7-Android Permission

إذا كان التطبيق غير مصدَّر (not exported)، فهذا يعني أن مكونات التطبيق (components) مثل الأنشطة (activities)، الخدمات (services)، وأجهزة استقبال البث (broadcast receivers) لن تكون متاحة للتطبيقات الخارجية، ولن يتمكن المهاجم من استدعائها مباشرةً.

ومع ذلك، إذا كان التطبيق مُصدَّرًا (exported) عمدًا أو بالخطأ، فإن هذا يفتح الباب أمام مهاجم لاستغلال هذه المكونات. لذلك، من المهم إضافة طبقات حماية مثل Permissions لتقييد الوصول وضمان أن الجهات المصرح لها فقط هي التي يمكنها استخدام هذه المكونات.

دي permission احنا بنستخدمه `android:protectionLevel` بنحددها لكل

المستوى	المعنى	كيف يعمل	مثال
`normal`	إذن ذو مخاطر منخفضة. يمنح الوصول إلى ميزات على مستوى التطبيق بدون مخاطر كبيرة.	يتم منحه تلقائيًا عند التثبيت بدون طلب موافقة المستخدم.	إذن تغيير الخلفية.
`dangerous`	إذن ذو مخاطر عالية. يسمح بالوصول إلى بيانات المستخدم الخاصة أو التحكم في الجهاز.	**يتطلب موافقة صريحة من المستخدم عبر نافذة طلب.**	**إذن الوصول إلى الموقع (GPS) أو قراءة جهات الاتصال.**
`signature`	**يمنح فقط إذا كان التطبيق المطلوب موقَّعًا بنفس شهادة التطبيق الذي أعلن الإذن.**	**يتم منح الإذن تلقائيًا إذا تطابق التوقيع، بدون إعلام المستخدم.**	**مشاركة البيانات بين تطبيقات تابعة لنفس الشركة.**
`knownSigner`	يمنح فقط إذا كان التطبيق المطلوب موقَّعًا بشهادة معترف بها (قائمة شهادات معروفة).	يتم فحص شهادة التطبيق، وإذا كانت معترف بها، يتم منح الإذن تلقائيًا بدون إعلام المستخدم.	بيئات مؤسسية مغلقة.
`signatureOrSystem` *(مهمل)*	يمنح للتطبيقات الموقَّعة بنفس شهادة التطبيق الذي أعلن الإذن أو التطبيقات المضمنة في النظام (system image).	يتم منح الإذن للتطبيقات الموقَّعة أو تلك الموجودة في مجلد خاص على صورة النظام. *تم إهماله لاعتبارات الأمان بدءًا من API 23.*	مشاركة وظائف بين تطبيقات نظامية.

هنتكلم بقي عن Permission

---

1. Permissions for Location

Permission	Description	Protection Level
`ACCESS_FINE_LOCATION`	Access precise location using GPS or network.	Dangerous
`ACCESS_COARSE_LOCATION`	Access approximate location using Wi-Fi or cell towers.	Dangerous
`ACCESS_BACKGROUND_LOCATION`	Access location in the background (introduced in Android 10).	Dangerous

2. Permissions for Storage

Permission	Description	Protection Level
`READ_EXTERNAL_STORAGE`	Read files from the external storage.	Dangerous
`WRITE_EXTERNAL_STORAGE`	Write files to the external storage.	Dangerous
`MANAGE_EXTERNAL_STORAGE`	Access/manage all files in external storage (Android 11+).	Dangerous

3. Permissions for Camera and Microphone

Permission	Description	Protection Level
`CAMERA`	Access and use the device's camera.	Dangerous
`RECORD_AUDIO`	Record audio using the microphone.	Dangerous

4. Permissions for Phone and Contacts

Permission	Description	Protection Level
`READ_PHONE_STATE`	Access phone state (e.g., in a call, network info).	Dangerous
`CALL_PHONE`	Make phone calls directly without user intervention.	Dangerous
`READ_CONTACTS`	Access the user's contacts.	Dangerous
`WRITE_CONTACTS`	Modify the user's contacts.	Dangerous

5. Permissions for Messaging and Call Logs

Permission	Description	Protection Level
`READ_SMS`	Access and read SMS messages.	Dangerous
`SEND_SMS`	Send SMS messages.	Dangerous

Permission	Description	Protection Level
`READ_CALL_LOG`	Access the user's call history.	Dangerous
`WRITE_CALL_LOG`	Modify the user's call history.	Dangerous

6. Permissions for Internet and Network

Permission	Description	Protection Level
`INTERNET`	Access the internet.	Normal
`ACCESS_NETWORK_STATE`	Access information about networks.	Normal
`ACCESS_WIFI_STATE`	Access information about Wi-Fi networks.	Normal
`CHANGE_WIFI_STATE`	Change Wi-Fi connectivity settings.	Dangerous

7. Permissions for Device and System

Permission	Description	Protection Level
`BLUETOOTH`	Access Bluetooth features.	Normal
`BLUETOOTH_ADMIN`	Discover and pair Bluetooth devices.	Dangerous
`VIBRATE`	Control the device's vibration feature.	Normal
`WAKE_LOCK`	Prevent the device from sleeping.	Normal

8. Permissions for Sensors

Permission	Description	Protection Level
`BODY_SENSORS`	Access data from body sensors like heart rate monitors.	Dangerous
`ACTIVITY_RECOGNITION`	Access physical activity recognition data.	Dangerous

9. Permissions for Background Activities

Permission	Description	Protection Level
`FOREGROUND_SERVICE`	Run a foreground service.	Normal
`REQUEST_IGNORE_BATTERY_OPTIMIZATIONS`	Ignore battery optimization for certain tasks.	Normal

مثلا لوعاوزين نعمل permission access to content

1- in AndroidMenifast add this permission

```
<uses-permission android:name="android.permission.READ_CONTACTS"/>
```

```
    <uses-permission android:name="android.permission.READ_CONTACTS"/>
```

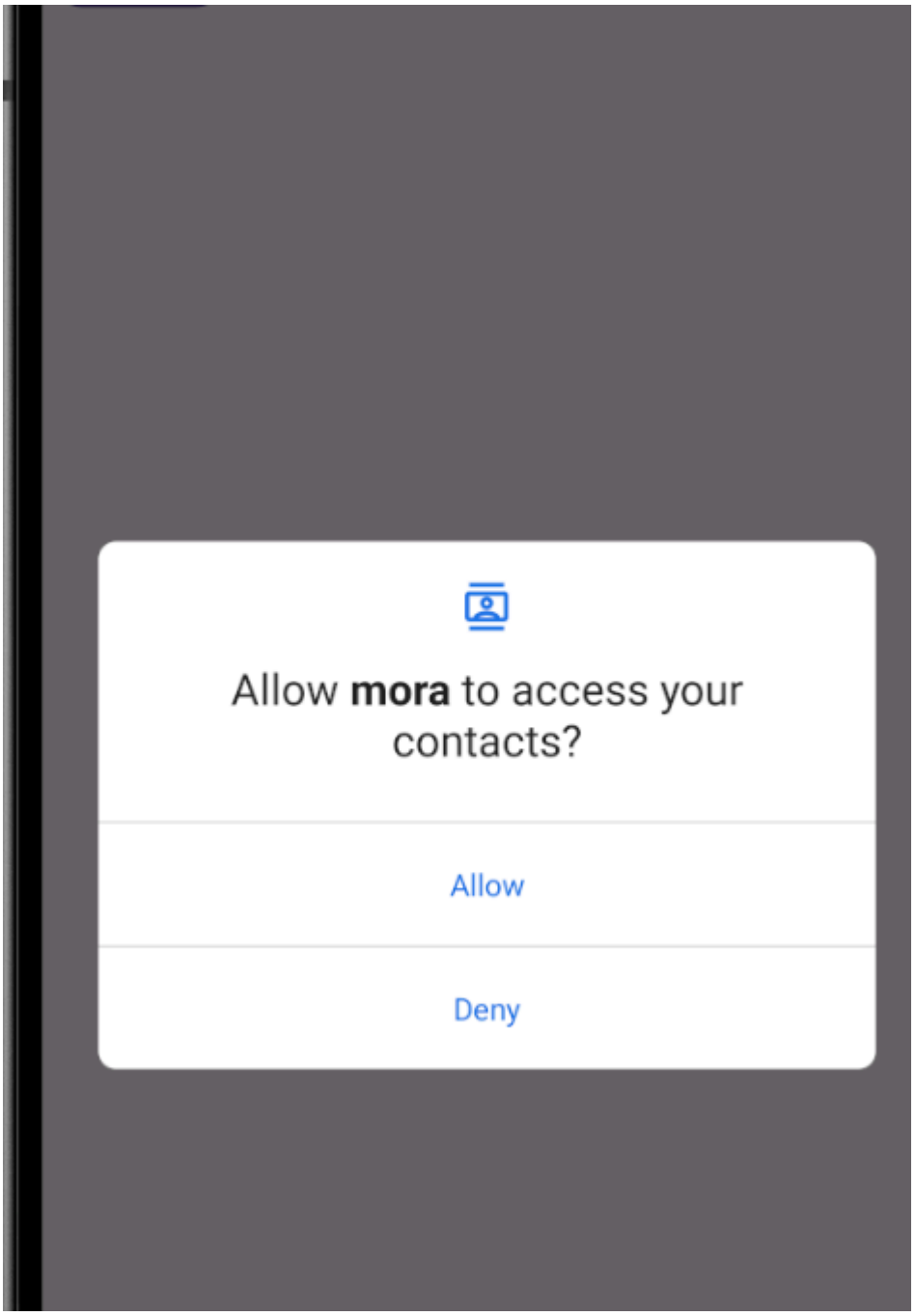
2- check the permission on MainActivity

```
                Log.i("Test","CONTACTS access?
"+checkSelfPermission("android.permission.READ_CONTACTS"));
```

1- غير مسموح ان احنا نقرأCONTACTS

```
---------------- PROCESS STARTED (26455) for package com.example.mora ----------------------------
00:41:08.231 26455-26455 Test                com.example.mora                I  CONTACTS access? -1
```

```
 if(checkSelfPermission("android.permission.READ_CONTACTS")==-1){
                    requestPermissions(new String[]
{"android.permission.READ_CONTACTS"},42);
                }
```



لو بقي عملنا allow وشغلناه تاني هنلاقي ان هو سمح ان نستخدم CONTACTS

```
91 27000-27000 Test                com.example.mora                I  CONTACTS access? 0
68 27000-27000 Test                com.example.mora                I  CONTACTS access? 0
```

1.
    - التطبيق يحتوي على إذن مثل `INTERNET` يسمح له بتنزيل الملفات من عناوين URL.
    - التطبيق لا يتحقق من سلامة الروابط أو أمانها قبل تنزيلها.
    - المستخدم لا يتم إبلاغه أو طلب موافقته قبل بدء عملية التنزيل.
2. **خطوات الاستغلال:**
    - المهاجم يُرسل رابط ضار (malicious URL) يحتوي على ملف خبيث (مثل RAT أو keylogger).
    - التطبيق يقوم بتنزيل الملف مباشرةً دون إشعار المستخدم.

- يتم تشغيل الملف (إذا كان التطبيق يدعم التشغيل التلقائي)، مما يتيح تنفيذ الأكواد الضارة.

3. **التأثير:**
   - تحميل برمجيات خبيثة إلى جهاز المستخدم.
   - إنشاء باب خلفي (backdoor) للوصول غير المصرح به.
   - تسريب بيانات المستخدم أو السيطرة على الجهاز.

لو عاوزين بقي نستخدم permission علشان مثلا لو هو exported واحنا مش عاوزين حد يستخدمه بنستخدم `"android.permission.BIND_JOB_SERVICE"`

- الإذن `BIND_JOB_SERVICE` يُستخدم لتقييد التطبيقات التي يمكنها الارتباط بخدمة `JobService` داخل تطبيق معين.
- **السماح فقط للتطبيقات المصرح لها باستخدام خدمة جدولة المهام (Job Scheduling).**
- **تطبيق جدولة المهام:**
  - عندما تُقدّم خدمة تُدير مهام مجدولة (مثل تنزيل ملفات في وقت محدد)، يمكن حماية الخدمة من التطبيقات الخارجية باستخدام هذا الإذن

```
<service
    android:name=".MyJobService"
    android:permission="android.permission.BIND_JOB_SERVICE">
    <intent-filter>
        <action android:name="android.app.job.JobService" />
    </intent-filter>
</service>
```

احنا بقي لو لاقينا permission ده بنشوف ايه protect level بتاعه علشان لو معمول normal احنا كده نقدر نستغله