Weakness/vunlerability	File path	Description	Auditor recommendation
WEAK HASHING ALGORITHMS	org/odk/collect/android/database/ItemsetDbAdapter.java org/odk/collect/android/utilities/EncryptionUtils.java org/odk/collect/android/utilities/FileUtils.java org/odk/collect/android/utilities/QRCodeUtils.java	The mobile application uses weak hashing algorithms (MD5) that can be vulnerable to collisions and other security weaknesses, and should not be used when reliable hashing of data is required	Use secure hashing algorithm like SHA-256
JS ENABLED IN A WEBVIEW	org/odk/collect/android/activities/WebViewActivity.java org/odk/collect/android/widgets/SelectImageMapWidget.java	The mobile application has enabled JavaScript in WebView and it can bring various JS-related security issues, such as Cross-Site Scripting (XSS) attacks.	Disable Javascript in WebView
TEMPORARY FILE CREATION	androidx/multidex/MultiDexExtractor.java org/odk/collect/android/utilities/FormDefCache.java org/odk/collect/android/utilities/FormDownloader.java com/journeyapps/barcodescanner/CaptureManager.java	The mobile application creates temporary files.	It is recommended to make sure that temporary files are securely deleted when they are not required by the application anymore
EXPORTED CONTENT PROVIDERS WITH INSUFFICIENT PROTECTION	android/AndroidManifest.xml	Content providers are normally used to share data between different applications. If exported without due protection, any application installed on the device, including malicious ones, will be able to disclose vulnerable application's data, including any confidential information contained therein.	Restrict access to it by setting up "android: protectionLevel" or "android: grantUriPermissions" or change exported value to False
MISSING TAPJACKING PROTECTION	see **XML index** sheet	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.	Use android:filterTouchesWhenObscured="true"
PREDICTABLE RANDOM NUMBER GENERATOR	org/osmdroid/tileprovider/tilesource/BitmapTileSourceBase.java org/odk/collect/android/tasks/sms/models/Message.java': org/odk/collect/android/tasks/sms/models/SmsSubmission.java org/javarosa/core/util/MathUtils.java org/javarosa/xform/parse/FisherYates.java	This weakness may jeopardize mobile application data encryption or other protection based on randomization.	Use **SecureRandom()**
EXTERNAL DATA STORAGE	see **Java index** sheet	The application's data stored on the external data storage may be accessed by other applications (including malicious ones)	Perform input validation and secure the files
EXTERNAL DATA IN RAW SQL QUERIES	org/odk/collect/android/spatial/MapHelper.java com/evernote/android/job/JobStorage.java org/odk/collect/android/database/helpers/FormsDatabaseHelper.java org/odk/collect/android/database/helpers/InstancesDatabaseHelper.java org/odk/collect/android/database/ItemsetDbAdapter.java org/osmdroid/tileprovider/modules/SqlTileWriter.java	Inclusion of input into raw SQL queries can potentially lead to a local SQL injection vulnerability in the mobile application	Use prepared SQL statements beyond user's control.