

“Keys” and “windows”

Before unlock,know keys type first

Multiple clicks



Long press

Password Protect

Dial codes

Voice control

USB Stick

Diagnostic toolkit

Malicious apps

Key

Material

MotionEvent.ACTION_DOWN(0x00000000) Press

MotionEvent.ACTION_UP (0x00000001) Rise

MotionEvent. ACTION_MOVE (0x00000001)

MotionEvent.ACTION_CANCEL(0x00000003)

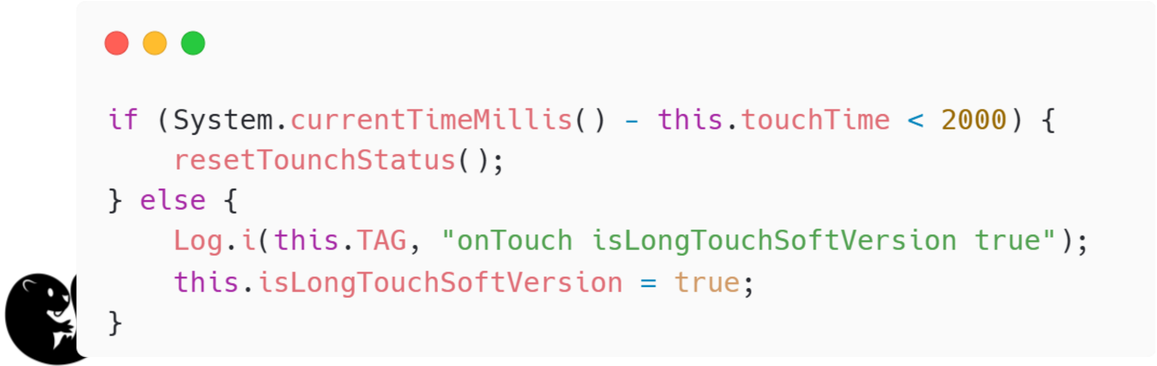
getAction()

getX()

Axis

getY()

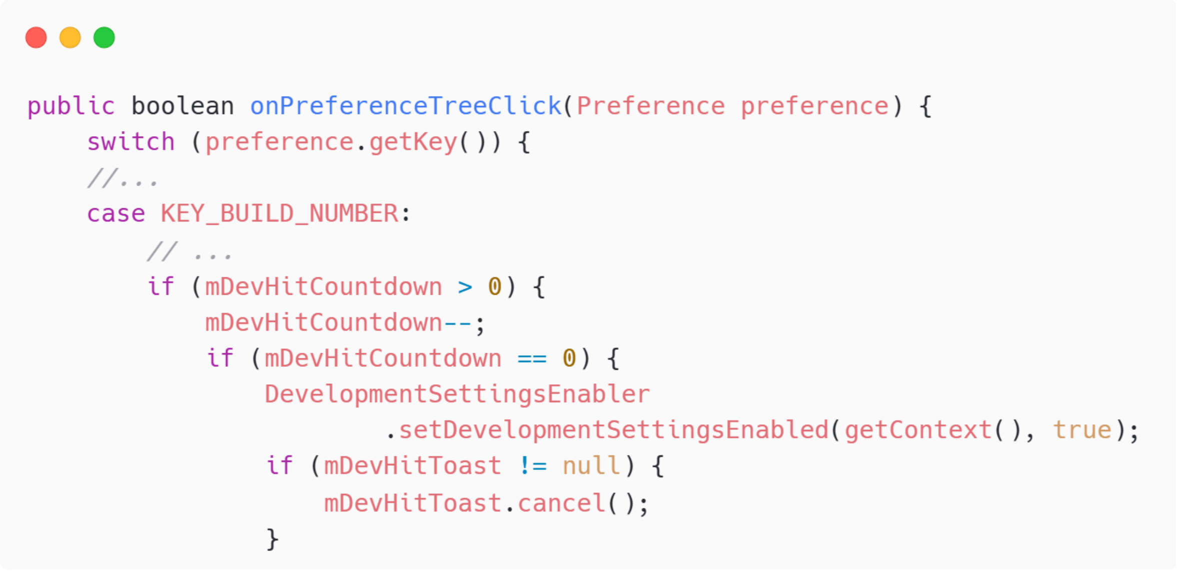
Long press on SoftVersion



```
if (System.currentTimeMillis() - this.touchTime < 2000) {
    resetTounchStatus();
} else {
    Log.i(this.TAG, "onTouch isLongTouchSoftVersion true");
    this.isLongTouchSoftVersion = true;
}
```

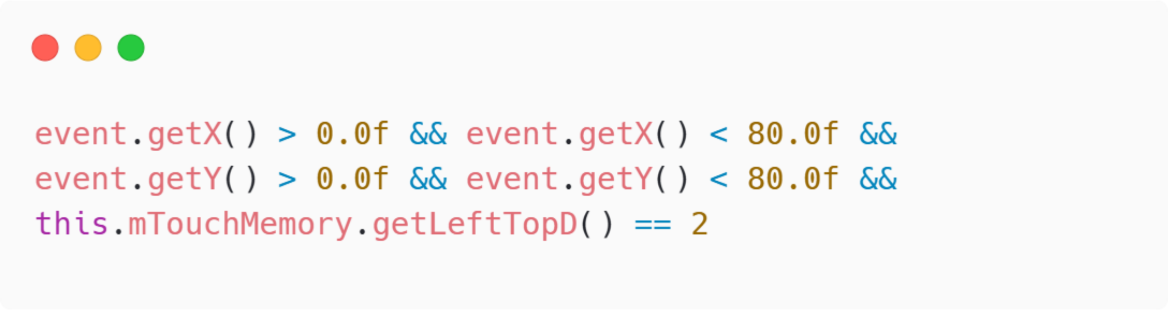
X3

Three click on preference



```
public boolean onPreferenceTreeClick(Preference preference) {
    switch (preference.getKey()) {
    //...
    case KEY_BUILD_NUMBER:
        // ...
        if (mDevHitCountdown > 0) {
            mDevHitCountdown--;
            if (mDevHitCountdown == 0) {
                DevelopmentSettingsEnabler
                        .setDevelopmentSettingsEnabled(getContext(), true);
                if (mDevHitToast != null) {
                    mDevHitToast.cancel();
                }
```

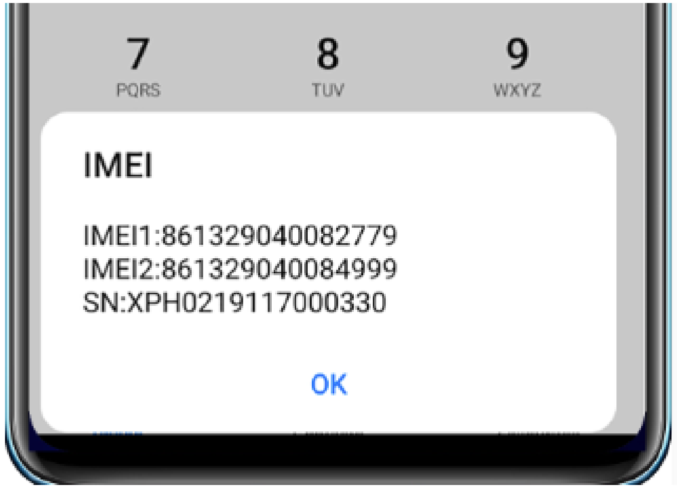
Specific areas long press



```
event.getX() > 0.0f && event.getX() < 80.0f &&
event.getY() > 0.0f && event.getY() < 80.0f &&
this.mTouchMemory.getLeftTopD() == 2
```

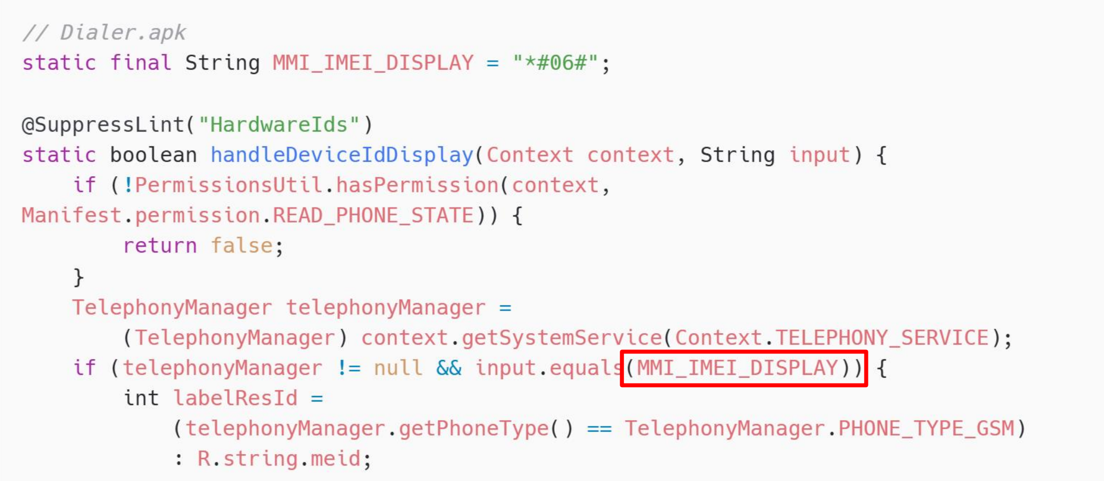
Key

android_secret_code



IMEI

IMEI1:861329040082779
IMEI2:861329040084999
SN:XPH0219117000330

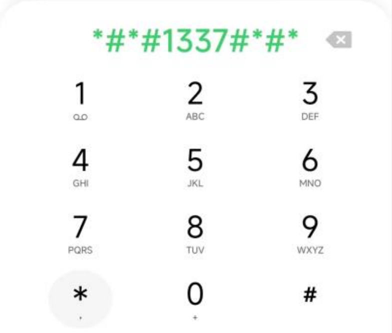


```
// Dialer.apk
static final String MMI_IMEI_DISPLAY = "*#06#";

@SuppressLint("HardwareIds")
static boolean handleDeviceIdDisplay(Context context, String input) {
    if (!PermissionsUtil.hasPermission(context,
Manifest.permission.READ_PHONE_STATE)) {
        return false;
    }
    TelephonyManager telephonyManager =
        (TelephonyManager) context.getSystemService(Context.TELEPHONY_SERVICE);
    if (telephonyManager != null && input.equals(MMI_IMEI_DISPLAY)) {
        int labelResId =
            (telephonyManager.getPhoneType() == TelephonyManager.PHONE_TYPE_GSM)
            : R.string.meid;
```

In Android a secret code is defined by this pattern: *#*#<code>#*#*.

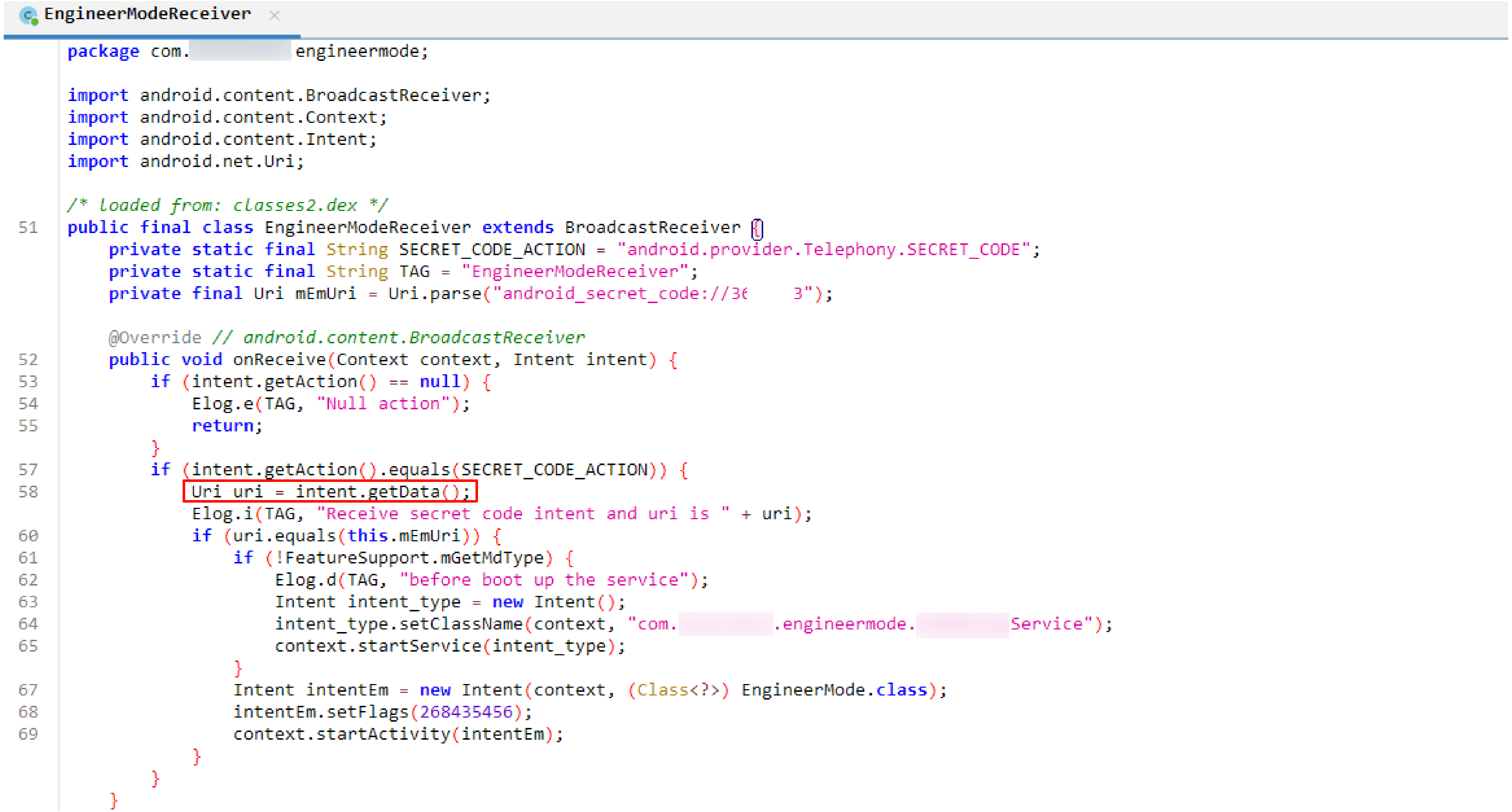
android.provider.Telephony.SECRET_CODE

```
//AndroidManifest.xml
<receiver android:name=".MySecretCodeReceiver">
    <intent-filter>
        <action android:name="android.provider.Telephony.SECRET_CODE" />
        <data android:scheme="android_secret_code" android:host="1337" />
    </intent-filter>
</receiver>
```



Key

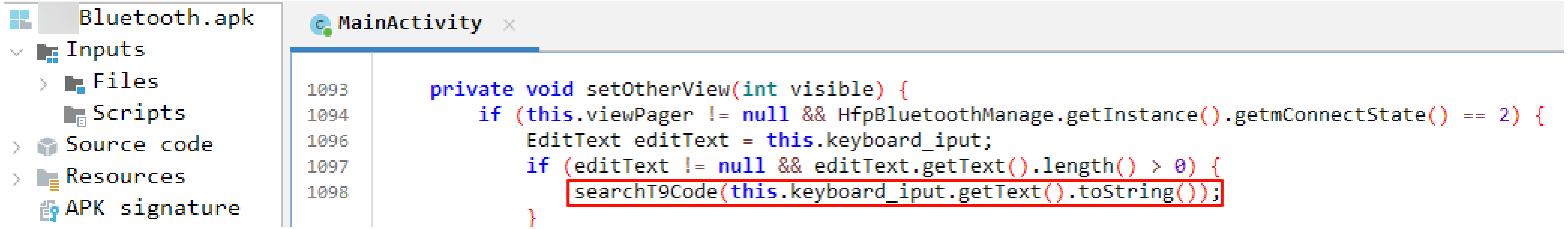
android_secret_code



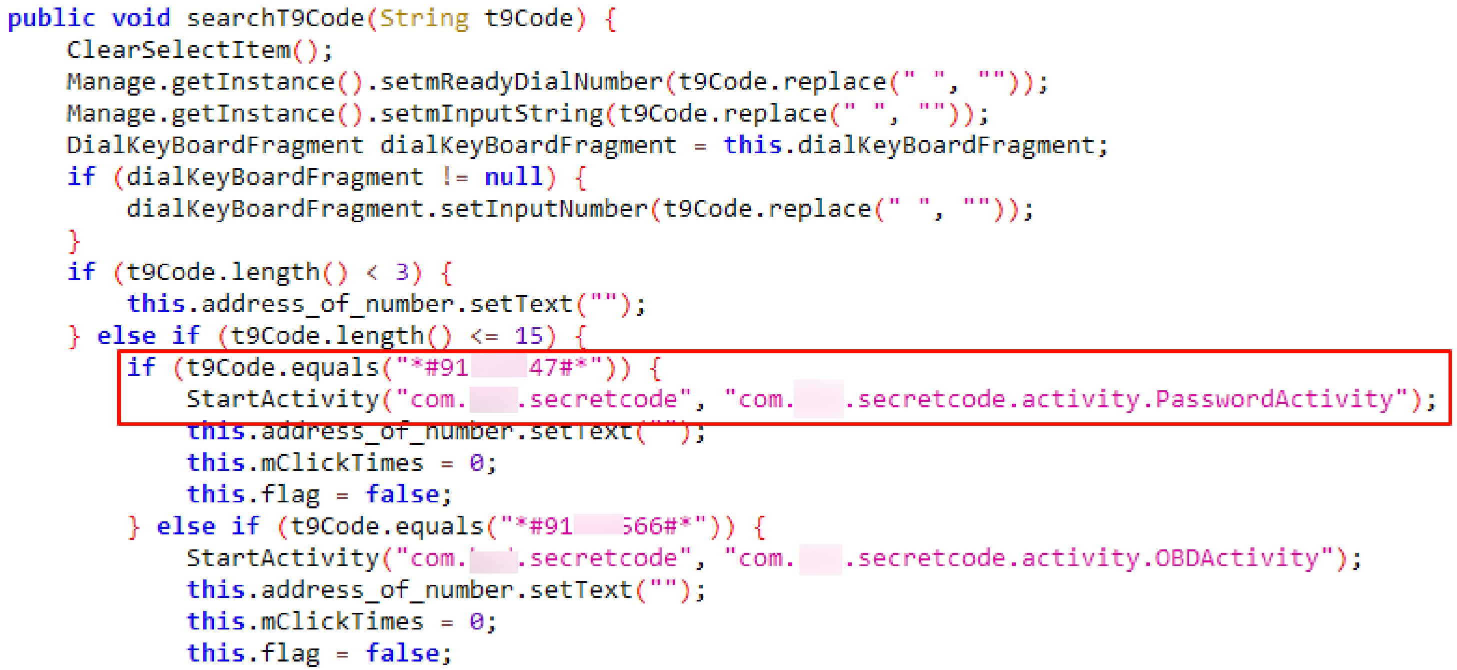
```
package com.[redacted]engineermode;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.net.Uri;

/* loaded from: classes2.dex */
public final class EngineerModeReceiver extends BroadcastReceiver {
    private static final String SECRET_CODE_ACTION = "android.provider.Telephony.SECRET_CODE";
    private static final String TAG = "EngineerModeReceiver";
    private final Uri mEmUri = Uri.parse("android_secret_code://3[redacted]3");

    @Override // android.content.BroadcastReceiver
    public void onReceive(Context context, Intent intent) {
        if (intent.getAction() == null) {
            Elog.e(TAG, "Null action");
            return;
        }
        if (intent.getAction().equals(SECRET_CODE_ACTION)) {
            Uri uri = intent.getData();
            Elog.i(TAG, "Receive secret code intent and uri is " + uri);
            if (uri.equals(this.mEmUri)) {
                if (!FeatureSupport.mGetMdType) {
                    Elog.d(TAG, "before boot up the service");
                    Intent intent_type = new Intent();
                    intent_type.setClassName(context, "com.[redacted].engineermode.[redacted]Service");
                    context.startService(intent_type);
                }
                Intent intentEm = new Intent(context, (Class<?>) EngineerMode.class);
                intentEm.setFlags(268435456);
                context.startActivity(intentEm);
            }
        }
    }
```

Key

Like android_secret_code



```
    private void setOtherView(int visible) {
        if (this.viewPager != null && HfpBluetoothManage.getInstance().getmConnectState() == 2) {
            EditText editText = this.keyboard_iput;
            if (editText != null && editText.getText().length() > 0) {
                searchT9Code(this.keyboard_iput.getText().toString());
            }
```



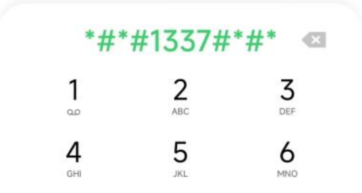
```
public void searchT9Code(String t9Code) {
    ClearSelectItem();
    Manage.getInstance().setmReadyDialNumber(t9Code.replace(" ", ""));
    Manage.getInstance().setmInputString(t9Code.replace(" ", ""));
    DialKeyBoardFragment dialKeyBoardFragment = this.dialKeyBoardFragment;
    if (dialKeyBoardFragment != null) {
        dialKeyBoardFragment.setInputNumber(t9Code.replace(" ", ""));
    }
    if (t9Code.length() < 3) {
        this.address_of_number.setText("");
    } else if (t9Code.length() <= 15) {
        if (t9Code.equals("*#91[redacted]47#*")) {
            StartActivity("com.[redacted].secretcode", "com.[redacted].secretcode.activity.PasswordActivity");
            this.address_of_number.setText("");
            this.mClickTimes = 0;
            this.flag = false;
        } else if (t9Code.equals("*#91[redacted]566#*")) {
            StartActivity("com.[redacted].secretcode", "com.[redacted].secretcode.activity.OBDActivity");
            this.address_of_number.setText("");
            this.mClickTimes = 0;
            this.flag = false;
```

Key

android_secret_code

Dail

```
String secretCode = "1337";
Intent intent = new Intent(Intent.ACTION_DIAL);
intent.setData(Uri.parse("tel:*#*#" + secretCode + "#*#*"));
startActivity(intent);
```



##1337#*#*

Deeplink

```
String secretCode = "1337";
String action = "android.provider.Telephony.SECRET_CODE";
Uri uri = Uri.parse("android_secret_code://" + secretCode);
Intent intent = new Intent(action, uri);
sendBroadcast(intent);
```

adb

```
adb shell am start -a android.intent.action.VIEW -d android_secret_code://1337
```

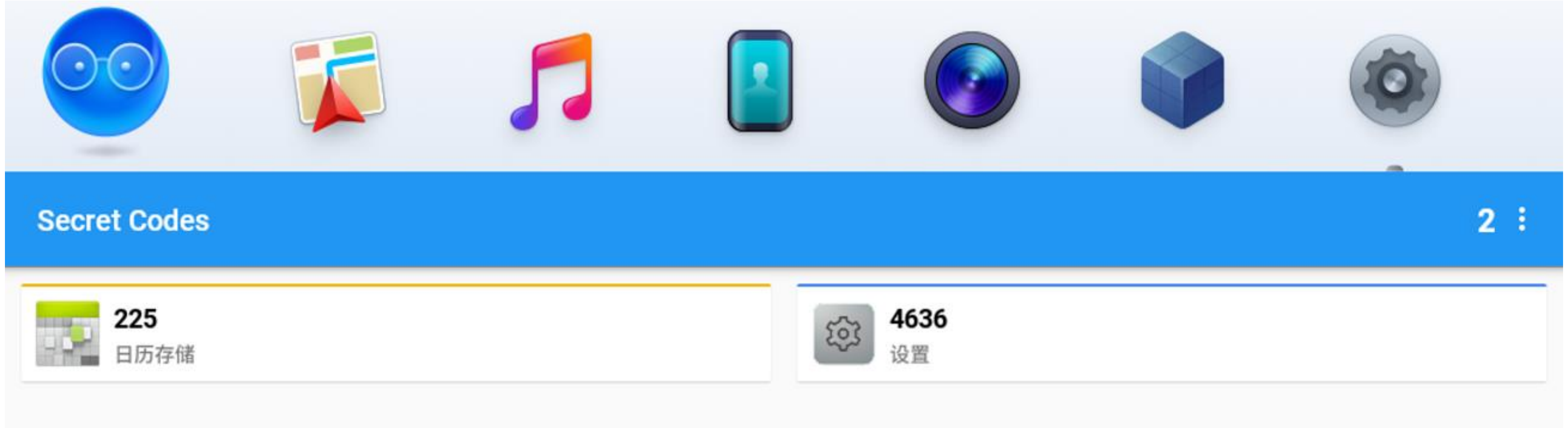
Browser/Webview

```
android_secret_code://1337
```

Key

android_secret_code

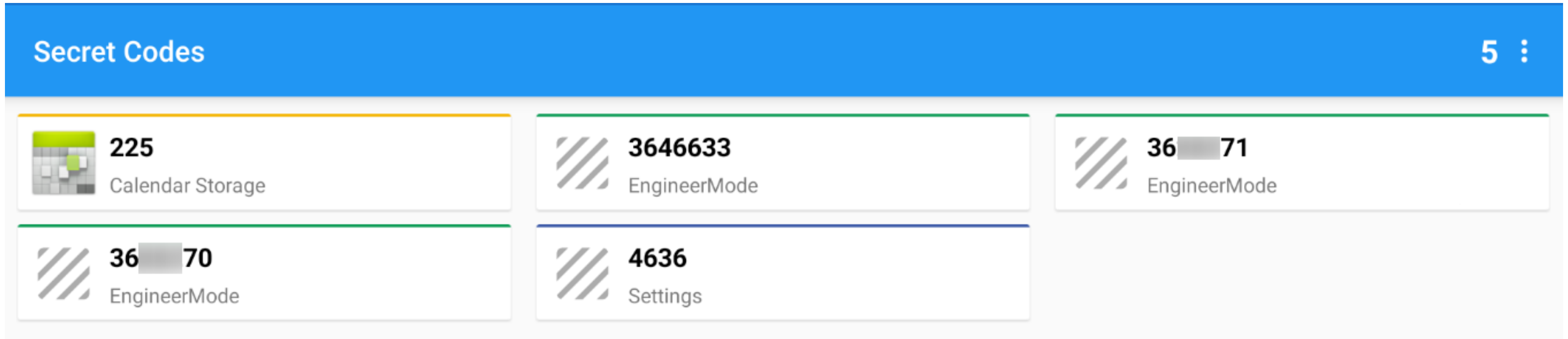
Open source application “Secret Codes” allows you to scan the secret codes available on your device through the dialer app.



Secret Codes 2 ⋮

225 日历存储

4636 设置



Secret Codes 5 ⋮

225 Calendar Storage

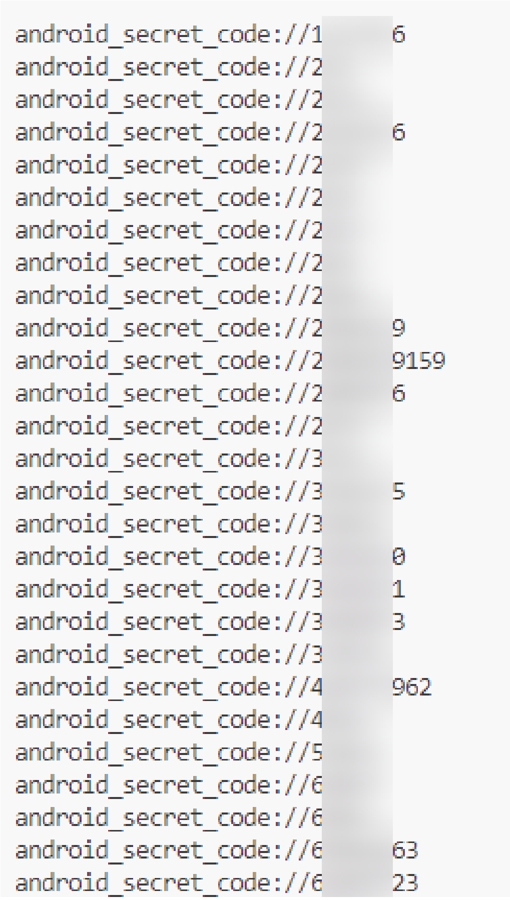
3646633 EngineerMode

36[redacted]71 EngineerMode

36[redacted]70 EngineerMode

4636 Settings

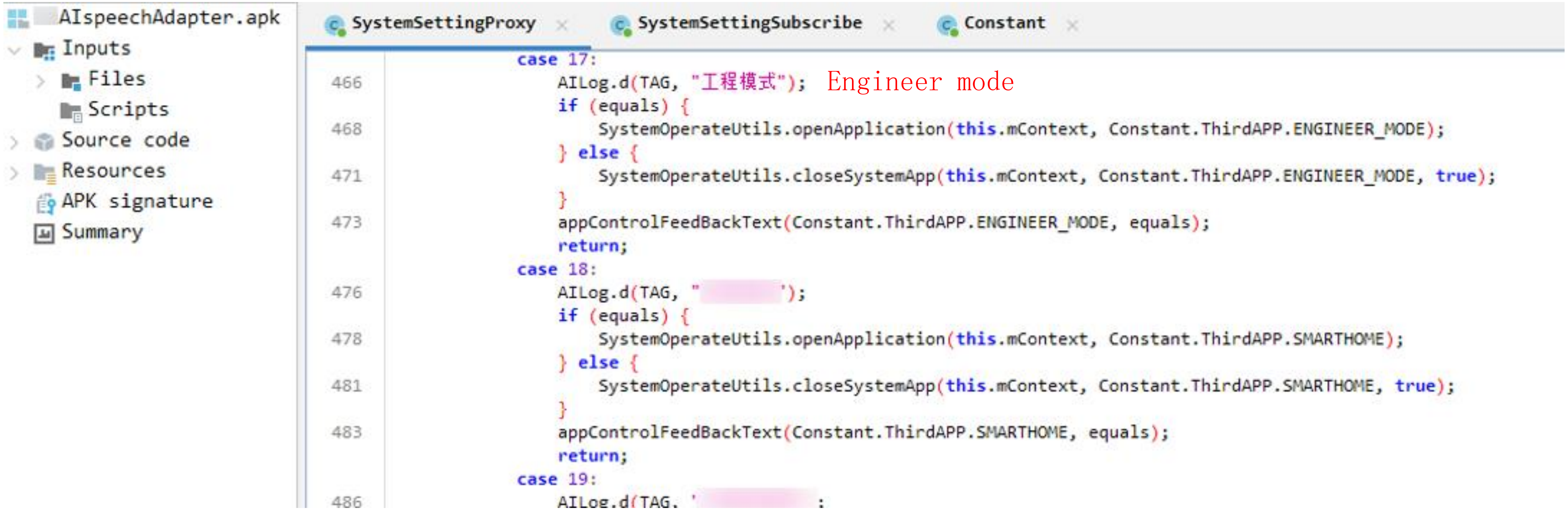
Use REGEX get Secret Codes without install application



```
android_secret_code://1[redacted]6
android_secret_code://2[redacted]
android_secret_code://2[redacted]
android_secret_code://2[redacted]6
android_secret_code://2[redacted]
android_secret_code://2[redacted]
android_secret_code://2[redacted]
android_secret_code://2[redacted]
android_secret_code://2[redacted]
android_secret_code://2[redacted]9
android_secret_code://2[redacted]9159
android_secret_code://2[redacted]6
android_secret_code://2[redacted]
android_secret_code://3[redacted]
android_secret_code://3[redacted]5
android_secret_code://3[redacted]
android_secret_code://3[redacted]0
android_secret_code://3[redacted]1
android_secret_code://3[redacted]3
android_secret_code://3[redacted]
android_secret_code://4[redacted]962
android_secret_code://4[redacted]
android_secret_code://5[redacted]
android_secret_code://6[redacted]
android_secret_code://6[redacted]
android_secret_code://6[redacted]63
android_secret_code://6[redacted]23
```

Key

Voice



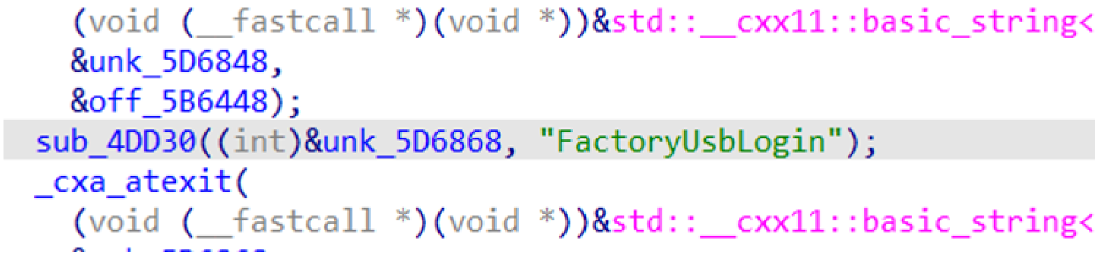
```
case 17:
    AILog.d(TAG, "工程模式");
    if (equals) {
        SystemOperateUtils.openApplication(this.mContext, Constant.ThirdAPP.ENGINEER_MODE);
    } else {
        SystemOperateUtils.closeSystemApp(this.mContext, Constant.ThirdAPP.ENGINEER_MODE, true);
    }
    appControlFeedBackText(Constant.ThirdAPP.ENGINEER_MODE, equals);
    return;
case 18:
    AILog.d(TAG, "[redacted]");
    if (equals) {
        SystemOperateUtils.openApplication(this.mContext, Constant.ThirdAPP.SMARTHOME);
    } else {
        SystemOperateUtils.closeSystemApp(this.mContext, Constant.ThirdAPP.SMARTHOME, true);
    }
    appControlFeedBackText(Constant.ThirdAPP.SMARTHOME, equals);
    return;
case 19:
    AILog.d(TAG, "[redacted]:
```

Engineer mode

Key

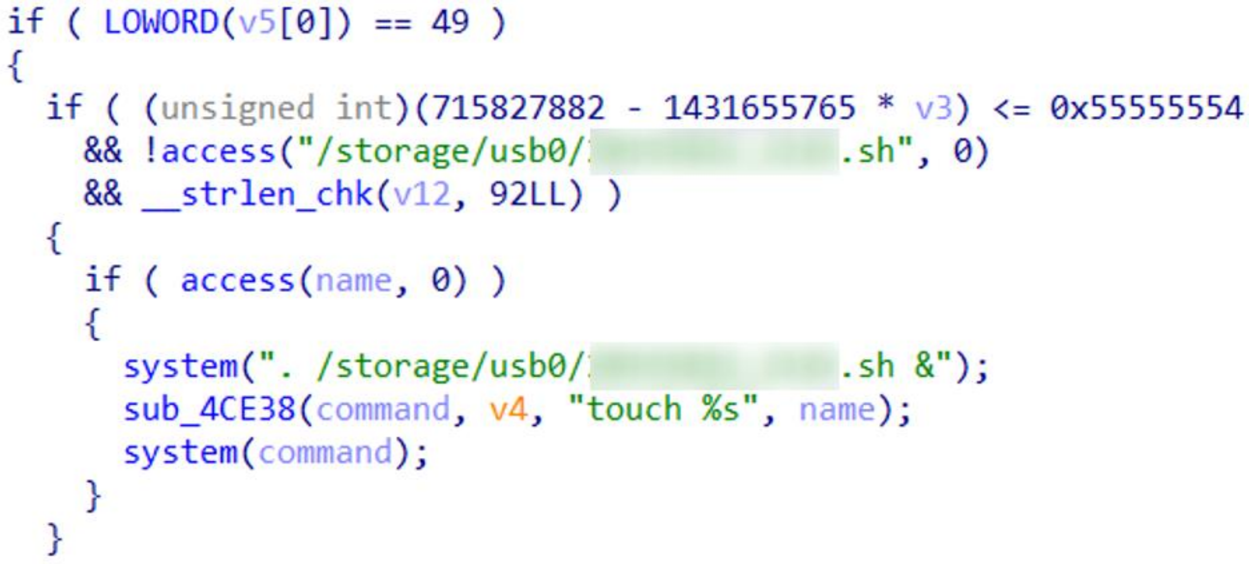
USB key/shell

Security debug

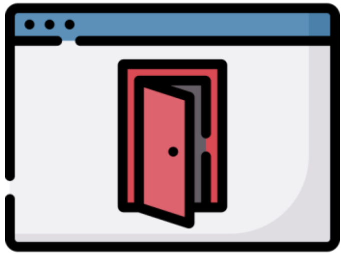


```
    (void (__fastcall *)(void *))&std::__cxx11::basic_string<
    &unk_5D6848,
    &off_5B6448);
  sub_4DD30((int)&unk_5D6868, "FactoryUsbLogin");
  _cxa_atexit(
    (void (__fastcall *)(void *))&std::__cxx11::basic_string<
```

Backdoor



```
if ( LOWORD(v5[0]) == 49 )
{
  if ( (unsigned int)(715827882 - 1431655765 * v3) <= 0x55555554
    && !access("/storage/usb0/[redacted].sh", 0)
    && __strlen_chk(v12, 92LL) )
  {
    if ( access(name, 0) )
    {
      system(". /storage/usb0/[redacted].sh &");
      sub_4CE38(command, v4, "touch %s", name);
      system(command);
    }
  }
```



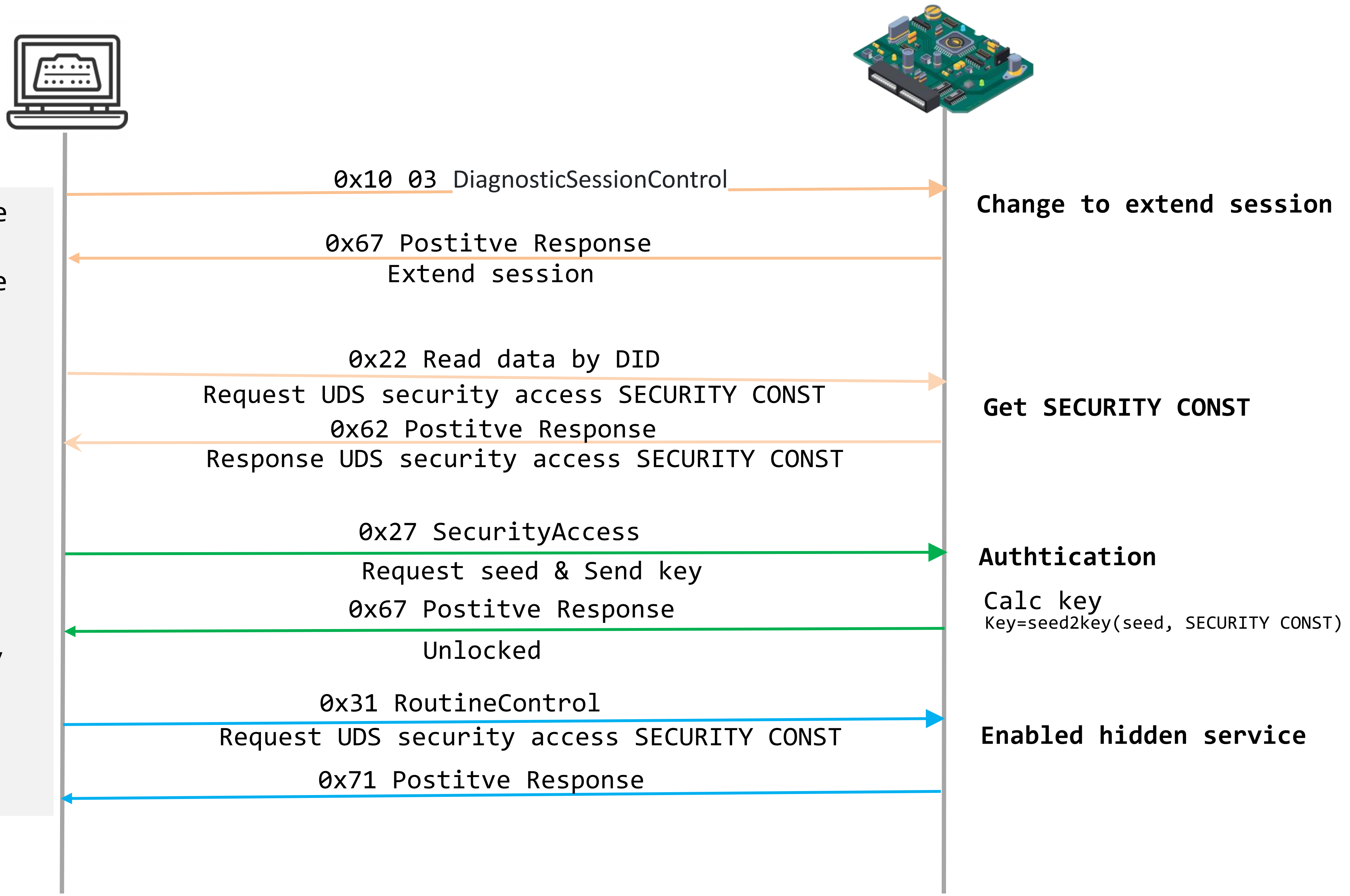
Key

UDS RoutineControl

The RoutineControl service is used by the client to execute a defined sequence of steps and obtain any relevant results.

typical usage

- erasing memory
- resetting
- learning adaptive data
- running a self-test
- overriding the normal
- server control strategy
- other custom functions



0x10 03 DiagnosticSessionControl

Change to extend session

0x67 Postitve Response

Extend session

0x22 Read data by DID

Request UDS security access SECURITY CONST

Get SECURITY CONST

0x62 Postitve Response

Response UDS security access SECURITY CONST

0x27 SecurityAccess

Request seed & Send key

Authtication

Calc key

Key=seed2key(seed, SECURITY CONST)

0x67 Postitve Response

Unlocked

0x31 RoutineControl

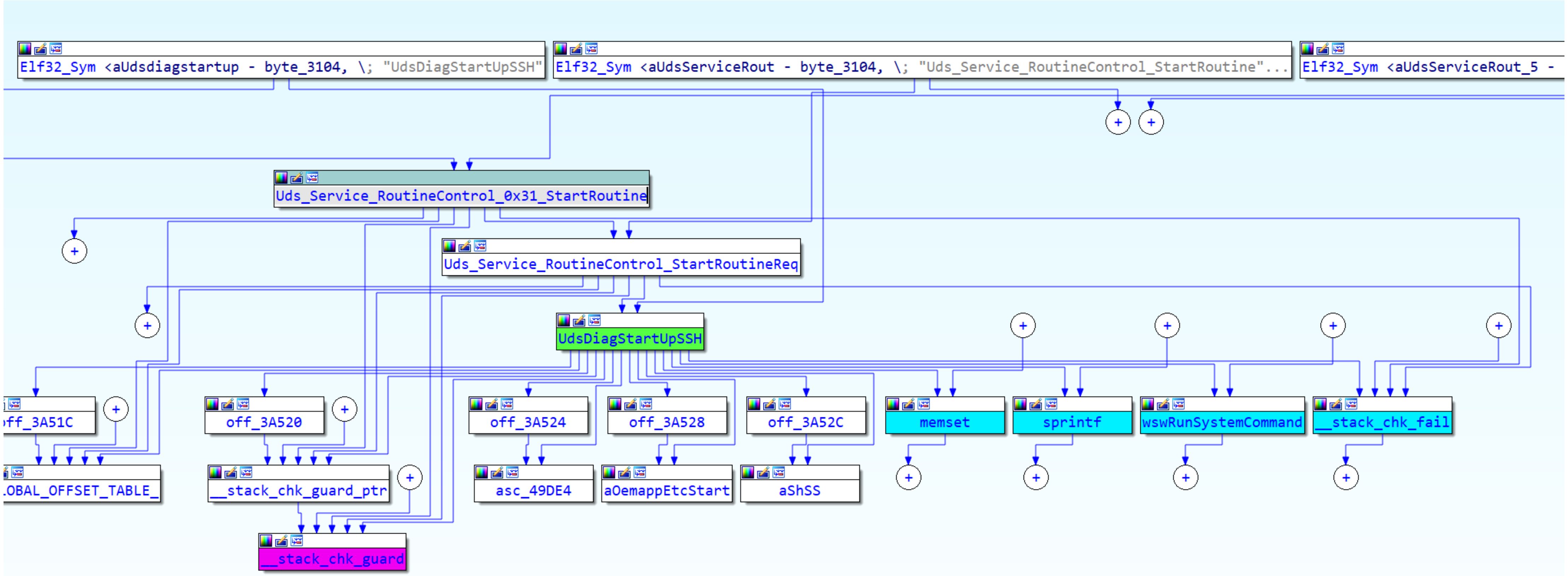
Request UDS security access SECURITY CONST

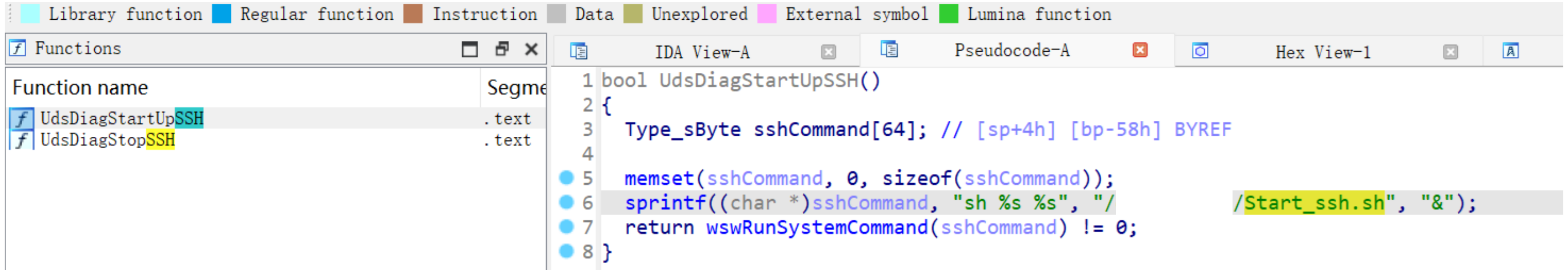
Enabled hidden service

0x71 Postitve Response

Key

UDS RoutineControl: enable SSH


Elf32_Sym <aUdsdiagstartup - byte_3104, \; "UdsDiagStartUpSSH"
Elf32_Sym <aUdsServiceRout - byte_3104, \; "Uds_Service_RoutineControl_StartRoutine"...
Elf32_Sym <aUdsServiceRout_5 -
Uds_Service_RoutineControl_0x31_StartRoutine
Uds_Service_RoutineControl_StartRoutineReq
UdsDiagStartUpSSH
off_3A51C
off_3A520
off_3A524
off_3A528
off_3A52C
memset
sprintf
wswRunSystemCommand
__stack_chk_fail
OBAL_OFFSET_TABLE_
__stack_chk_guard_ptr
asc_49DE4
aOemappEtcStart
aShSS
__stack_chk_guard



Function name	Segme
UdsDiagStartUpSSH	.text
UdsDiagStopSSH	.text

```
bool UdsDiagStartUpSSH()
{
  Type_sByte sshCommand[64]; // [sp+4h] [bp-58h] BYREF

  memset(sshCommand, 0, sizeof(sshCommand));
  sprintf((char *)sshCommand, "sh %s %s", "/          /Start_ssh.sh", "&");
  return wswRunSystemCommand(sshCommand) != 0;
}
```

Unlock hidden serivce

Bypass

Brute Force

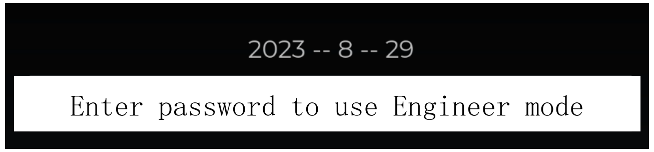
Escape

Deeplink

Bypass

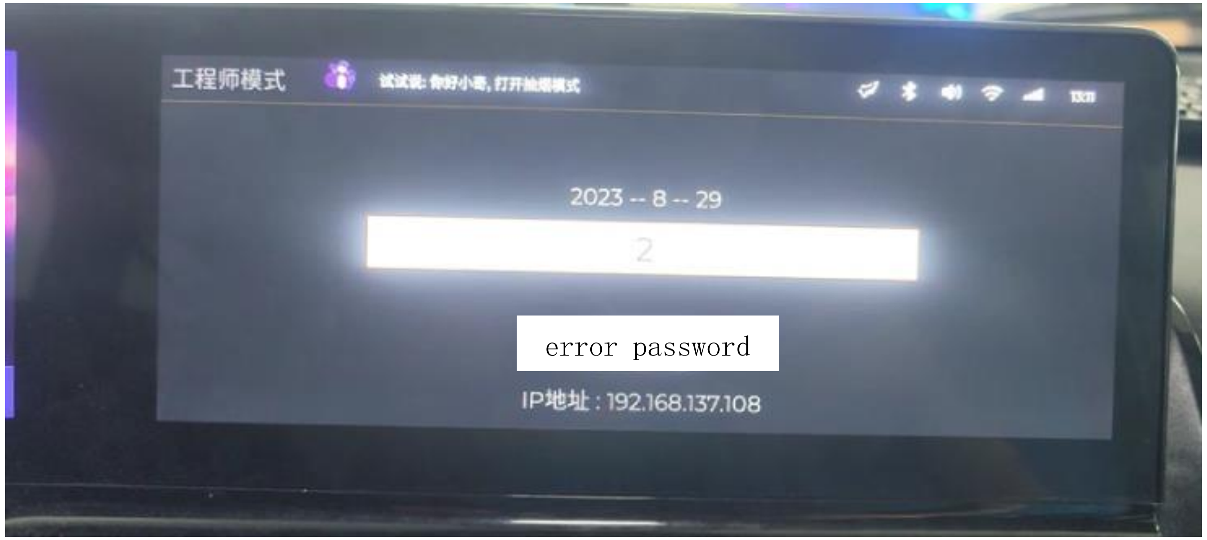
Badusb Brute Force

Entering engineering mode requires password authentication.



2023 -- 8 -- 29

Enter password to use Engineer mode



工程师模式

试试说: 你好小奇, 打开抽烟模式

2023 -- 8 -- 29

2

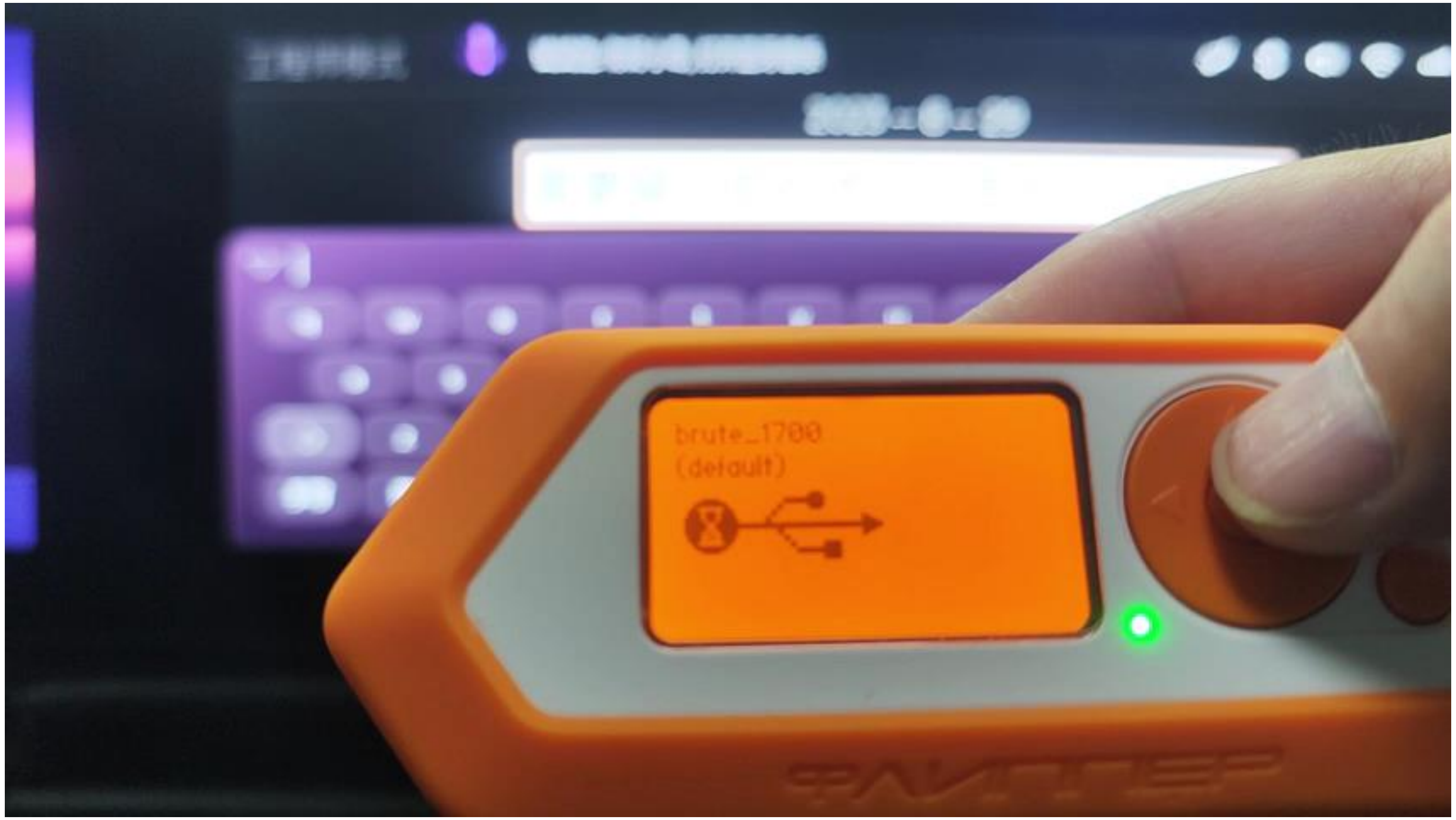
error password

IP地址：192.168.137.108

Bypass

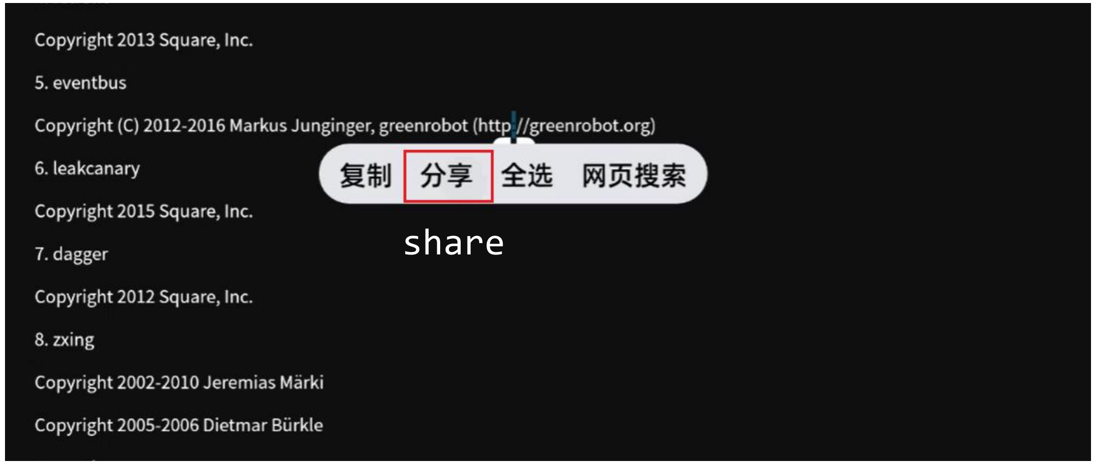
Badusb Brute Force

```
STRING 1800
ENTER
BACKSPACE
BACKSPACE
BACKSPACE
BACKSPACE
BACKSPACE
DELAY 500
STRING 1801
ENTER
BACKSPACE
BACKSPACE
BACKSPACE
BACKSPACE
BACKSPACE
```



Bypass

Escape to native Setting



Copyright 2013 Square, Inc.

5. eventbus

Copyright (C) 2012-2016 Markus Junginger, greenrobot (http://greenrobot.org)

6. leakcanary

Copyright 2015 Square, Inc.

7. dagger

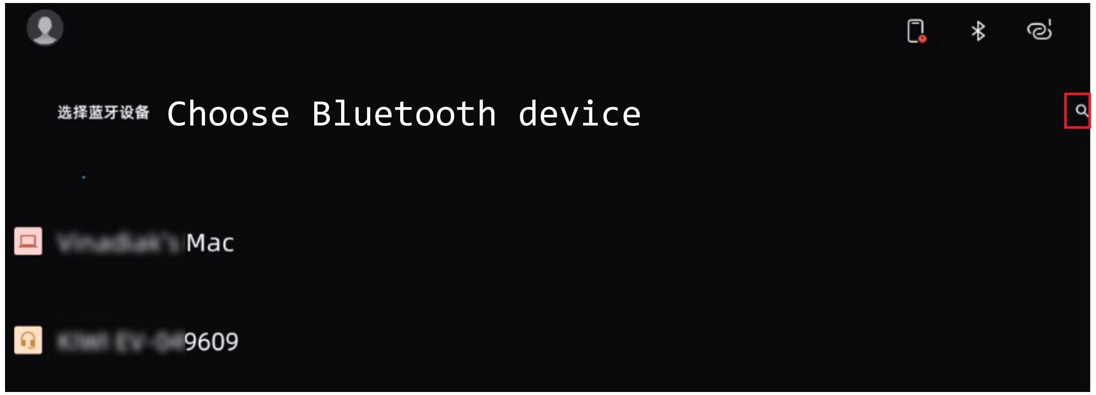
Copyright 2012 Square, Inc.

8. zxing

Copyright 2002-2010 Jeremias Märki

Copyright 2005-2006 Dietmar Bürkle

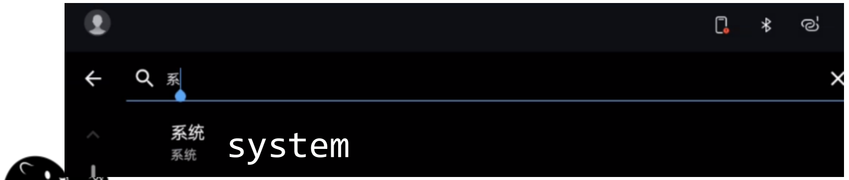
share



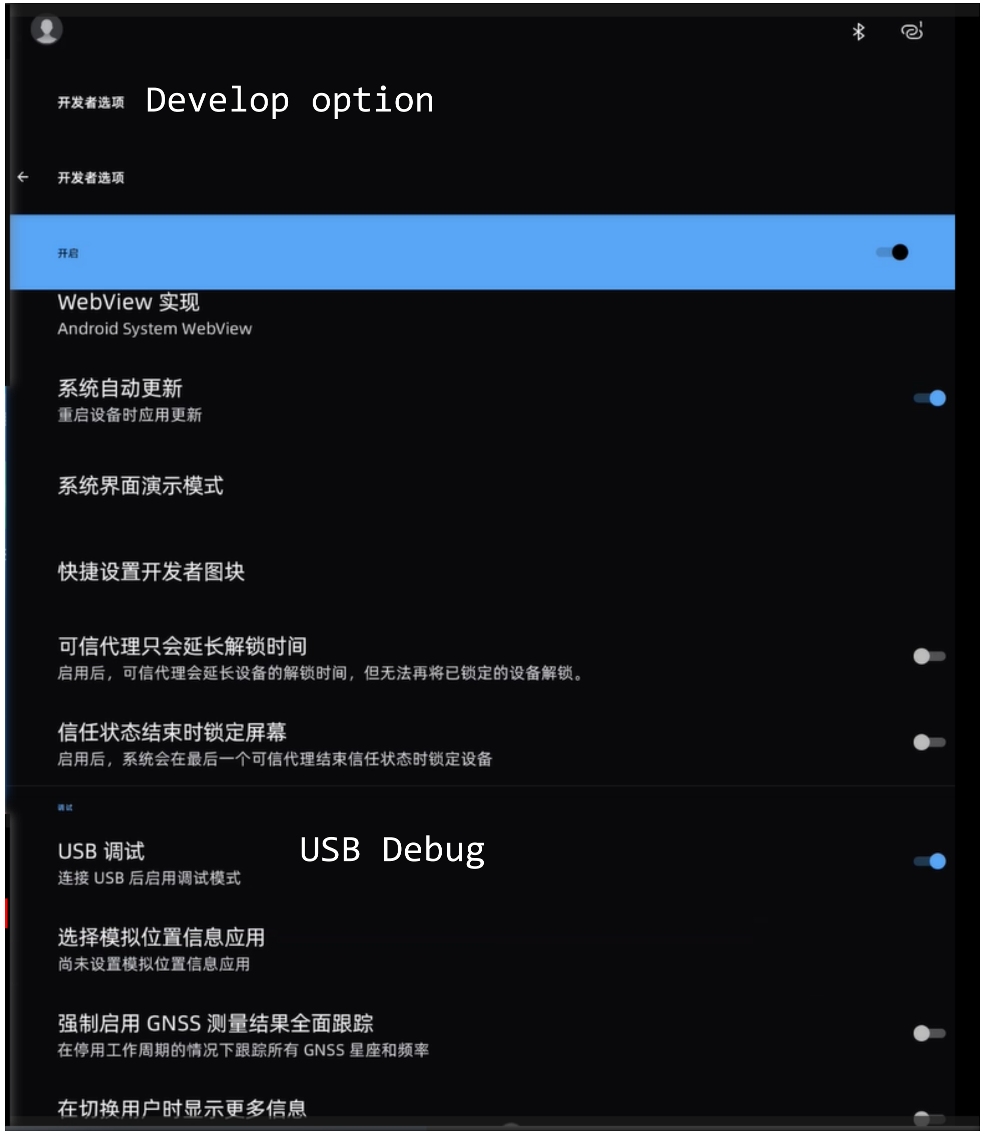
选择蓝牙设备 Choose Bluetooth device

[redacted] Mac

[redacted]9609



系统
系统 system



开发者选项 Develop option

开启

WebView 实现
Android System WebView

系统自动更新
重启设备时应用更新

系统界面演示模式

快捷设置开发者图块

可信代理只会延长解锁时间
启用后，可信代理会延长设备的解锁时间，但无法再将已锁定的设备解锁。

信任状态结束时锁定屏幕
启用后，系统会在最后一个可信代理结束信任状态时锁定设备

调试

USB 调试 USB Debug
连接 USB 后启用调试模式

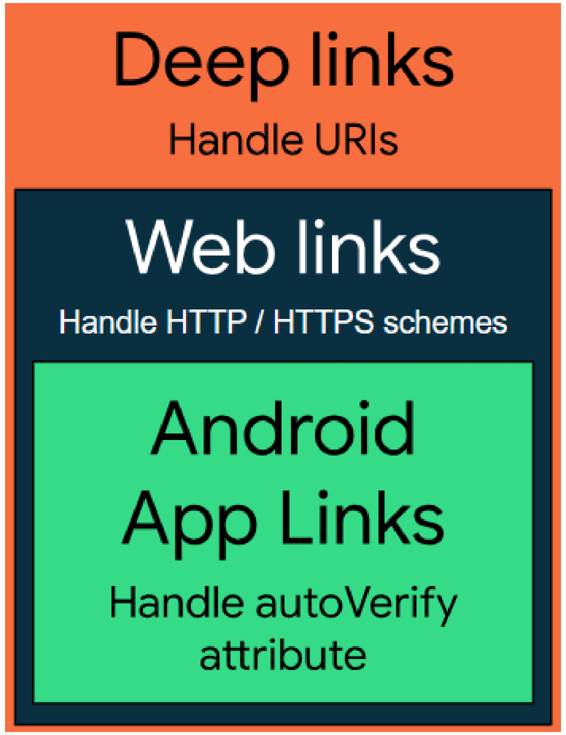
选择模拟位置信息应用
尚未设置模拟位置信息应用

强制启用 GNSS 测量结果全面跟踪
在停用工作周期的情况下跟踪所有 GNSS 星座和频率

在切换用户时显示更多信息

Bypass

Deeplinks



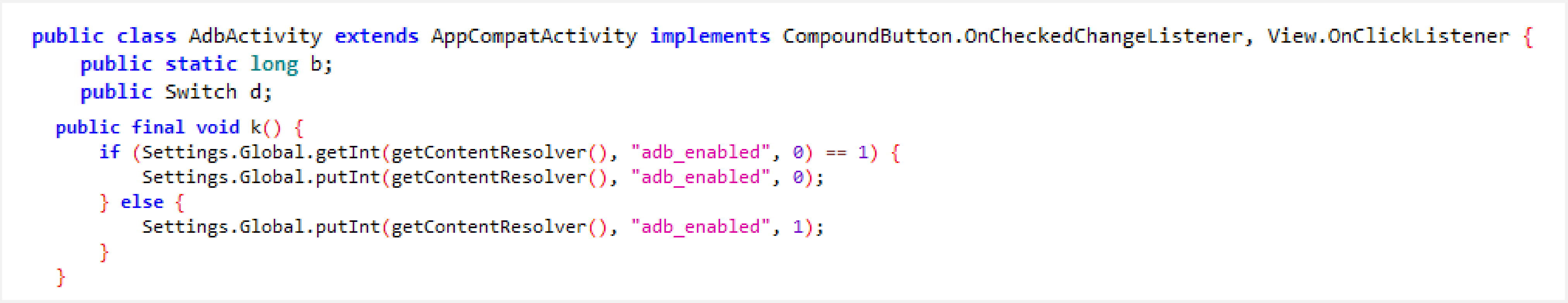
Deep links are URIs of any scheme that take users directly to a specific part of app.

```
<activity
    android:name=".MyMapActivity"
    android:exported="true"
    ...>
    <intent-filter>
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="android.intent.category.BROWSABLE" />
        <data android:scheme="geo" />
    </intent-filter>
</activity>
```

Deeplinks

```
<activity
    android:name="com.xxx.secretcode.activity.AdbActivity"
    android:exported="true"
    android:launchMode="singleTask">
    <intent-filter>
        <action android:name="android.intent.action.VIEW"/>
        <category android:name="android.intent.category.BROWSER"/>
        <category android:name="android.intent.category.DEFAULT"/>
        <data
            android:scheme="headunit"
            android:host="secretcode"
            android:port="9527"
            android:path="/adbactivity"/>
    </intent-filter>
</activity>
```

→ Headunit:/secretcode:9627/adbactivity



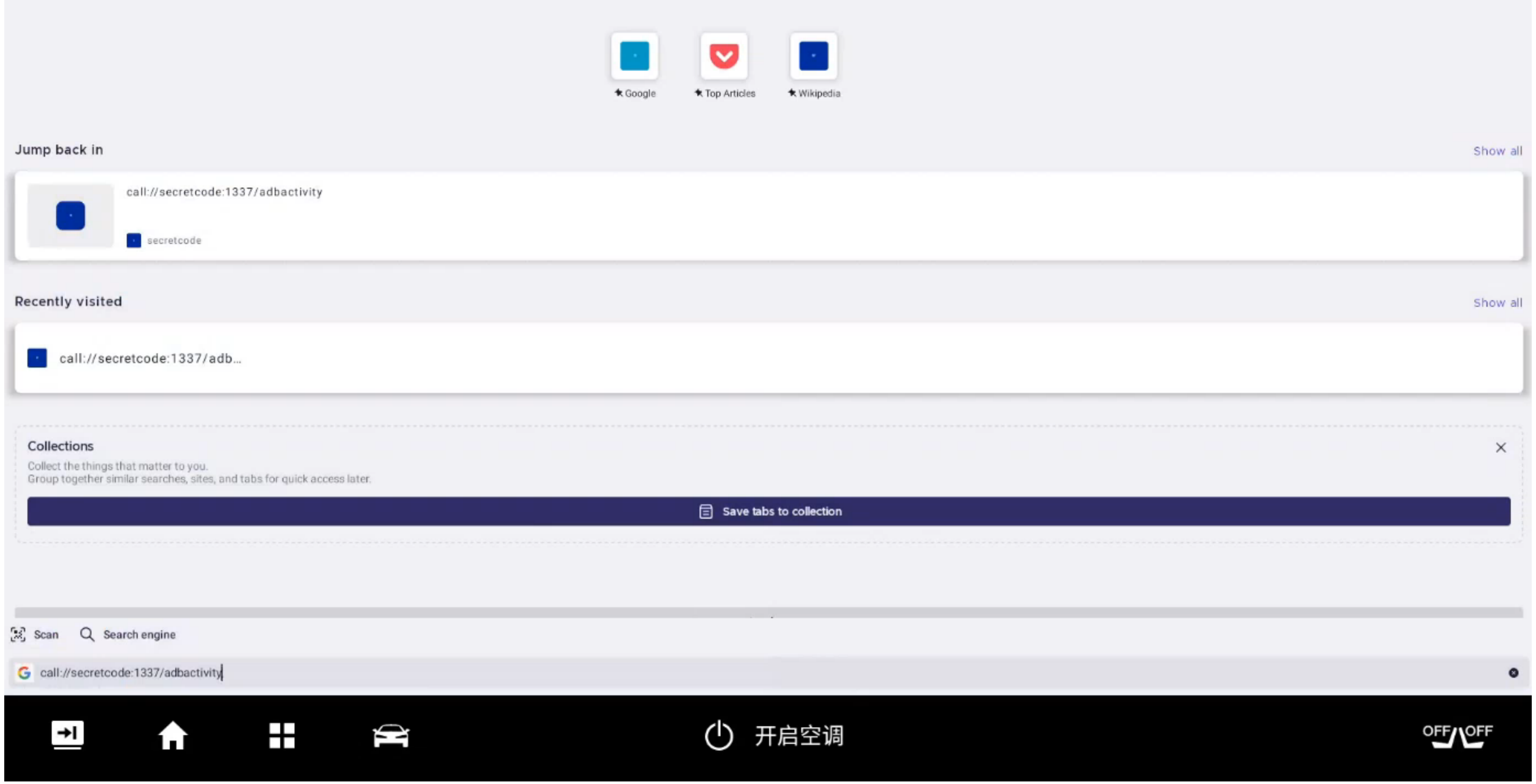
```
public class AdbActivity extends AppCompatActivity implements CompoundButton.OnCheckedChangeListener, View.OnClickListener {
    public static long b;
    public Switch d;

  public final void k() {
      if (Settings.Global.getInt(getContentResolver(), "adb_enabled", 0) == 1) {
          Settings.Global.putInt(getContentResolver(), "adb_enabled", 0);
      } else {
          Settings.Global.putInt(getContentResolver(), "adb_enabled", 1);
      }
  }
```

Bypass

Deeplinks

call://secretcode:1337/adbactivity

android.intent.category.BROWSABLE

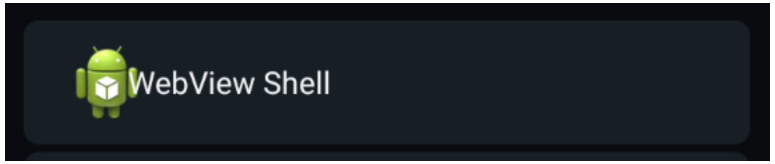


Google
Top Articles
Wikipedia
Jump back in
Show all
call://secretcode:1337/adbactivity
secretcode
Recently visited
Show all
call://secretcode:1337/adb…
Collections
Collect the things that matter to you.
Group together similar searches, sites, and tabs for quick access later.
Save tabs to collection
Scan
Search engine
call://secretcode:1337/adbactivity
开启空调
OFF
OFF

Bypass

Deeplinks

```
call://secretcode:1337/adbactivity

android.intent.category.BROWSABLE
```



call://secretcode:1337/adbactivity

```
ha[redacted]/app
h:[redacted]link://
ar[redacted]//eastereggs
ar[redacted]//platformapi/startApp
ar[redacted]//uploadlog
du[redacted]n-reader://test
du[redacted]n-reader://test.user.collect
ch[redacted]an://[redacted]sentry
ch[redacted]an://[redacted]softmanager
ch[redacted]an://[redacted]travelreminder
ch[redacted]an://[redacted]vehiclesetting
ce[redacted]r://openurl
ce[redacted]r://openwebkit
se[redacted]ngs://[redacted]ngs.slices
hw[redacted]launch[redacted]op_center
x:[redacted]eng://[redacted]acyservice
so[redacted]://com[redacted]ctivity
so[redacted]://com[redacted]tActivity
so[redacted]://com[redacted]ivity
[redacted]933/launcher
[redacted]/mainActivity
[redacted]count
[redacted]countbound
[redacted]login
[redacted]vorite
[redacted]lpcenter
[redacted]mp
[redacted]in
[redacted]account
[redacted]csetting
[redacted]login
[redacted]wechatqrcodelogin
[redacted]ltispeed
[redacted]der
[redacted]derresult
[redacted]onelogin
[redacted]oneqrcord
[redacted]nyinchoosesong
```

FF-BY-ONE 2024

Hidden Service

Discovery

Disabled Activity

property

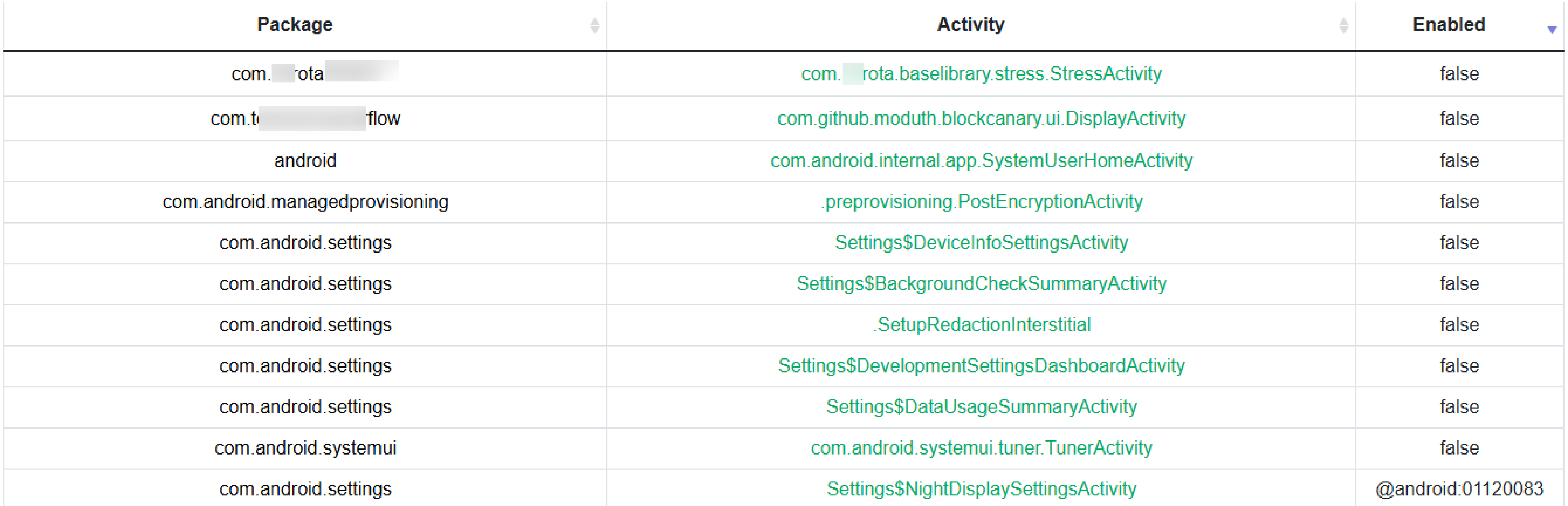
Permission

TEST BIN

Discovery

Disabled Activity

```
<activity android:name=".SecondActivity" android:exported="true" android:label="Droid2" android:enabled="false">
```



Package	Activity	Enabled
com.[redacted]rota	com.[redacted]rota.baselibrary.stress.StressActivity	false
com.t[redacted]flow	com.github.moduth.blockcanary.ui.DisplayActivity	false
android	com.android.internal.app.SystemUserHomeActivity	false
com.android.managedprovisioning	.preprovisioning.PostEncryptionActivity	false
com.android.settings	Settings$DeviceInfoSettingsActivity	false
com.android.settings	Settings$BackgroundCheckSummaryActivity	false
com.android.settings	.SetupRedactionInterstitial	false
com.android.settings	Settings$DevelopmentSettingsDashboardActivity	false
com.android.settings	Settings$DataUsageSummaryActivity	false
com.android.systemui	com.android.systemui.tuner.TunerActivity	false
com.android.settings	Settings$NightDisplaySettingsActivity	@android:01120083

Additional Prop

Android Automotive VehiclePropertyIds(only 129)

```
WINDOW_POS 322964416
WINDOW_MOVE 322964417
WINDOW_LOCK 320867268
VEHICLE_MAP_SERVICE 299895808
OBD2_LIVE_FRAME 299896064
OBD2_FREEZE_FRAME 299896065
OBD2_FREEZE_FRAME_INFO 299896066
OBD2_FREEZE_FRAME_CLEAR 299896067
HEADLIGHTS_STATE 289410560
HIGH_BEAM_LIGHTS_STATE 289410561
FOG_LIGHTS_STATE 289410562
HAZARD_LIGHTS_STATE 289410563
HEADLIGHTS_SWITCH 289410576
HIGH_BEAM_LIGHTS_SWITCH 289410577
FOG_LIGHTS_SWITCH 289410578
HAZARD_LIGHTS_SWITCH 289410579
CABIN_LIGHTS_STATE 289410817
CABIN_LIGHTS_SWITCH 289410818
READING_LIGHTS_STATE 356519683
READING_LIGHTS_SWITCH 356519684
```

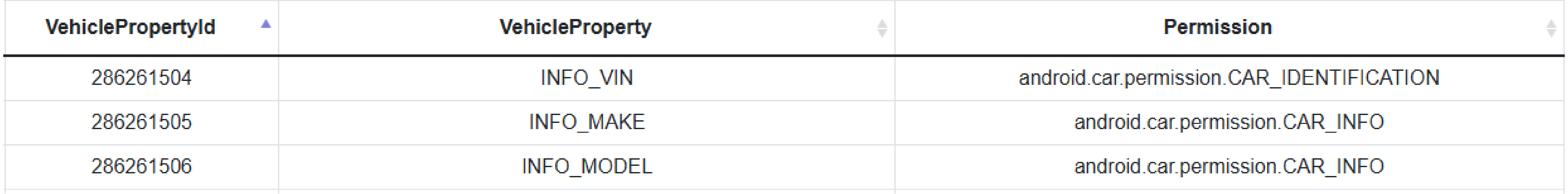
Thousands of VehiclePropertyIds

```
+------------------+-------------+
|       Car        | PropertyNum |
+------------------+-------------+
|    Mxxxda-xxx    |     139     |
|     VW-2xx9      |     114     |
|      xxxNG-P5    |     2530    |
|     xxx-Coffee   |     1282    |
|     SxxxT-HX11   |     114     |
|      CxxY-07     |     190     |
|     Axxi-3875    |     114     |
```

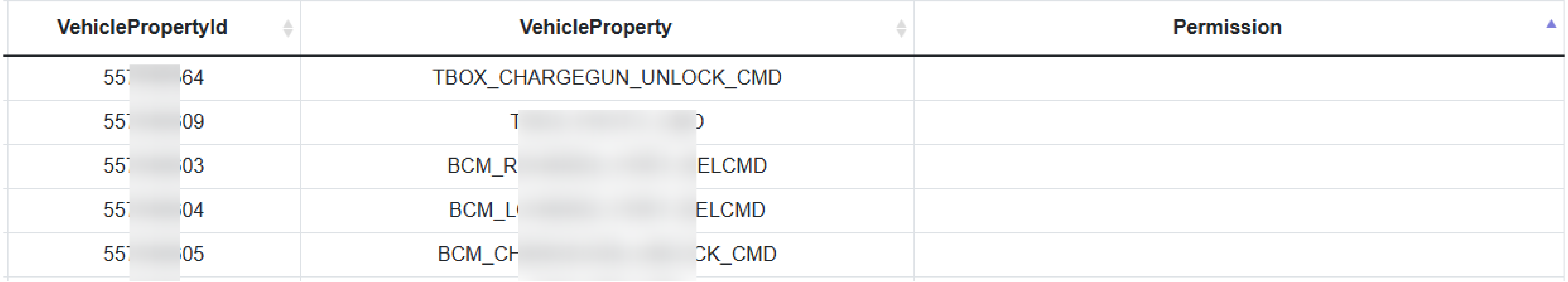
STATUS
CABIN_DOOR_OPEN_STATUS
DATA
UART_DATA
DVR_SSSIDPasswd
Command
MCU_CMD
SEND_RAW_CAN
DIAG_CAN_2E_1E02_WRITE_PKI

Discovery

Prop & Permission



VehiclePropertyId	VehicleProperty	Permission
286261504	INFO_VIN	android.car.permission.CAR_IDENTIFICATION
286261505	INFO_MAKE	android.car.permission.CAR_INFO
286261506	INFO_MODEL	android.car.permission.CAR_INFO



VehiclePropertyId	VehicleProperty	Permission
55[redacted]64	TBOX_CHARGEGUN_UNLOCK_CMD	
55[redacted]09	T[redacted]D	
55[redacted]03	BCM_R[redacted]ELCMD	
55[redacted]04	BCM_L[redacted]ELCMD	
55[redacted]05	BCM_CH[redacted]CK_CMD	

```
TBOX_CHARGEGUN_UNLOCK_CMD
public void setBooleanProperty (int propertyId, int areaId, boolean val)
setBooleanProperty(55123464,0,1)
```

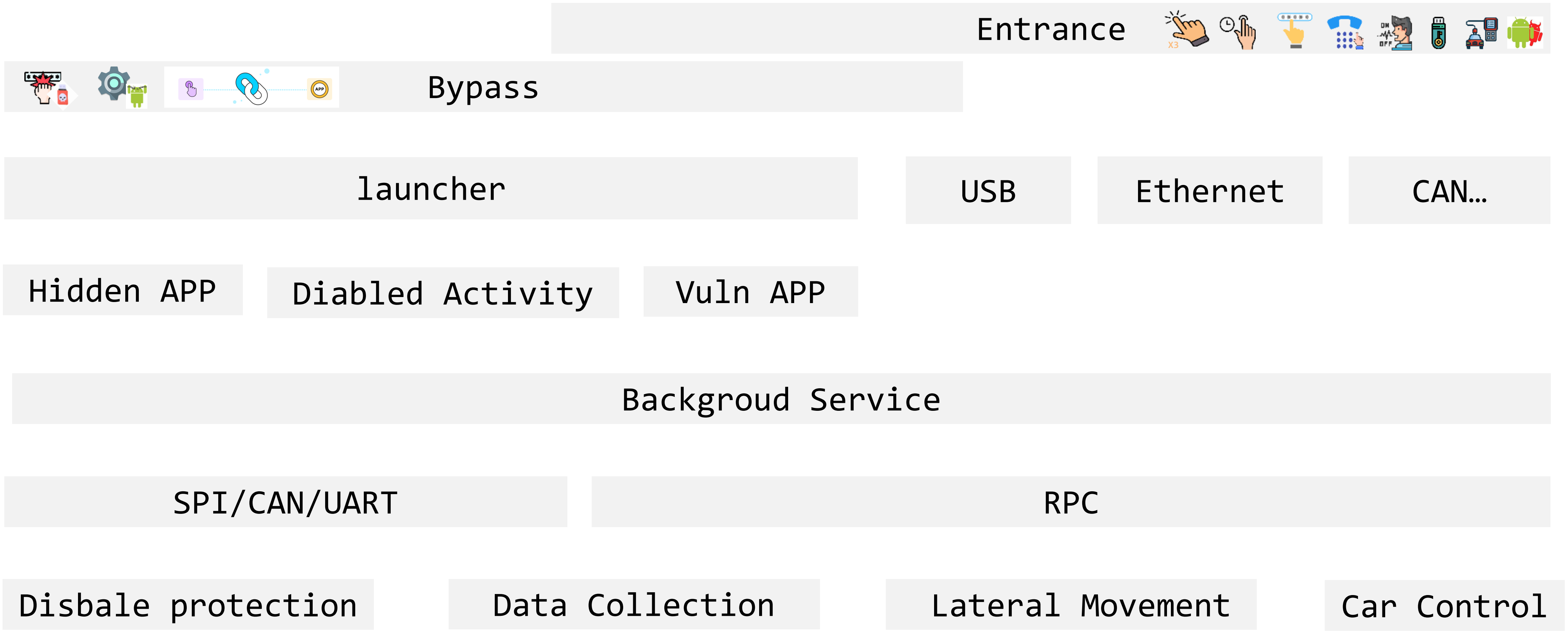
Test binnary

```
car:/ $ vehicle-hal-tool
./vehicle-hal-tool [-l] [-m -p -t [-v]]
-l - List properties
-m - Mode (cannot be used with -l). Accepted strings: get, set or sub.
-p - Property (only used with -m)
-t - Type (only used with -m)
-w - Wait time in seconds (only used with -m set to sub)
-v - Value to which vehicle_prop_value is set
./vehicle-hal-tool -m set -p 10 -t 1 -v random_property
```

```
car:/ $ oem_car
Car service commands:
  -h,help
    Print this help text.
  get <propId> [areaId]
    get a car property value by propId and areaId
  -e,event <propId> [areaId] <value>
    Inject a car property for testing.
    such as: -e Climate#ID_TEMPERATURE 0x10 30.5
  error-event <propId> [areaId] <value>
    Inject an error event for testing.
    such as: error-event Climate#ID_TEMPERATURE 0x10 30.5
  set <propId> [areaId] <value>
    send a car property for testing.
    such as: set 33554442 0x40 28
```

UNLOCK

Summary



Entrance

Bypass

launcher

USB

Ethernet

CAN…

Hidden APP

Diabled Activity

Vuln APP

Backgroud Service

SPI/CAN/UART

RPC

Disbale protection

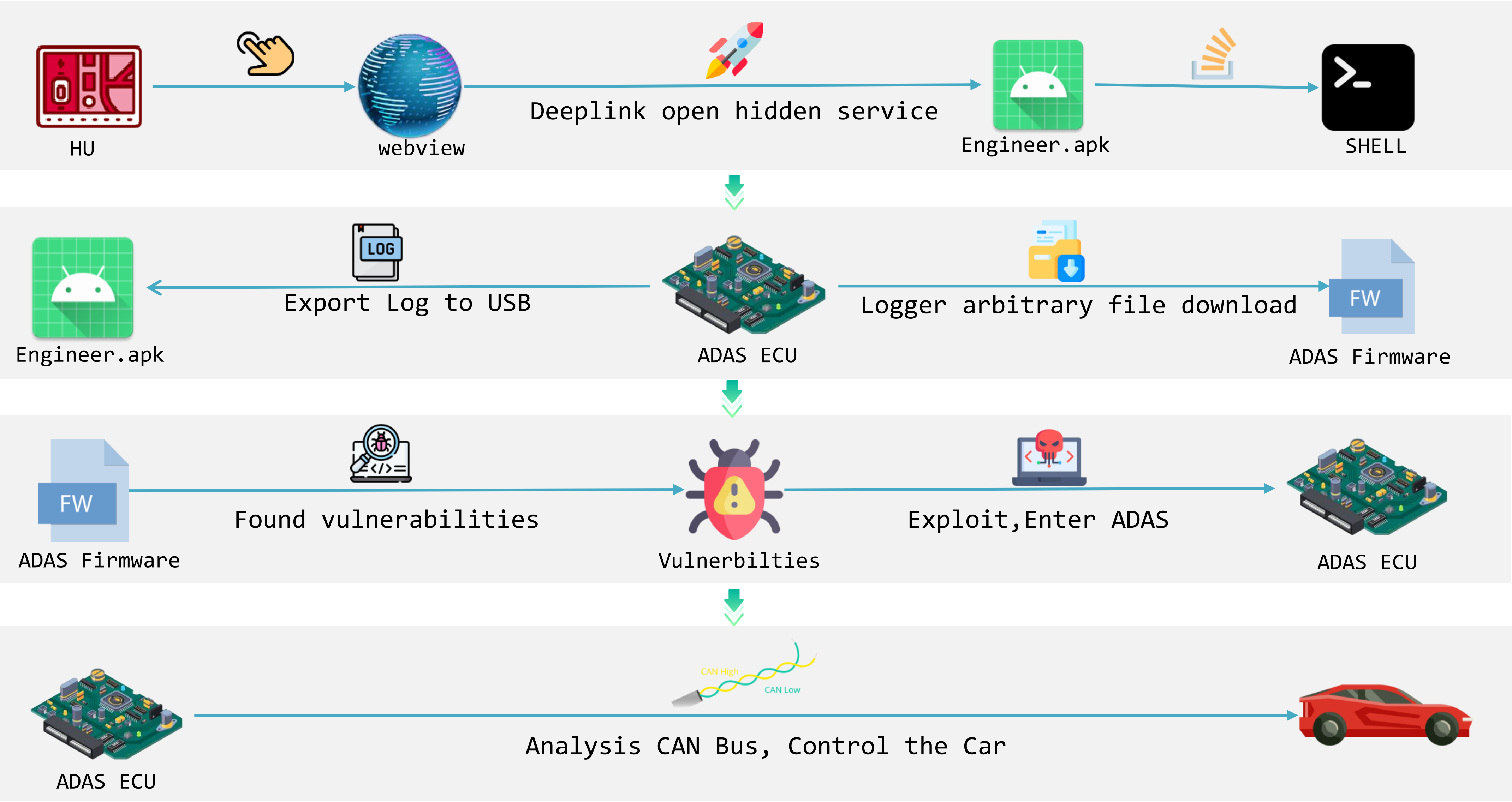
Data Collection

Lateral Movement

Car Control

CASE

From unlock to control car



HU

webview

Deeplink open hidden service

Engineer.apk

SHELL

Engineer.apk

Export Log to USB

ADAS ECU

Logger arbitrary file download

ADAS Firmware

ADAS Firmware

Found vulnerabilities

Vulnerbilties

Exploit,Enter ADAS

ADAS ECU

ADAS ECU

Analysis CAN Bus, Control the Car

Tool

Relation

APP

Activities

services

Depends

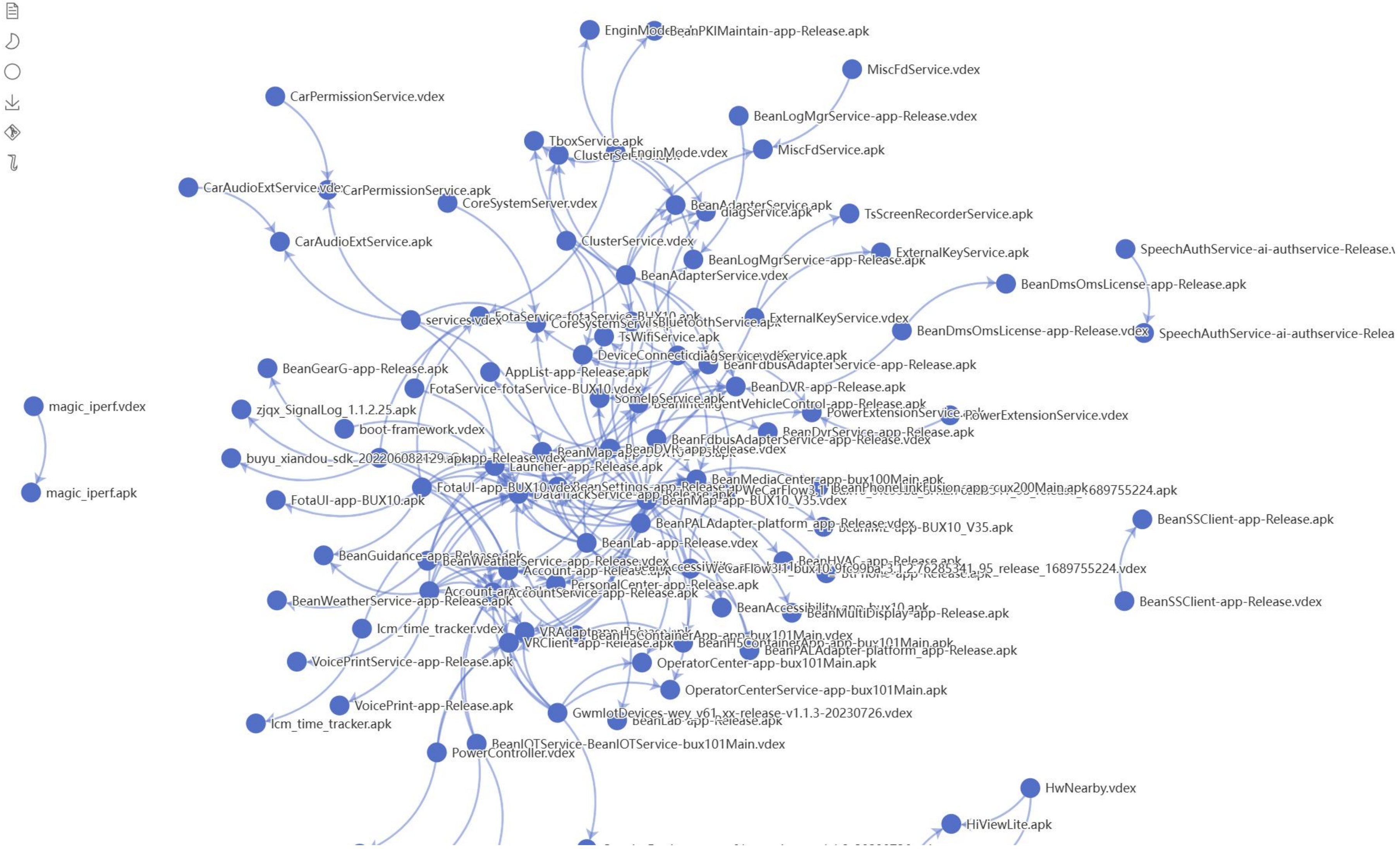
IPC

RPC

Tool

Relation

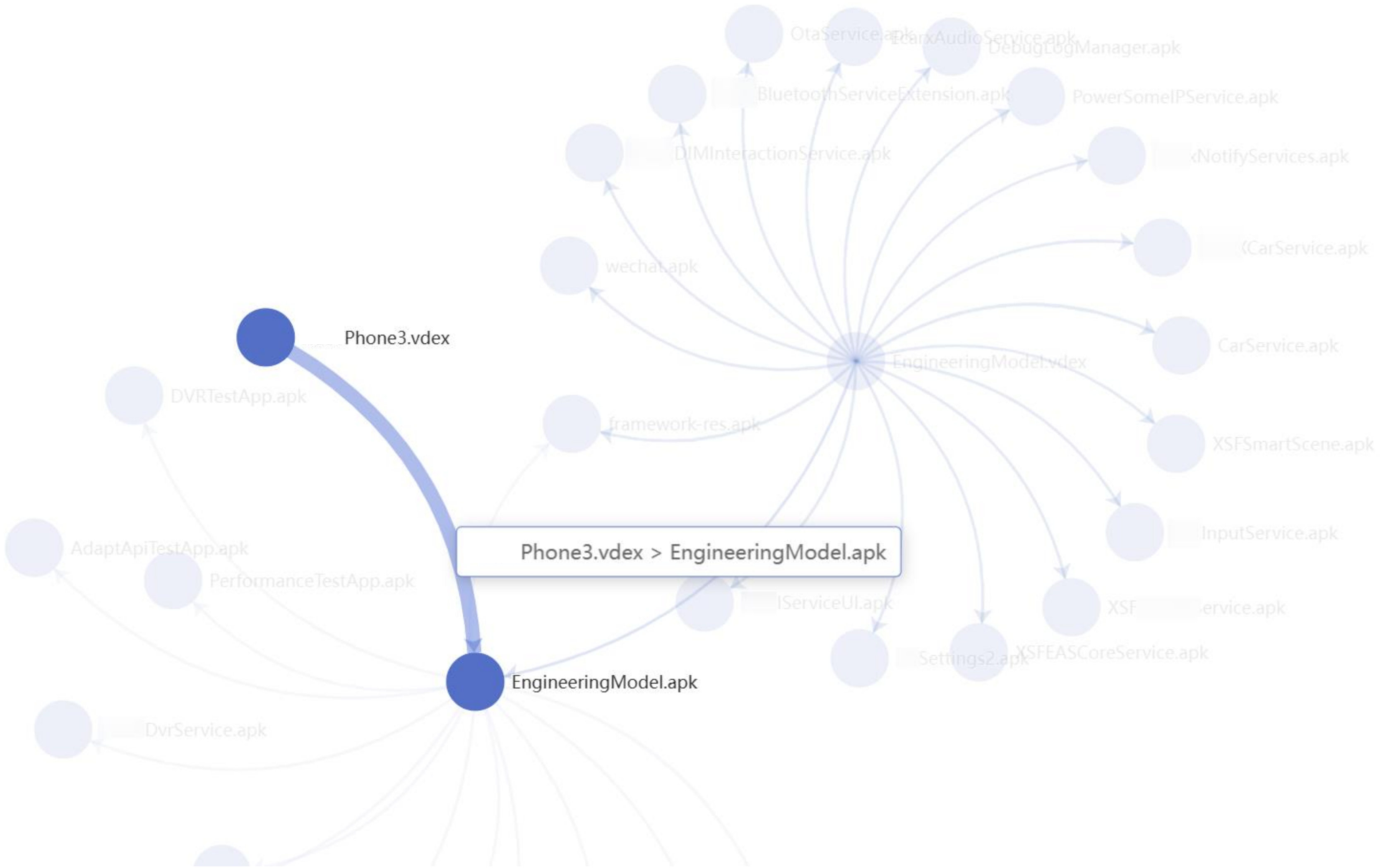
- **Apk Relation**
  - /system/app
  - /system/priv-app
  - /data/app/
  - /vendor/app
  - /product/app
  - /product/priv-app
  - /product/overlay


EnginModeBeanPKIMaintain-app-Release.apk
MiscFdService.vdex
CarPermissionService.vdex
BeanLogMgrService-app-Release.vdex
TboxService.apk
ClusterSer EnginMode.vdex
MiscFdService.apk
CarAudioExtService.vdex
CarPermissionService.apk
CoreSystemServer.vdex
BeanAdapterService.apk
diagService.apk
TsScreenRecorderService.apk
CarAudioExtService.apk
ClusterService.vdex
BeanLogMgrService-app-Release.apk
ExternalKeyService.apk
SpeechAuthService-ai-authservice-Release.
BeanAdapterService.vdex
BeanDmsOmsLicense-app-Release.apk
services.vdex
FotaService-fotaService-BUX10.apk
CoreSystemServisBluetoothService.apk
ExternalKeyService.vdex
BeanDmsOmsLicense-app-Release.vdex
SpeechAuthService-ai-authservice-Relea
TsWifiService.apk
DeviceConnectiodiagServicevdexService.apk
BeanFdbusAdapterService-app-Release.apk
BeanGearG-app-Release.apk
AppList-app-Release.apk
magic_iperf.vdex
FotaService-fotaService-BUX10.vdex
BeanDVR-app-Release.apk
SomeIpService.apk
BeanIntelligentVehicleControl-app-Release.apk
zjqx_SignalLog_1.1.2.25.apk
PowerExtensionService.apk
PowerExtensionService.vdex
boot-framework.vdex
BeanDvrService-app-Release.apk
BeanFdbusAdapterService-app-Release.vdex
BeanMap-app BeanDVR-app-Release.vdex
buyu_xiandou_sdk_20220608129.apkapp-Release.vdex
Launcher-app-Release.apk
BeanMediaCenter-app-bux100Main.apk
FotaUI-app-BUX10.vdex BeanSettings-app-Release.apk WeCarFlow3.1 BeanPhoneLinkFusion-app-cux200Main.apk 689755224.apk
magic_iperf.apk
DataTrackService-app-Release.apk
FotaUI-app-BUX10.apk
BeanMap-app-BUX10_V35.vdex
BeanPALAdapter-platform_app-Release.vdex
BeanMap-app-BUX10_V35.apk
BeanSSClient-app-Release.apk
BeanLab-app-Release.vdex
BeanGuidance-app-Release.apk
BeanWeatherService-app-Release.vdex
BeanHVAC-app-Release.apk
Account-app-Release.apk BeanAccessi WeCarFlow3.1 bux10_9fc99ba_3.1.2.76285341_95_release_1689755224.vdex
PersonalCenter-app-Release.apk
Account-ar AccountService-app-Release.apk
BeanSSClient-app-Release.vdex
BeanWeatherService-app-Release.apk
BeanAccessibility-app-bux10.apk
BeanMultiDisplay-app-Release.apk
Icm_time_tracker.vdex
VRAdapter BeanH5ContainerApp-app-bux101Main.vdex
VRClient-app-Release.apk
BeanH5ContainerApp-app-bux101Main.apk
BeanPALAdapter-platform_app-Release.apk
VoicePrintService-app-Release.apk
OperatorCenter-app-bux101Main.apk
OperatorCenterService-app-bux101Main.apk
VoicePrint-app-Release.apk
GwmIotDevices-wey_v61_xx-release-v1.1.3-20230726.vdex
Icm_time_tracker.apk
BeanLab-app-Release.apk
BeanIOTService-BeanIOTService-bux101Main.vdex
PowerController.vdex
HwNearby.vdex
HiViewLite.apk

Tool

Relation

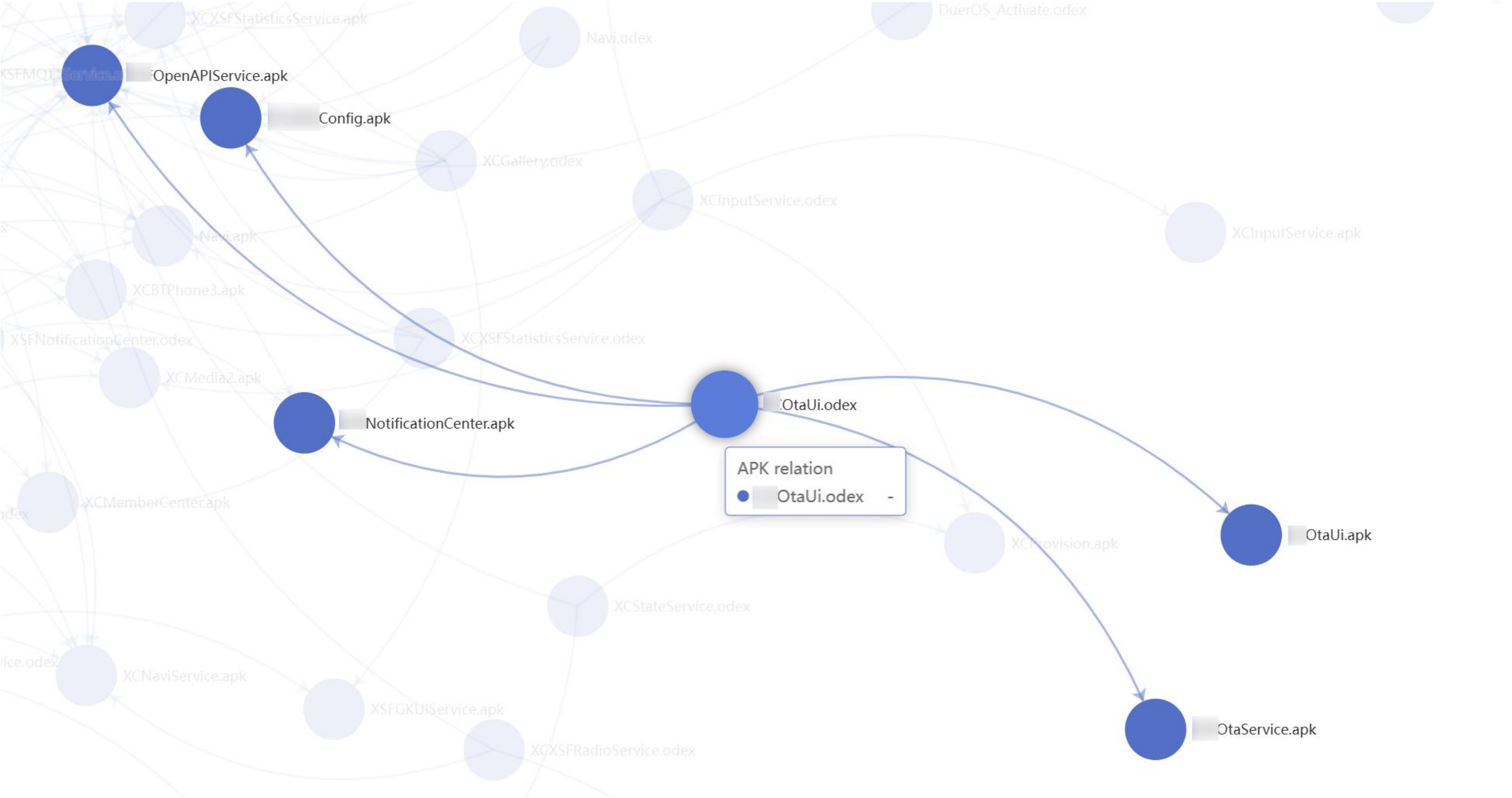
Use diagram, find how to start engineermode application



Tool

Relation

According diagram, find applications related to business,such as OTA



FF-BY-ONE 2024

Project: https://github.com/delikely/ERH