Test security and privacy of your mobile application (iOS & Android), detect OWASP Mobile Top 10 and other weaknesses.

Summary of Mobile Application Security Test

APP NAME
GreenPass

APP ID
com.italinnovation.green_pass

APP VERSION
2.0.1

DEVICE TYPE
Android

TEST STARTED
May 12th 2022, 13:45

TEST FINISHED
May 12th 2022, 14:14

Mobile App Permissions and Privacy	OWASP Mobile Top 10 Security Test	Mobile App External Communications	Software Composition Analysis
7 PERMISSIONS	3 MAJOR RISKS FOUND	NOT FOUND	8 COMPONENTS FOUND

Malware test: no malicious code or behavioral patterns detected in the mobile app.

Mobile Application Permissions and Privacy Test

If the application processes or stores any PII of EU residents, the following requirements of EU GDPR may apply:

Privacy Policy

Privacy Policy was not found in application — Misconfiguration or weakness

Mobile Application Functionality

The mobile application requests access to the following functionality that may endanger user's privacy under certain circumstances:

Camera

The mobile application can use the camera for taking pictures or videos.

Mobile Application Permissions

The mobile application requests the following permissions that may endanger user's privacy under certain circumstances:

CAMERA dangerous

Allows application to take pictures and videos with the camera. This allows the application to collect images that the camera is seeing at any time.

ACCESS_NETWORK_STATE normal

Allows an application to view the status of all networks.

FOREGROUND_SERVICE normal

Allows a regular application to use Service.startForeground.

INTERNET normal

Allows an application to create network sockets.

QUERY_ALL_PACKAGES normal

Allows query of any normal app on the device, regardless of manifest declarations.

RECEIVE_BOOT_COMPLETED normal

Allows an application to start itself as soon as the system has finished booting. This can make it take longer to start the phone and allow the application to slow down the overall phone by always running.

WAKE_LOCK normal

Allows an application to prevent the phone from going to sleep.

OWASP Mobile Top 10 Security Test

The automated audit revealed the following security flaws and weaknesses that may impact the application:

HIGH RISKS	MEDIUM RISK	LOW RISKS	WARNINGS
2	1	3	7

Zero false-positive SLA and advanced manual testing of application is only available in ImmuniWeb® MobileSuite.

EXTERNAL DATA IN SQL QUERIES [M7] [CWE-89] [SAST]

HIGH

Description:

Inclusion of input into raw SQL queries can potentially lead to a local SQL injection vulnerability in the mobile application. The correct approach is to use prepared SQL statements beyond user's control.

Example of insecure code:

```
db.rawQuery("SELECT username FROM users_table WHERE id = '"+ input_id +"'");
db.execSQL("SELECT username FROM users_table WHERE id = '"+ input_id +"'");
```

Example of secure code:

```
PreparedStatement pstmt = con.prepareStatement("UPDATE EMPLOYEES SET SALARY = ? WHERE
ID = ?");
pstmt.setBigDecimal(1, 153833.00)
pstmt.setInt(2, 110592)
```

Details:

There is **'rawQuery()'** found in file 'd/b/b/a/i/x/j/h0.java':

```
[line 85:         this.m = true;]
[line 86:         sQLiteDatabase.rawQuery("PRAGMA busy_timeout=0;", new
String[0]).close();]
[line 87:         if (VERSION.SDK_INT >= 16) {]
```

CVSSv3 Base Score:

7.3 (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:L)

Reference:

- https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase.html
- https://developer.android.com/reference/java/sql/PreparedStatement.html

CLEARTEXT SQLITE DATABASE [M2] [CWE-312] [DAST]

HIGH

Description:

The mobile application uses an unencrypted SQLite database.
This database can be accessed by an attacker with physical access to the mobile device or a malicious application with root access to the device. The application should not store sensitive information in clear text.

Details:

In file com.google.android.datatransport.events:

```
TABLES:
android_metadata
events
event_metadata
transport_contexts
events_backend_id
contexts_backend_priority_extras
event_payloads
sqlite_autoindex_event_payloads_1

RAW DUMP:
CREATE TABLE android_metadata (locale TEXT);CREATE TABLE events (_id INTEGER PRIMARY
KEY, context_id INTEGER NOT NULL, transport_name TEXT NOT NULL, timestamp_ms INTEGER
NOT NULL, uptime_ms INTEGER NOT NULL, payload BLOB NOT NULL, code INTEGER, num_attempts
INTEGER NOT NULL, payload_encoding TEXT, inline BOOLEAN NOT NULL DEFAULT 1,FOREIGN KEY
(context_id) REFERENCES transport_contexts(_id) ON DELETE CASCADE);CREATE TABLE
event_metadata (_id INTEGER PRIMARY KEY, event_id INTEGER NOT NULL, name TEXT NOT NULL,
value TEXT NOT NULL,FOREIGN KEY (event_id) REFERENCES events(_id) ON DELETE
CASCADE);CREATE TABLE transport_contexts (_id INTEGER PRIMARY KEY, backend_name TEXT
NOT NULL, priority INTEGER NOT NULL, next_request_ms INTEGER NOT NULL, extras
BLOB);CREATE INDEX events_backend_id on events(context_id);CREATE UNIQUE INDEX
contexts_backend_priority_extras on transport_contexts(backend_name, priority,
extras);CREATE TABLE event_payloads (sequence_num INTEGER NOT NULL, event_id INTEGER
NOT NULL, bytes BLOB NOT NULL,FOREIGN KEY (event_id) REFERENCES events(_id) ON DELETE
CASCADE,PRIMARY KEY (sequence_num, event_id));
```

In file androidx.work.workdb:

```
TABLES:
android_metadata
Dependency
sqlite_autoindex_Dependency_1
index_Dependency_work_spec_id
index_Dependency_prerequisite_id
WorkSpec
sqlite_autoindex_WorkSpec_1
index_WorkSpec_schedule_requested_at
index_WorkSpec_period_start_time
WorkTag
sqlite_autoindex_WorkTag_1
index_WorkTag_work_spec_id
SystemIdInfo
sqlite_autoindex_SystemIdInfo_1
WorkName
sqlite_autoindex_WorkName_1
index_WorkName_work_spec_id
WorkProgress
sqlite_autoindex_WorkProgress_1
Preference
sqlite_autoindex_Preference_1
room_master_table

RAW DUMP:
CREATE TABLE android_metadata (locale TEXT);CREATE TABLE `Dependency` (`work_spec_id`
TEXT NOT NULL, `prerequisite_id` TEXT NOT NULL, PRIMARY KEY(`work_spec_id`,
`prerequisite_id`), FOREIGN KEY(`work_spec_id`) REFERENCES `WorkSpec`(`id`) ON UPDATE
CASCADE ON DELETE CASCADE , FOREIGN KEY(`prerequisite_id`) REFERENCES `WorkSpec`(`id`)
```

```
CASCADE ON DELETE CASCADE , FOREIGN KEY( `prerequisite_id` ) REFERENCES `WorkSpec` ( `id`
ON UPDATE CASCADE ON DELETE CASCADE );CREATE INDEX `index_Dependency_work_spec_id` ON
`Dependency` (`work_spec_id`);CREATE INDEX `index_Dependency_prerequisite_id` ON
`Dependency` (`prerequisite_id`);CREATE TABLE `WorkSpec` (`id` TEXT NOT NULL, `state`
INTEGER NOT NULL, `worker_class_name` TEXT NOT NULL, `input_merger_class_name` TEXT,
`input` BLOB NOT NULL, `output` BLOB NOT NULL, `initial_delay` INTEGER NOT NULL,
`interval_duration` INTEGER NOT NULL, `flex_duration` INTEGER NOT NULL,
`run_attempt_count` INTEGER NOT NULL, `backoff_policy` INTEGER NOT NULL,
`backoff_delay_duration` INTEGER NOT NULL, `period_start_time` INTEGER NOT NULL,
`minimum_retention_duration` INTEGER NOT NULL, `schedule_requested_at` INTEGER NOT
NULL, `run_in_foreground` INTEGER NOT NULL, `out_of_quota_policy` INTEGER NOT NULL,
`required_network_type` INTEGER, `requires_charging` INTEGER NOT NULL,
`requires_device_idle` INTEGER NOT NULL, `requires_battery_not_low` INTEGER NOT NULL,
`requires_storage_not_low` INTEGER NOT NULL, `trigger_content_update_delay` INTEGER NOT
NULL, `trigger_max_content_delay` INTEGER NOT NULL, `content_uri_triggers` BLOB,
PRIMARY KEY(`id`));CREATE INDEX `index_WorkSpec_schedule_requested_at` ON `WorkSpec`
(`schedule_requested_at`);CREATE INDEX `index_WorkSpec_period_start_time` ON `WorkSpec`
(`period_start_time`);CREATE TABLE `WorkTag` (`tag` TEXT NOT NULL, `work_spec_id` TEXT
NOT NULL, PRIMARY KEY(`tag`, `work_spec_id`), FOREIGN KEY(`work_spec_id`) REFERENCES
`WorkSpec`(`id`) ON UPDATE CASCADE ON DELETE CASCADE );CREATE INDEX
`index_WorkTag_work_spec_id` ON `WorkTag` (`work_spec_id`);CREATE TABLE `SystemIdInfo`
(`work_spec_id` TEXT NOT NULL, `system_id` INTEGER NOT NULL, PRIMARY
KEY(`work_spec_id`), FOREIGN KEY(`work_spec_id`) REFERENCES `WorkSpec`(`id`) ON UPDATE
CASCADE ON DELETE CASCADE );CREATE TABLE `WorkName` (`name` TEXT NOT NULL,
`work_spec_id` TEXT NOT NULL, PRIMARY KEY(`name`, `work_spec_id`), FOREIGN
KEY(`work_spec_id`) REFERENCES `WorkSpec`(`id`) ON UPDATE CASCADE ON DELETE CASCADE
);CREATE INDEX `index_WorkName_work_spec_id` ON `WorkName` (`work_spec_id`);CREATE
TABLE `WorkProgress` (`work_spec_id` TEXT NOT NULL, `progress` BLOB NOT NULL, PRIMARY
KEY(`work_spec_id`), FOREIGN KEY(`work_spec_id`) REFERENCES `WorkSpec`(`id`) ON UPDATE
CASCADE ON DELETE CASCADE );CREATE TABLE `Preference` (`key` TEXT NOT NULL,
`long_value` INTEGER, PRIMARY KEY(`key`));CREATE TABLE room_master_table (id INTEGER
PRIMARY KEY,identity_hash TEXT);
```

CVSSv3 Base Score:

7.1 (AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N)

JS ENABLED IN A WEBVIEW [M10] [CWE-749] [SAST]

MEDIUM

Description:

The mobile application has enabled JavaScript in WebView. By default, JavaScript is disabled in WebView, if enabled it can bring various JS-related security issues, such as Cross-Site Scripting (XSS) attacks.

Example of insecure code:

```
WebSettings settings = webView.getSettings();
settings.setJavaScriptEnabled(true);
```

Example of secure code:

```
// Don't enable Javascript in WebView
```

Details:

There is **'setJavaScriptEnabled()'** found in file 'io/flutter/plugins/urllauncher/WebViewActivity.java':

```
[line 123:         this.l.getSettings().setJavaScriptEnabled(booleanExtra);]
```

CVSSv3 Base Score:

4.8 (AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:L/A:N)

Reference:

- https://developer.android.com/reference/android/webkit/WebView.html
- https://developer.android.com/reference/android/webkit/WebSettings.html

HARDCODED DATA [M2] [CWE-200] [SAST]

LOW

Description:

The mobile application contains debugging or other technical information that may be extracted and used by an attacker to facilitate further attacks.

https:// with value https://github.com/flutter/flutter/issues/2897 in following files:

- **io/flutter/plugin/platform/l.java:**

```
[line 409: stringBuilder.append("] may result in
problems(https://github.com/flutter/flutter/issues/2897).It is larger than
the device screen size: [");]
```

https:// with value https://firebaseinstallations.googleapis.com/v1 in following files:

- **d/b/b/b/e/n/dt.java:**

```
[line 28: this.g = "https://firebaseinstallations.googleapis.com/v1";]
```

- **d/b/b/b/e/z/oc.java:**

```
[line 15: private final String f =
"https://firebaseinstallations.googleapis.com/v1";]
```

- **d/b/b/b/e/m/ka.java:**

```
[line 19: private final String f =
"https://firebaseinstallations.googleapis.com/v1";]
```

https:// with value https://goo.gl/J1sWQy in following files:

- **d/b/b/b/e/g/g0.java:**

```
[line 64: Log.v(this.b, "Deferring to Google Analytics for Firebase for
event data collection. https://goo.gl/J1sWQy");]
```

chrome:// with value chrome://global/locale/intl.properties in following files:

- **d/b/b/b/e/r/it.java:**

```
[line 378: no-NO,mk,vi-VN,el-gr,es-co,ka,nb,nn,es-ve,es-CL,ro-RO,is,he-
IL,hr-HR,fil,ms,sq,te,vi-vn,ml-IN,de-ch,de-at,es-cl,ro-ro,id-ID,ml,en-NZ,fr-
be,br,nl-nl,en-nz,es-PE,nb-NO,lt-LT,th-th,ta,bs,az,sw,mr,be,af,uk-UA,en-
ie,oc,es-VE,hr-hr,kr,tg-IN,ga,kn,gu-IN,gbk,lb,io,kn-IN,zh-Hans,en-
ZA,my,jv,win,en-za,ar-EG,zh-HK,mn,be-TARASK,et-EE,zh-hk,sl-SI,la,sr-RS,fr-
ch,*,fr-CH,en-in,hy,hi-IN,qu,en-SG,si,ht,new,lv-LV,ps,ur,mt,es-
us,vo,gu,fo,es-EC,bn,en-us.,ak,us,mi-NZ,fa-IR,es-cr,ne,ab,en-sg,tt,es-US,en-
MY,es-ec,da-dk,ar-ae,mi,su,ar-jo,ca-ES,kk,ar-kw,en-UK,km,ko-kr,sh,war,sr-
rs,sr-cs,ar-AE,gl-ES,gd,nb-no,en-my,yo,fil-PH,en-BE,ar-iq,an,es-sv,am,et-
ee,fy,zh-hant,sah,ar-KW,it-CH,ar-sy,kis,es-ES_tradnl,nds,es-gt,yue,es-pa,es-
CR,scn,ast,ms-MY,pms,es-uy,ceb,ar-qa,es-do,lah,ar-JO,en-securid,pl-
pl,pa,eng,en-ph,ar-IQ,zh_TW,co,ar-om,en-PH,es-GT,hk,ar-bh,dv,lmo,ar-
ma,ia,iso,ar-dz,it-ch,es-DO,gsw,bo,q=0.5,zh-hans,es-PA,ar-QA,es-pr,ug,se,es-
bo,rup,ksh,ee,bpy,es-SV,es-
hn,ku,nan,vec,chrome://global/locale/intl.properties,[CUTTED]]
```

CVSSv3 Base Score:

3.3 (AV:L/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N)

MISSING TAPJACKING PROTECTION [M1] [CWE-451] [SAST]

LOW

Description:

The mobile application does not have a tapjacking protection required to mitigate tapjacking attacks.
By default, Android OS permits a mobile application to display its user interface over the user interface of another application installed and run on the device. When user touches the screen, application may pass the touch event to another application below its user interface layer that the user does not see, serving like a proxy to pass unintended touch activities. This attack is quite similar to clickjacking but for mobile devices. In order to be successfully exploited, a malicious application shall be already installed on the mobile phone of the victim. An example of exploitation would be a malware app that tricks user to unwittingly tap on a payment button (or any other functionality) of a sensitive application when playing a game or doing other innocent activity in the malicious application screen.

Example of secure code:

```
public class MyActivity extends Activity {
    protected void onCreate(Bundle bundle) {
        super.onCreate(bundle);

        final Button myButton = (Button)findViewById(R.id.button_id);
        myButton.setFilterTouchesWhenObscured(true);

        myButton.setOnClickListener(new View.OnClickListener() {
            // Perform action on click
        }
    }
}

<Button
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
    android:text="@string/self_destruct"
    android:onClick="selfDestruct"
    android:filterTouchesWhenObscured="true" />
```

Details:

There is **android:filterTouchesWhenObscured="true"** missing in files:

- android/res/layout-watch/abc_alert_dialog_button_bar_material.xml
- android/res/layout-watch/abc_alert_dialog_title_material.xml
- android/res/layout/abc_list_menu_item_checkbox.xml
- android/res/layout/abc_alert_dialog_button_bar_material.xml
- android/res/layout/abc_activity_chooser_view_list_item.xml
- android/res/layout/abc_screen_content_include.xml
- android/res/layout/abc_activity_chooser_view.xml
- android/res/layout/abc_select_dialog_material.xml
- android/res/layout/abc_action_bar_title_item.xml
- android/res/layout/abc_screen_toolbar.xml
- android/res/layout/ucrop_controls.xml
- android/res/layout/abc_dialog_title_material.xml
- android/res/layout/abc_search_view.xml
- android/res/layout/abc_search_dropdown_item_icons_2line.xml
- android/res/layout/abc_action_menu_item_layout.xml
- android/res/layout/abc_action_bar_up_container.xml
- android/res/layout/notification_template_icon_group.xml
- android/res/layout/abc_list_menu_item_layout.xml
- android/res/layout/notification_action_tombstone.xml
- android/res/layout/abc_action_menu_layout.xml
- android/res/layout/ucrop_aspect_ratio.xml
- android/res/layout/notification_template_part_time.xml
- android/res/layout/notification_action.xml
- android/res/layout/abc_list_menu_item_radio.xml
- android/res/layout/custom_dialog.xml
- android/res/layout/abc_screen_simple_overlay_action_mode.xml
- android/res/layout/abc_popup_menu_item_layout.xml
- android/res/layout/select_dialog_singlechoice_material.xml
- android/res/layout/browser_actions_context_menu_page.xml
- android/res/layout/select_dialog_item_material.xml
- android/res/layout/ucrop_layout_scale_wheel.xml
- android/res/layout/browser_actions_context_menu_row.xml
- android/res/layout/ucrop_fragment_photobox.xml
- android/res/layout/ucrop_layout_rotate_wheel.xml
- android/res/layout/support_simple_spinner_dropdown_item.xml
- android/res/layout/abc_alert_dialog_material.xml
- android/res/layout/abc_tooltip.xml
- android/res/layout/abc_action_mode_bar.xml
- android/res/layout/abc_action_mode_close_item_material.xml
- android/res/layout/abc_list_menu_item_icon.xml
- android/res/layout/zxing_capture.xml
- android/res/layout/abc_expanded_menu_layout.xml
- android/res/layout/abc_cascading_menu_item_layout.xml
- android/res/layout/ucrop_activity_photobox.xml
- android/res/layout/ucrop_view.xml
- android/res/layout/zxing_barcode_scanner.xml
- android/res/layout/notification_template_custom_big.xml
- android/res/layout/abc_popup_menu_header_item_layout.xml
- android/res/layout/abc_alert_dialog_title_material.xml
- android/res/layout/notification_template_part_chronometer.xml
- android/res/layout/abc_screen_simple.xml
- android/res/layout/select_dialog_multichoice_material.xml
- android/res/layout-v26/abc_screen_toolbar.xml

There is **'extends ImageView'** found in file 'io/flutter/embedding/android/c.java':

```
[line 41: ]
[line 42:     public static class b extends ImageView {]
[line 43:         public b(Context context) {]
```

There is **'extends View'** found in file 'io/flutter/embedding/android/h.java':

```
[line 22: @TargetApi(19)]
[line 23: public class h extends View implements c {]
[line 24:     private ImageReader f;]
```

There is **'extends View'** found in file 'com/yalantis/ucrop/view/OverlayView.java':

```
[line 21: ]
[line 22: public class OverlayView extends View {]
[line 23:     private float A;]
```

There is **'extends View'** found in file 'com/yalantis/ucrop/view/widget/HorizontalProgressWheelView.java':

```
[line 14: ]
[line 15: public class HorizontalProgressWheelView extends View {]
[line 16:     private final Rect f;]
```

There is **'extends View'** found in file 'com/journeyapps/barcodescanner/ViewfinderView.java':

```
[line 18: ]
[line 19: public class ViewfinderView extends View {]
[line 20:     protected static final String f = ViewfinderView.class.getSimpleName();]
```

There is **'extends FrameLayout'** found in file 'io/flutter/embedding/android/FlutterSplashView.java':

```
[line 14: ]
[line 15: final class FlutterSplashView extends FrameLayout {]
[line 16:     private static String f = "FlutterSplashView";]
```

There is **'extends FrameLayout'** found in file 'io/flutter/embedding/android/k.java':

```
[line 43: ]
[line 44: public class k extends FrameLayout implements f.a.c.c.a.c {]
[line 45:     private i f;]
```

There is **'extends FrameLayout'** found in file 'io/flutter/embedding/engine/mutatorsstack/a.java':

```
[line 17: ]
[line 18: public class a extends FrameLayout {]
[line 19:     private FlutterMutatorsStack f;]
```

There is **'extends FrameLayout'** found in file 'com/yalantis/ucrop/view/UCropView.java':

```
[line 14: ]
[line 15: public class UCropView extends FrameLayout {]
[line 16:     private GestureCropImageView f;]
```

There is **'extends FrameLayout'** found in file 'com/journeyapps/barcodescanner/DecoratedBarcodeView.java':

```
[line 20: ]
[line 21: public class DecoratedBarcodeView extends FrameLayout {]
[line 22:     private BarcodeView f;]
```

There is **'extends SurfaceView'** found in file 'io/flutter/embedding/android/i.java':

```
[line 11: ]
[line 12: public class i extends SurfaceView implements c {]
[line 13:     private final boolean f;]
```

There is **'extends TextureView'** found in file 'io/flutter/embedding/android/j.java':

```
[line 11: ]
[line 12: public class j extends TextureView implements c {]
[line 13:     private boolean f;]
```

There is **'extends ViewGroup'** found in file 'io/flutter/plugin/platform/SingleViewPresentation.java':

```
[line 57: ]
[line 58:     static class b extends ViewGroup {]
[line 59:         private final Rect f = new Rect();]
```

There is **'extends ViewGroup'** found in file 'c/o/z.java':

```
[line 17: ]
[line 18:     static class a extends ViewGroup {]
[line 19:         static Method f;]
```

There is **'extends ViewGroup'** found in file 'com/journeyapps/barcodescanner/i.java':

```
[line 32: ]
[line 33: public class i extends ViewGroup {]
[line 34:     private static final String f = i.class.getSimpleName();]
```

CVSSv3 Base Score:

3.3 (AV:L/AC:H/PR:L/UI:R/S:U/C:L/I:L/A:N)

Reference:

- https://developer.android.com/guide/topics/ui/declaring-layout.html
- https://developer.android.com/guide/topics/resources/layout-resource.html
- https://blog.lookout.com/blog/2010/12/09/android-touch-event-hijacking/

EXPORTED ACTIVITIES [M1] [CWE-926] [SAST]

LOW

Description:

The mobile application contains exported activities that can be loaded and executed by other applications residing on the mobile device, including malicious ones, to trigger a legitimate application activity.
An activity is an Android component that allows to interact with the application in a particular way (e.g. perform certain actions or functions).

Example of insecure code:

```
<activity
    android:name=".SomeActivity"
    ....
    android:exported="true" />
```

Example of secure code:

```
<activity
    android:name=".SomeActivity"
    ....
    android:exported="false" />
```

Details:

There is **'MainActivity'** found in file 'android/AndroidManifest.xml':

```
[line 30:         <activity
android:configChanges="density|fontScale|keyboard|keyboardHidden|layoutDirection|locale
|orientation|screenLayout|screenSize|smallestScreenSize|uiMode" android:exported="true"
android:hardwareAccelerated="true" android:launchMode="singleTop"
android:name="com.italinnovation.green_pass.MainActivity"
android:theme="@style/LaunchTheme" android:windowSoftInputMode="adjustResize">]
```

CVSSv3 Base Score:

3.6 (AV:L/AC:H/PR:L/UI:N/S:U/C:L/I:L/A:N)

Reference:

- https://developer.android.com/reference/android/app/Activity.html
- https://developer.android.com/guide/topics/manifest/activity-element.html

TEMPORARY FILE CREATION [SAST]

WARNING

Description:

The mobile application creates temporary files. Despite that cache files are usually private by default, it is recommended to make sure that temporary files are securely deleted when they are not required by the application anymore.

Details:

There is **'createTempFile()'** found in file 'io/flutter/plugins/imagepicker/c.java':

```
[line 98:         r5 = r5.getCacheDir();         Catch:{ IOException -> 0x003a, all ->
0x0036 }]
[line 99:         r5 = java.io.File.createTempFile(r3, r2, r5);  Catch:{ IOException ->
0x003a, all -> 0x0036 }]
[line 100:         r5.deleteOnExit();    Catch:{ IOException -> 0x0034, all -> 0x0036
}]
```

There is **'createTempFile()'** found in file 'io/flutter/plugins/imagepicker/e.java':

```
[line 310:             this.h.mkdirs();]
[line 311:             return File.createTempFile(uuid, str, this.h);]
[line 312:         } catch (IOException e) {]
```

There is **'createTempFile()'** found in file 'c/h/a/a.java':

```
[line 4529:         r3 = "tmp";]
[line 4530:         r2 = java.io.File.createTempFile(r2, r3);    Catch:{ Exception ->
0x0153, all -> 0x0150 }]
[line 4531:         r3 = r14.d0;         Catch:{ Exception -> 0x0153, all -> 0x0150 }]
```

There is **'createTempFile()'** found in file 'com/journeyapps/barcodescanner/j.java':

```
[line 158:             try {]
[line 159:                 File createTempFile = File.createTempFile("barcodeimage",
".jpg", this.c.getCacheDir());]
[line 160:                 FileOutputStream fileOutputStream = new
FileOutputStream(createTempFile);]
```

There is **'createTempFile()'** found in file 'l/a/a/a/a/a.java':

```
[line 173:         r4 = r4.getCacheDir();        Catch:{ IOException -> 0x0035, all ->
0x0031 }]
[line 174:         r4 = java.io.File.createTempFile(r2, r3, r4);         Catch:{
IOException -> 0x0035, all -> 0x0031 }]
[line 175:         r2 = new java.io.FileOutputStream;    Catch:{ IOException -> 0x002f,
all -> 0x0031 }]
```

Reference:

- https://developer.android.com/training/basics/data-storage/files.html

USAGE OF IMPLICIT INTENT [M1] [CWE-927] [SAST]

WARNING

Description:

The mobile application uses implicit intent that may be insecure under certain conditions.
Intents enable mobile applications to communicate with each other by requesting to perform different actions for which they are better suited. An implicit intent, however, does not specify to which particular application it sends a request to perform an action. If a malicious application is installed on victim's device, it may also receive the implicit intent, respond to it and perform some action instead, or in addition to, a legitimate application.

Example of insecure code:

```
Intent sendIntent = new Intent();
```

Example of secure code:

```
Intent downloadIntent = new Intent(this, DownloadService.class);
```

Details:

There is **'new Intent()'** found in file 'io/flutter/plugins/firebase/messaging/t.java':

```
[line 475:             if (intent == null) {]
[line 476:                 intent = new Intent();]
[line 477:             }]
```

There is **'new Intent()'** found in file 'dev/fluttercommunity/plus/share/b.java':

```
[line 168:         }]
[line 169:         Intent intent = new Intent();]
[line 170:         intent.setAction("android.intent.action.SEND");]
```

```
[line 183:         ArrayList h = h(list);]
[line 184:         Intent intent = new Intent();]
[line 185:         if (h.isEmpty()) {]
```

There is **'new Intent()'** found in file 'd/a/a/n.java':

```
[line 127:         String packageName = this.g.getPackageName();]
[line 128:         Intent intent = new Intent();]
[line 129:         intent.setAction(str);]
```

There is **'new Intent()'** found in file 'd/a/a/j.java':

```
[line 25:         try {]
[line 26:             Intent intent = new Intent();]
[line 27:
intent.setAction("android.settings.APPLICATION_DETAILS_SETTINGS");]
```

There is **'new Intent()'** found in file 'd/b/e/a/c/n.java':

```
[line 28:     public static void c(@RecentlyNonNull Context context, @RecentlyNonNull
List<String> list) {]
[line 29:         Intent intent = new Intent();]
[line 30:         intent.setClassName("com.google.android.gms",
"com.google.android.gms.vision.DependencyBroadcastReceiverProxy");]
```

There is **'new Intent()'** found in file 'd/b/b/b/c/d.java':

```
[line 145:         }]
[line 146:         Intent intent = new Intent();]
[line 147:         intent.setPackage("com.google.android.gms");]
```

```
[line 203:             if (b == null) {]
[line 204:                 Intent intent2 = new Intent();]
[line 205:                 intent2.setPackage("com.google.example.invalidpackage");]
```

There is **'new Intent()'** found in file 'com/yalantis/ucrop/j.java':

```
[line 12: public class j {]
[line 13:     private Intent a = new Intent();]
[line 14:     private Bundle b;]
```

There is **'new Intent()'** found in file 'com/yalantis/ucrop/UCropActivity.java':

```
[line 546:     public void h0(Throwable th) {]
[line 547:         setResult(96, new Intent().putExtra("com.yalantis.ucrop.Error",
th));]
[line 548:     }]
```

```
[line 551:     public void i0(Uri uri, float f, int i, int i2, int i3, int i4) {]
[line 552:         setResult(-1, new Intent().putExtra("com.yalantis.ucrop.OutputUri",
uri).putExtra("com.yalantis.ucrop.CropAspectRatio",
f).putExtra("com.yalantis.ucrop.ImageWidth",
i3).putExtra("com.yalantis.ucrop.ImageHeight",
i4).putExtra("com.yalantis.ucrop.OffsetX", i).putExtra("com.yalantis.ucrop.OffsetY",
i2));]
[line 553:     }]
```

Reference:

- https://developer.android.com/guide/components/intents-filters.html
- https://developer.android.com/training/articles/security-tips.html

USAGE OF INTENT FILTER [M1] [CWE-927] [SAST]

WARNING

Description:

The mobile application uses an intent filter that may be a serious security risk if not properly implemented and filtered. Developers should not solely rely on intent filters for security purposes because they place no restrictions on explicit intents. Intent filters are defined in the Android Manifest file, they let developers choose which type of intents their application components are supposed to receive and handle.

Example of insecure code:

```
<intent-filter>
    <action android:name="android.intent.action.VIEW" />
    <action android:name="android.intent.action.EDIT" />
    <action android:name="android.intent.action.PICK" />
    <category android:name="android.intent.category.DEFAULT" />
    <data mimeType:name="vnd.android.cursor.dir/vnd.google.note" />
</intent-filter>
```

Example of secure code:

```
// When you use intent-filter, you have to perform input validation in your code.
```

Details:

There is **'<intent-filter>'** found in file 'android/AndroidManifest.xml':

```
[line 32:             <meta-data
android:name="io.flutter.embedding.android.SplashScreenDrawable"
android:resource="@drawable/launch_background"/>]
[line 33:             <intent-filter>]
[line 34:                 <action android:name="android.intent.action.MAIN"/>]
```

```
[line 40:         <service android:exported="false"
android:name="io.flutter.plugins.firebase.messaging.FlutterFirebaseMessagingService">]
[line 41:             <intent-filter>]
[line 42:                 <action android:name="com.google.firebase.MESSAGING_EVENT"/>]
```

```
[line 45:         <receiver android:exported="true"
android:name="io.flutter.plugins.firebase.messaging.FlutterFirebaseMessagingReceiver"
android:permission="com.google.android.c2dm.permission.SEND">]
[line 46:             <intent-filter>]
[line 47:                 <action
android:name="com.google.android.c2dm.intent.RECEIVE"/>]
```

```
[line 91:         <receiver android:exported="true"
android:name="com.google.firebase.iid.FirebaseInstanceIdReceiver"
android:permission="com.google.android.c2dm.permission.SEND">]
[line 92:             <intent-filter>]
[line 93:                 <action
android:name="com.google.android.c2dm.intent.RECEIVE"/>]
```

```
[line 115:         <receiver android:directBootAware="false" android:enabled="false"
android:exported="false"
android:name="androidx.work.impl.background.systemalarm.ConstraintProxy$BatteryCharging
Proxy">]
[line 116:             <intent-filter>]
[line 117:                 <action
android:name="android.intent.action.ACTION_POWER_CONNECTED"/>]
```

```
[line 121:         <receiver android:directBootAware="false" android:enabled="false"
android:exported="false"
android:name="androidx.work.impl.background.systemalarm.ConstraintProxy$BatteryNotLowPr
oxy">]
[line 122:             <intent-filter>]
[line 123:                 <action android:name="android.intent.action.BATTERY_OKAY"/>]
```

```
[line 127:         <receiver android:directBootAware="false" android:enabled="false"
android:exported="false"
android:name="androidx.work.impl.background.systemalarm.ConstraintProxy$StorageNotLowPr
oxy">]
[line 128:             <intent-filter>]
[line 129:                 <action
android:name="android.intent.action.DEVICE_STORAGE_LOW"/>]
```

```
[line 133:         <receiver android:directBootAware="false" android:enabled="false"
android:exported="false"
android:name="androidx.work.impl.background.systemalarm.ConstraintProxy$NetworkStatePro
xy">]
[line 134:             <intent-filter>]
[line 135:                 <action
android:name="android.net.conn.CONNECTIVITY_CHANGE"/>]
```

```
[line 138:         <receiver android:directBootAware="false" android:enabled="false"
android:exported="false"
android:name="androidx.work.impl.background.systemalarm.RescheduleReceiver">]
[line 139:             <intent-filter>]
[line 140:                 <action
android:name="android.intent.action.BOOT_COMPLETED"/>]
```

```
[line 145:         <receiver android:directBootAware="false"
android:enabled="@bool/enable_system_alarm_service_default" android:exported="false"
android:name="androidx.work.impl.background.systemalarm.ConstraintProxyUpdateReceiver">
]
[line 146:             <intent-filter>]
[line 147:                 <action
android:name="androidx.work.impl.background.systemalarm.UpdateProxies"/>]
```

```
[line 150:         <receiver android:directBootAware="false" android:enabled="true"
android:exported="true"
android:name="androidx.work.impl.diagnostics.DiagnosticsReceiver"
android:permission="android.permission.DUMP">]
[line 151:             <intent-filter>]
[line 152:                 <action
android:name="androidx.work.diagnostics.REQUEST_DIAGNOSTICS"/>]
```

Reference:

- https://developer.android.com/guide/components/intents-filters.html
- https://developer.android.com/training/articles/security-tips.html

DYNAMIC LOAD OF CODE [M7] [CWE-94] [SAST]

WARNING

Description:

The mobile application uses dynamic load of executable code. Under certain circumstances, dynamic load of code can be dangerous. For example, if the code is located on an external storage (e.g. SD card), this can lead to code injection vulnerability if the external storage is world readable and/or writable and an attacker can access it.

Example of insecure code:

```
Object test = loader.loadClass("Test", true).newInstance();
```

Example of secure code:

```
// If you are using code from unsafe place (like external storage),
// you should sign and cryptographically verify your code.
```

Details:

There is **'ClassLoader'** found in file 'io/flutter/plugins/firebase/messaging/t.java':

```
[line 257:         /* JADX WARNING: Missing block: B:12:0x0011, code skipped:]
[line 258:             r1.getIntent().setExtrasClassLoader(r5.a.getClassLoader());]
[line 259:      */]
```

```
[line 283:             r2 = r5.a;]
[line 284:             r2 = r2.getClassLoader();]
[line 285:             r0.setExtrasClassLoader(r2);]
[line 286:             r0 = new io.flutter.plugins.firebase.messaging.t$g$a;]
```

There is **'ClassLoader'** found in file 'io/flutter/embedding/android/FlutterSplashView.java':

```
[line 50:             this.previousCompletedSplashIsolate = parcel.readString();]
[line 51:             this.splashScreenState =
parcel.readBundle(getClass().getClassLoader());]
[line 52:         }]
```

There is **'ClassLoader'** found in file 'io/flutter/plugin/platform/SingleViewPresentation.java':

```
[line 205:         public WindowManager b() {]
[line 206:             return (WindowManager)
Proxy.newProxyInstance(WindowManager.class.getClassLoader(), new Class[]
{WindowManager.class}, this);]
[line 207:         }]
```

There is **'ClassLoader'** found in file 'h/v/j/a/i.java':

```
[line 26:         try {]
[line 27:             a aVar2 = new a(Class.class.getDeclaredMethod("getModule", new
Class[0]),
aVar.getClass().getClassLoader().loadClass("java.lang.Module").getDeclaredMethod("getDe
scriptor", new Class[0]),
aVar.getClass().getClassLoader().loadClass("java.lang.module.ModuleDescriptor").getDecl
aredMethod("name", new Class[0]));]
[line 28:             c = aVar2;]
```

There is **'ClassLoader'** found in file 'h/w/b.java':

```
[line 14:         r2 = "ClassCastException(\"Inst…baseTypeCL\").initCause(e)";]
[line 15:         r3 = ", base type ClassLoader: ";]
[line 16:         r4 = "Instance ClassLoader: ";]
[line 17:         r5 = "forName(\"kotlin.internal…entations\").newInstance()";]
```

```
[line 29:         r6 = r6.getClass();    Catch:{ ClassNotFoundException -> 0x0053 }]
[line 30:         r6 = r6.getClassLoader();      Catch:{ ClassNotFoundException ->
0x0053 }]
[line 31:         r8 = r0.getClassLoader();      Catch:{ ClassNotFoundException ->
0x0053 }]
[line 32:         r9 = new java.lang.ClassCastException;         Catch:{
ClassNotFoundException -> 0x0053 }]
```

```
[line 53:         r6 = r6.getClass();    Catch:{ ClassNotFoundException -> 0x0093 }]
[line 54:         r6 = r6.getClassLoader();      Catch:{ ClassNotFoundException ->
0x0093 }]
[line 55:         r8 = r0.getClassLoader();      Catch:{ ClassNotFoundException ->
0x0093 }]
[line 56:         r9 = new java.lang.ClassCastException;         Catch:{
ClassNotFoundException -> 0x0093 }]
```

```
[line 81:         r1 = r1.getClass();    Catch:{ ClassNotFoundException -> 0x00d8 }]
[line 82:         r1 = r1.getClassLoader();      Catch:{ ClassNotFoundException ->
0x00d8 }]
[line 83:         r7 = r0.getClassLoader();      Catch:{ ClassNotFoundException ->
0x00d8 }]
[line 84:         r8 = new java.lang.ClassCastException;         Catch:{
ClassNotFoundException -> 0x00d8 }]
```

```
[line 106:         r1 = r1.getClass();   Catch:{ ClassNotFoundException -> 0x0118 }]
[line 107:         r1 = r1.getClassLoader();     Catch:{ ClassNotFoundException ->
0x0118 }]
[line 108:         r0 = r0.getClassLoader();     Catch:{ ClassNotFoundException ->
0x0118 }]
[line 109:         r6 = new java.lang.ClassCastException;        Catch:{
ClassNotFoundException -> 0x0118 }]
```

There is **'ClassLoader'** found in file 'j/e0/j/b.java':

```
[line 441:             try {]
[line 442:                 v = g.v(sSLSocketFactory,
Class.forName("com.google.android.gms.org.conscrypt.SSLParametersImpl", false,
sSLSocketFactory.getClass().getClassLoader()), str);]
[line 443:             } catch (ClassNotFoundException unused) {]
```

There is **'ClassLoader'** found in file 'j/e0/j/e.java':

```
[line 107:         try {]
[line 108:             Object newProxyInstance =
Proxy.newProxyInstance(g.class.getClassLoader(), new Class[]{this.f, this.g}, new
a(b));]
[line 109:             this.c.invoke(null, new Object[]{sSLSocket, newProxyInstance});]
```

There is **'ClassLoader'** found in file 'c/g/a/a.java':

```
[line 4: import android.os.Parcelable;]
[line 5: import android.os.Parcelable.ClassLoaderCreator;]
[line 6: import android.os.Parcelable.Creator;]
```

```
[line 12: ]
[line 13:     static class b implements ClassLoaderCreator<a> {]
[line 14:         b() {]
```

```
[line 22:         /* renamed from: b */]
[line 23:         public a createFromParcel(Parcel parcel, ClassLoader ClassLoader) {]
[line 24:             if (parcel.readParcelable(ClassLoader) == null) {]
[line 25:                 return a.f;]
```

```
[line 45: ]
[line 46:     protected a(Parcel parcel, ClassLoader ClassLoader) {]
[line 47:         Parcelable readParcelable = parcel.readParcelable(ClassLoader);]
[line 48:         if (readParcelable == null) {]
```

There is **'ClassLoader'** found in file 'c/a/n/g.java':

```
[line 115:             try {]
[line 116:                 Constructor constructor = Class.forName(str, false,
g.this.e.getClassLoader()).getConstructor(clsArr);]
[line 117:                 constructor.setAccessible(true);]
```

There is **'ClassLoader'** found in file 'd/b/b/c/a/b/l0.java':

```
[line 9:     static {]
[line 10:         l0.class.getClassLoader();]
[line 11:     }]
```

There is **'ClassLoader'** found in file 'd/b/b/b/c/h.java':

```
[line 5: ]
[line 6: public final class h extends ClassLoader {]
[line 7:     /* Access modifiers changed, original: protected|final */]
```

There is **'ClassLoader'** found in file 'd/b/b/b/c/d.java':

```
[line 58:                 Intent intent = (Intent) obj;]
[line 59:                 intent.setExtrasClassLoader(new h());]
[line 60:                 if (intent.hasExtra("google.messenger")) {]
```

There is **'ClassLoader'** found in file 'd/b/b/b/d/c.java':

```
[line 37:                 try {]
[line 38:                     this.b = a((IBinder)
context.getClassLoader().loadClass(this.a).newInstance());]
[line 39:                 } catch (ClassNotFoundException e) {]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/r/kz.java':

```
[line 21:         Class cls2 = kz.class;]
[line 22:         ClassLoader classLoader = cls2.getClassLoader();]
[line 23:         if (cls.equals(cz.class)) {]
```

```
[line 30:         try {]
[line 31:             return (cz) cls.cast(((kz) Class.forName(str, true,
ClassLoader).getConstructor(new Class[0]).newInstance(new Object[0])).a());]
[line 32:         } catch (NoSuchMethodException e) {]
```

```
[line 40:         } catch (ClassNotFoundException unused) {]
[line 41:             Iterator it = ServiceLoader.load(cls2, ClassLoader).iterator();]
[line 42:             ArrayList arrayList = new ArrayList();]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/r/tz.java':

```
[line 62:             try {]
[line 63:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 64:                 t = (tz) map.get(cls);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/u/w7.java':

```
[line 167:             try {]
[line 168:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 169:                 t = (w7) zzd.get(cls);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/i/yt.java':

```
[line 21:         Class cls2 = yt.class;]
[line 22:         ClassLoader classLoader = cls2.getClassLoader();]
[line 23:         if (cls.equals(qt.class)) {]
```

```
[line 30:         try {]
[line 31:             return (qt) cls.cast(((yt) Class.forName(str, true,
ClassLoader).getConstructor(new Class[0]).newInstance(new Object[0])).a());]
[line 32:         } catch (NoSuchMethodException e) {]
```

```
[line 40:         } catch (ClassNotFoundException unused) {]
[line 41:             Iterator it = ServiceLoader.load(cls2, ClassLoader).iterator();]
[line 42:             ArrayList arrayList = new ArrayList();]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/i/gu.java':

```
[line 66:             try {]
[line 67:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 68:                 t = (gu) map.get(cls);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/i/z6.java':

```
[line 22:             }]
[line 23:             InputStream resourceAsStream = cls.getClassLoader() != null ?
cls.getClassLoader().getResourceAsStream(str) :
ClassLoader.getSystemResourceAsStream(str);]
[line 24:             String d;]
```

```
[line 27:                 StringBuilder stringBuilder = new
StringBuilder(String.valueOf(str).length() + 37);]
[line 28:                 stringBuilder.append("Couldn't find ClassLoader resource
'");]
[line 29:                 stringBuilder.append(str);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/n/e0.java':

```
[line 21:             try {]
[line 22:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 23:                 t = (e0) map.get(cls);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/g/h0.java':

```
[line 8: public final class h0 {]
[line 9:     private static final ClassLoader a = h0.class.getClassLoader();]
[line 10:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/t/b1.java':

```
[line 7: public final class b1 {]
[line 8:     private static final ClassLoader a = b1.class.getClassLoader();]
[line 9:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/a/c.java':

```
[line 5: public class c {]
[line 6:     private static final ClassLoader a = c.class.getClassLoader();]
[line 7:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/o/c1.java':

```
[line 7: public final class c1 {]
[line 8:     private static final ClassLoader a = c1.class.getClassLoader();]
[line 9:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/a0/b1.java':

```
[line 8: public final class b1 {]
[line 9:     private static final ClassLoader a = b1.class.getClassLoader();]
[line 10:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/c/f1.java':

```
[line 244:             try {]
[line 245:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 246:                 t = (f1) zzjr.get(cls);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/c/y0.java':

```
[line 8: public class y0 {]
[line 9:     private static final ClassLoader a = y0.class.getClassLoader();]
[line 10:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/e/c.java':

```
[line 8: public final class c {]
[line 9:     private static final ClassLoader a = c.class.getClassLoader();]
[line 10:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/p/c1.java':

```
[line 8: public final class c1 {]
[line 9:     private static final ClassLoader a = c1.class.getClassLoader();]
[line 10:]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/p/i3.java':

```
[line 22:             try {]
[line 23:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 24:                 t = (i3) map.get(cls);]
```

There is **'ClassLoader'** found in file 'd/b/b/b/e/w/wn.java':

```
[line 26:             try {]
[line 27:                 Class.forName(cls.getName(), true, cls.getClassLoader());]
[line 28:                 t = (wn) map.get(cls);]
```

Reference:

- https://developer.android.com/reference/java/lang/ClassLoader.html
- https://developer.android.com/reference/dalvik/system/DexClassLoader.html
- https://developer.android.com/reference/java/security/SecureClassLoader.html
- https://developer.android.com/reference/java/net/URLClassLoader.html

OBJECT DESERIALIZATION FOUND [M7] [CWE-502] [SAST]

WARNING

Description:

Object deserialization performed on an untrusted resource (e.g. user-supplied input or external storage), can be dangerous if the data for deserialization is tampered by an attacker.

Example of insecure code:

```
bundle.putSerializable("exampleClass", exampleOfSerializabledClass);
exampleOfSerializabledClass = bundle.getSerializable("exampleClass");
```

Example of secure code:

```
// Use only serialization when you have the control over data
```

Details:

There is **'implements Serializable'** found in file 'h/l.java':

```
[line 5: ]
[line 6: public final class l<A, B> implements Serializable {]
[line 7:     private final A f;]
```

There is **'implements Serializable'** found in file 'h/m.java':

```
[line 6: ]
[line 7: public final class m<T> implements Serializable {]
[line 8:     public static final a f = new a();]
```

```
[line 18: ]
[line 19:     public static final class b implements Serializable {]
[line 20:         public final Throwable f;]
```

There is **'implements Serializable'** found in file 'h/z/c.java':

```
[line 10: ]
[line 11:     public static final class a extends c implements Serializable {]
[line 12:         private a() {]
```

There is **'implements Serializable'** found in file 'h/y/d/a.java':

```
[line 20: ]
[line 21:     private static class a implements Serializable {]
[line 22:         private static final a f = new a();]
```

There is **'implements Serializable'** found in file 'd/b/g/y/c/e.java':

```
[line 19: ]
[line 20:     private static final class b implements Serializable, Comparator<d> {]
[line 21:         private b() {]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/y60.java':

```
[line 5: ]
[line 6: public class y60 implements Serializable, Comparable<y60> {]
[line 7:     static final char[] f = new char[]{'0', '1', '2', '3', '4', '5', '6', '7',
'8', '9', 'a', 'b', 'c', 'd', 'e', 'f'};]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/o3.java':

```
[line 9: ]
[line 10: abstract class o3<K, V> extends q3<K, V> implements Serializable {]
[line 11:     private final transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/u4.java':

```
[line 4: ]
[line 5: final class u4<K, V> extends p3<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/x1.java':

```
[line 4: ]
[line 5: public final class x1<A, B> implements Serializable {]
[line 6:     public final A f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/k4.java':

```
[line 8: ]
[line 9: final class k4<E> extends AbstractSet<E> implements Serializable {]
[line 10:     private transient Object f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/s2.java':

```
[line 5: ]
[line 6: final class s2<T> implements Serializable, r2 {]
[line 7:     final r2<T> f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/i4.java':

```
[line 10: ]
[line 11: final class i4<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/t4.java':

```
[line 8: ]
[line 9: public abstract class t4<E> extends AbstractCollection<E> implements
Serializable {]
[line 10:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/u2.java':

```
[line 5: ]
[line 6: final class u2<T> implements Serializable, r2 {]
[line 7:     final T f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/it.java':

```
[line 10: ]
[line 11: public final class it implements Serializable {]
[line 12:     public static final it A = a("sq");]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/e5.java':

```
[line 7: ]
[line 8: public class e5<K, V> extends u3<K, V> implements Serializable {]
[line 9:     final transient c5<K, ? extends t4<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/yp.java':

```
[line 9: ]
[line 10: public final class yp<K, V> extends AbstractMap<K, V> implements Serializable
{]
[line 11:     private static final Comparator<Comparable> f = new rp();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/r/w1.java':

```
[line 5: ]
[line 6: public abstract class w1<T> implements Serializable {]
[line 7:     w1() {]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/u/p8.java':

```
[line 10: ]
[line 11: abstract class p8<K, V> extends vc<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/u/j.java':

```
[line 4: ]
[line 5: final class j<K, V> extends sc<K, V> implements Serializable {]
[line 6:     private final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/u/r6.java':

```
[line 8: ]
[line 9: public abstract class r6 implements Serializable, Iterable<Byte> {]
[line 10:     public static final r6 f = new b7(z7.c);]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/u/k.java':

```
[line 8: ]
[line 9: public abstract class k<E> extends AbstractCollection<E> implements
Serializable {]
[line 10:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/u/yc.java':

```
[line 11: ]
[line 12: final class yc<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 13:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/u4.java':

```
[line 9: ]
[line 10: public abstract class u4<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/u3.java':

```
[line 9: ]
[line 10: abstract class u3<K, V> extends w3<K, V> implements Serializable {]
[line 11:     private final transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/v4.java':

```
[line 4: ]
[line 5: final class v4<K, V> extends v3<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/m4.java':

```
[line 10: ]
[line 11: final class m4<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/y2.java':

```
[line 5: ]
[line 6: final class y2<T> implements Serializable, x2 {]
[line 7:     final x2<T> f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/v20.java':

```
[line 5: ]
[line 6: public class v20 implements Serializable, Comparable<v20> {]
[line 7:     static final char[] f = new char[]{'0', '1', '2', '3', '4', '5', '6', '7',
'8', '9', 'a', 'b', 'c', 'd', 'e', 'f'};]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/d2.java':

```
[line 5: ]
[line 6: public abstract class d2<T> implements Serializable {]
[line 7:     d2() {]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/o6.java':

```
[line 4: ]
[line 5: public final class o6 implements Serializable {]
[line 6:     public static final o6 f = new o6(0.0d, 0.0d);]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/i/a3.java':

```
[line 5: ]
[line 6: final class a3<T> implements Serializable, x2 {]
[line 7:     final T f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/n/da.java':

```
[line 4: ]
[line 5: public abstract class da<T> implements Serializable {]
[line 6:     da() {]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/n/ui.java':

```
[line 9: ]
[line 10: public final class ui<K, V> extends AbstractMap<K, V> implements Serializable
{]
[line 11:     private static final Comparator<Comparable> f = new ni();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/n/pm.java':

```
[line 9: ]
[line 10: public abstract class pm<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/z/d.java':

```
[line 10: ]
[line 11: abstract class d<K, V> extends f<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/z/n2.java':

```
[line 9: ]
[line 10: public final class n2<K, V> extends AbstractMap<K, V> implements Serializable
{]
[line 11:     private static final Comparator<Comparable> f = new g2();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/z/s.java':

```
[line 10: ]
[line 11: final class s<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/z/y.java':

```
[line 4: ]
[line 5: final class y<K, V> extends e<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/z/x.java':

```
[line 9: ]
[line 10: public abstract class x<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/t/a0.java':

```
[line 4: ]
[line 5: final class a0<K, V> extends i<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/t/z.java':

```
[line 9: ]
[line 10: public abstract class z<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/t/h.java':

```
[line 10: ]
[line 11: abstract class h<K, V> extends j<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/t/u.java':

```
[line 10: ]
[line 11: final class u<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/s/e0.java':

```
[line 10: ]
[line 11: final class e0<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/s/q.java':

```
[line 10: ]
[line 11: abstract class q<K, V> extends s<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/s/j0.java':

```
[line 9: ]
[line 10: public abstract class j0<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/s/k0.java':

```
[line 4: ]
[line 5: final class k0<K, V> extends r<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/o/z0.java':

```
[line 9: ]
[line 10: public abstract class z0<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/o/g0.java':

```
[line 10: ]
[line 11: abstract class g0<K, V> extends i0<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/o/a1.java':

```
[line 4: ]
[line 5: final class a1<K, V> extends h0<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/o/u0.java':

```
[line 10: ]
[line 11: final class u0<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/h/k.java':

```
[line 9: ]
[line 10: public abstract class k<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/a0/p.java':

```
[line 10: ]
[line 11: abstract class p<K, V> extends r<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/a0/u0.java':

```
[line 8: ]
[line 9: final class u0<F, T> extends AbstractSequentialList<T> implements Serializable
{]
[line 10:     final List<F> f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/a0/k0.java':

```
[line 4: ]
[line 5: final class k0<K, V> extends q<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/m/xc.java':

```
[line 9: ]
[line 10: public abstract class xc<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/m/w0.java':

```
[line 9: ]
[line 10: public final class w0<K, V> extends AbstractMap<K, V> implements Serializable
{]
[line 11:     private static final Comparator<Comparable> f = new p0();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/j/e9.java':

```
[line 9: ]
[line 10: public abstract class e9<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/c/a0.java':

```
[line 6: ]
[line 7: public abstract class a0 implements Serializable, Iterable<Byte> {]
[line 8:     public static final a0 f = new h0(h1.c);]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/q/v7.java':

```
[line 9: ]
[line 10: public abstract class v7<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/x/q.java':

```
[line 9: ]
[line 10: public abstract class q<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/x/ye.java':

```
[line 10: ]
[line 11: abstract class ye<K, V> extends af<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/x/r.java':

```
[line 4: ]
[line 5: final class r<K, V> extends ze<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/x/l.java':

```
[line 10: ]
[line 11: final class l<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/x/g2.java':

```
[line 9: ]
[line 10: public final class g2<K, V> extends AbstractMap<K, V> implements Serializable
{]
[line 11:     private static final Comparator<Comparable> f = new y1();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/l/j8.java':

```
[line 9: ]
[line 10: public abstract class j8<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/y/s.java':

```
[line 4: ]
[line 5: final class s<K, V> extends db<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/y/cb.java':

```
[line 10: ]
[line 11: abstract class cb<K, V> extends b<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/y/r.java':

```
[line 9: ]
[line 10: public abstract class r<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/y/m.java':

```
[line 10: ]
[line 11: final class m<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/w/ho.java':

```
[line 10: ]
[line 11: abstract class ho<K, V> extends jq<K, V> implements Serializable {]
[line 12:     private transient Map<K, Collection<V>> h;]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/w/e1.java':

```
[line 9: ]
[line 10: public abstract class e1<E> extends AbstractCollection<E> implements
Serializable {]
[line 11:     private static final Object[] f = new Object[0];]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/w/z0.java':

```
[line 10: ]
[line 11: final class z0<K, V> extends AbstractMap<K, V> implements Serializable {]
[line 12:     private static final Object f = new Object();]
```

There is **'implements Serializable'** found in file 'd/b/b/b/e/w/f1.java':

```
[line 4: ]
[line 5: final class f1<K, V> extends ip<K, V> implements Serializable {]
[line 6:     final K f;]
```

There is **'implements Serializable'** found in file 'k/f.java':

```
[line 7: ]
[line 8: public class f implements Serializable, Comparable<f> {]
[line 9:     static final char[] f = new char[]{'0', '1', '2', '3', '4', '5', '6', '7',
'8', '9', 'a', 'b', 'c', 'd', 'e', 'f'};]
```

There is **'getSerializableExtra()'** found in file 'com/yalantis/ucrop/j.java':

```
[line 116:     public static Throwable a(Intent intent) {]
[line 117:         return (Throwable)
intent.getSerializableExtra("com.yalantis.ucrop.Error");]
[line 118:     }]
```

Reference:

- https://developer.android.com/reference/android/os/Bundle.html

MISSING ANTI-EMULATION [SAST]

WARNING

Description:

The mobile application does not use any anti-emulation or anti-debugger techniques (e.g. detecting rooted devices or checking if contacts are authentic).
This can significantly facilitate application debugging and reverse-engineering processes.

Reference:

- https://github.com/strazzere/anti-emulator

NETWORK SECURITY CONFIGURATION IS NOT PRESENT [SAST]

WARNING

Description:

The mobile application does not use Network Security Configuration to define which certificates and Certificate Authorities (CA) can be used for different environments (e.g. Development, Test and Production). The Network Security Configuration on Android feature lets application developers customize their network security settings in a safe, declarative configuration file without modifying the application code.

Reference:

- https://developer.android.com/training/articles/security-config.html

Software Composition Analysis Test

The mobile application uses the following external and native libraries:

External

- org.tensorflow
- kotlinx.coroutines
- io.flutter
- okhttp3.internal
- defpackage
- dev.fluttercommunity
- com.yalantis
- com.journeyapps

Android Native

- androidx.room
- androidx.core
- androidx.versionedparcelable
- androidx.activity
- androidx.lifecycle
- androidx.annotation
- androidx.browser
- androidx.fragment
- androidx.window
- androidx.work
- androidx.appcompat
- androidx.startup
- androidx.savedstate
- android.support