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	APP ID	APP VERSION
ionic-sql	io.ionicsql.com	1.0

DEVICE TYPE	TEST STARTED	TEST FINISHED
Android	February 26th 2022, 04:43	February 26th 2022, 04:51

Mobile App Permissions and Privacy	OWASP Mobile Top 10 Security Test	Mobile App External Communications	Software Composition Analysis
4 PERMISSIONS	4 MAJOR RISKS FOUND	NO ISSUES FOUND	NO COMPONENTS FOUND

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 Permissions

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

INTERNET normal

Allows an application to create network sockets.

USE_BIOMETRIC normal

Allows an app to use device supported biometric modalities.

USE_FINGERPRINT normal

This constant was deprecated in API level 28. Applications should request USE_BIOMETRIC instead.

VIBRATE normal

Allows the application to control the vibrator.

OWASP Mobile Top 10 Security Test

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

HIGH RISKS	MEDIUM RISKS	LOW RISKS	WARNINGS
2	2	2	5

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 **'execSQL()'** found in file 'com/getcapacitor/community/database/sqlite/SQLite/UtilsSQLCipher.java':

```
[line 235:            compileStatement.execute();]
[line 236:            openDatabase2.rawexecSQL("SELECT sqlcipher_export('main',
'plaintext')");]
[line 237:            openDatabase2.rawexecSQL("DETACH DATABASE plaintext");]
[line 238:            openDatabase2.setVersion(version);]
```

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 ionic-sqlSQLite.db:

```
TABLES:
app_data
offline_login

RAW DUMP:
CREATE TABLE app_data ( id INTEGER PRIMARY KEY NOT NULL, key_name TEXT NOT NULL,
value_data TEXT NOT NULL, date_added INTEGER DEFAULT (strftime('%s', 'now')),
last_modified INTEGER );CREATE TABLE offline_login ( id INTEGER PRIMARY KEY NOT NULL,
login_date INTEGER DEFAULT (strftime('%s', 'now')), login_data TEXT NOT NULL,
last_modified INTEGER );
```

In file Web Data:

```
TABLES:
meta
sqlite_autoindex_meta_1
autofill
sqlite_autoindex_autofill_1
autofill_name
autofill_name_value_lower
credit_cards
sqlite_autoindex_credit_cards_1
autofill_profiles
sqlite_autoindex_autofill_profiles_1
autofill_profile_names
autofill_profile_emails
autofill_profile_phones
autofill_profiles_trash
masked_credit_cards
unmasked_credit_cards
server_card_metadata
server_addresses
server_address_metadata
autofill_sync_metadata
sqlite_autoindex_autofill_sync_metadata_1
autofill_model_type_state

RAW DUMP:
```

```
CREATE TABLE meta(key LONGVARCHAR NOT NULL UNIQUE PRIMARY KEY, value
LONGVARCHAR);CREATE TABLE autofill (name VARCHAR, value VARCHAR, value_lower VARCHAR,
date_created INTEGER DEFAULT 0, date_last_used INTEGER DEFAULT 0, count INTEGER DEFAULT
1, PRIMARY KEY (name, value));CREATE INDEX autofill_name ON autofill (name);CREATE
INDEX autofill_name_value_lower ON autofill (name, value_lower);CREATE TABLE
credit_cards ( guid VARCHAR PRIMARY KEY, name_on_card VARCHAR, expiration_month
INTEGER, expiration_year INTEGER, card_number_encrypted BLOB, date_modified INTEGER NOT
NULL DEFAULT 0, origin VARCHAR DEFAULT '', use_count INTEGER NOT NULL DEFAULT 0,
use_date INTEGER NOT NULL DEFAULT 0, billing_address_id VARCHAR);CREATE TABLE
autofill_profiles ( guid VARCHAR PRIMARY KEY, company_name VARCHAR, street_address
VARCHAR, dependent_locality VARCHAR, city VARCHAR, state VARCHAR, zipcode VARCHAR,
sorting_code VARCHAR, country_code VARCHAR, date_modified INTEGER NOT NULL DEFAULT 0,
origin VARCHAR DEFAULT '', language_code VARCHAR, use_count INTEGER NOT NULL DEFAULT 0,
use_date INTEGER NOT NULL DEFAULT 0, validity_bitfield UNSIGNED NOT NULL DEFAULT
0);CREATE TABLE autofill_profile_names ( guid VARCHAR, first_name VARCHAR, middle_name
VARCHAR, last_name VARCHAR, full_name VARCHAR);CREATE TABLE autofill_profile_emails (
guid VARCHAR, email VARCHAR);CREATE TABLE autofill_profile_phones ( guid VARCHAR,
number VARCHAR);CREATE TABLE autofill_profiles_trash ( guid VARCHAR);CREATE TABLE
masked_credit_cards (id VARCHAR,status VARCHAR,name_on_card VARCHAR,network
VARCHAR,last_four VARCHAR,exp_month INTEGER DEFAULT 0,exp_year INTEGER DEFAULT 0,
bank_name VARCHAR, type INTEGER DEFAULT 0);CREATE TABLE unmasked_credit_cards (id
VARCHAR,card_number_encrypted VARCHAR, use_count INTEGER NOT NULL DEFAULT 0, use_date
INTEGER NOT NULL DEFAULT 0, unmask_date INTEGER NOT NULL DEFAULT 0);CREATE TABLE
server_card_metadata (id VARCHAR NOT NULL,use_count INTEGER NOT NULL DEFAULT 0,
use_date INTEGER NOT NULL DEFAULT 0, billing_address_id VARCHAR);CREATE TABLE
server_addresses (id VARCHAR,company_name VARCHAR,street_address VARCHAR,address_1
VARCHAR,address_2 VARCHAR,address_3 VARCHAR,address_4 VARCHAR,postal_code
VARCHAR,sorting_code VARCHAR,country_code VARCHAR,language_code VARCHAR, recipient_name
VARCHAR, phone_number VARCHAR);CREATE TABLE server_address_metadata (id VARCHAR NOT
NULL,use_count INTEGER NOT NULL DEFAULT 0, use_date INTEGER NOT NULL DEFAULT 0,
has_converted BOOL NOT NULL DEFAULT FALSE);CREATE TABLE autofill_sync_metadata
(model_type INTEGER NOT NULL, storage_key VARCHAR NOT NULL, value BLOB, PRIMARY KEY
(model_type, storage_key));CREATE TABLE autofill_model_type_state (model_type INTEGER
NOT NULL PRIMARY KEY, value BLOB);
```

CVSSv3 Base Score:

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

ENABLED APPLICATION BACKUP [M2] [CWE-921] [SAST]

MEDIUM

Description:

The mobile application uses external backup functionality (default Android backup mechanism) that may store inside sensitive data from the application. In certain conditions, this may lead to information disclosure (e.g. when a backup server or your Gmail account is compromised).

Example of insecure code:

```
android:allowBackup="true"
```

Example of secure code:

```
android:allowBackup="false"
```

Details:

There is **'android:allowBackup="true"'** found in file 'android/AndroidManifest.xml':

```
[line 6:     <application android:allowBackup="true"
android:appComponentFactory="androidx.core.app.CoreComponentFactory"
android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round" android:supportsRtl="true"
android:theme="@style/AppTheme">]
```

CVSSv3 Base Score:

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

Reference:

- https://developer.android.com/guide/topics/manifest/application-element.html
- http://resources.infosecinstitute.com/android-hacking-security-part-15-hacking-android-apps-using-backup-techniques/

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(true)'** found in file 'com/getcapacitor/Bridge.java':

```
[line 410:         settings.setJavaScriptEnabled(true);]
```

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://capacitorjs.com/docs/apis/splash-screen#hiding-the-splash-screen in following files:

- **com/capacitorjs/plugins/splashscreen/SplashScreen.java:**

```
[line 264: Logger.debug("SplashScreen was automatically hidden after the
launch timeout. You should call `SplashScreen.hide()` as soon as your web
app is loaded (or increase the timeout).Read more at
https://capacitorjs.com/docs/apis/splash-screen#hiding-the-splash-screen");]
[line 296: Logger.debug("SplashScreen was automatically hidden after the
launch timeout. You should call `SplashScreen.hide()` as soon as your web
app is loaded (or increase the timeout).Read more at
https://capacitorjs.com/docs/apis/splash-screen#hiding-the-splash-screen");]
```

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-v22/abc_alert_dialog_button_bar_material.xml
- 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/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/bridge_layout_main.xml
- android/res/layout/abc_action_menu_layout.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/select_dialog_item_material.xml
- android/res/layout/fragment_bridge.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/abc_expanded_menu_layout.xml
- android/res/layout/abc_cascading_menu_item_layout.xml
- android/res/layout/activity_main.xml
- android/res/layout/notification_template_custom_big.xml
- android/res/layout/fingerprint_dialog_layout.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 WebView'** found in file 'com/getcapacitor/CapacitorWebView.java':

```
[line 10: ]
[line 11: public class CapacitorWebView extends WebView {]
[line 12:     private BaseInputConnection capInputConnection;]
```

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/

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 'com/getcapacitor/BridgeWebChromeClient.java':

```
[line 394:         stringBuilder.append("_");]
[line 395:         return File.createTempFile(stringBuilder.toString(), ".jpg",
activity.getExternalFilesDir(Environment.DIRECTORY_PICTURES));]
[line 396:     }]
```

There is **'createTempFile()'** found in file 'com/getcapacitor/community/database/sqlite/SQLite/UtilsSQLCipher.java':

```
[line 224:         if (file.exists()) {]
[line 225:             File createTempFile = File.createTempFile("sqlcipherutils",
"tmp", context.getCacheDir());]
[line 226:             SQLiteDatabase openDatabase =
SQLiteDatabase.openDatabase(file.getAbsolutePath(), "", null, 0);]
```

Reference:

- https://developer.android.com/training/basics/data-storage/files.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 7:         <activity
android:configChanges="keyboard|keyboardHidden|locale|orientation|screenLayout|screenSi
ze|smallestScreenSize|uiMode" android:label="@string/title_activity_main"
android:launchMode="singleTask" android:name="io.ionicsql.com.MainActivity"
android:theme="@style/AppTheme.NoActionBarLaunch">]
[line 8:             <intent-filter>]
[line 9:                 <action android:name="android.intent.action.MAIN"/>]
```

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 'com/getcapacitor/AndroidProtocolHandler.java':

```
[line 49:     private static int getFieldId(Context context, String str, String str2)
throws ClassNotFoundException, NoSuchFieldException, IllegalAccessException {]
[line 50:         ClassLoader classLoader = context.getClassLoader();]
[line 51:         StringBuilder stringBuilder = new StringBuilder();]
```

```
[line 54:         stringBuilder.append(str);]
[line 55:         return
ClassLoader.loadClass(stringBuilder.toString()).getField(str2).getInt(null);]
[line 56:     }]
```

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

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

External Communications and Outgoing Traffic

Mobile Application Endpoints

Static mobile application security test revealed the following remote hosts where the mobile application may send or receive data:

Hostname	IP:Port	SSL Encryption	Websec Server Security	Domain Domain Security
capacitorjs.com:443	76.223.125.115:443	A+	Not Tested Yet	Not Tested Yet

Software Composition Analysis Test

The mobile application seems not to use any external or native libraries.

External
None

Native
None