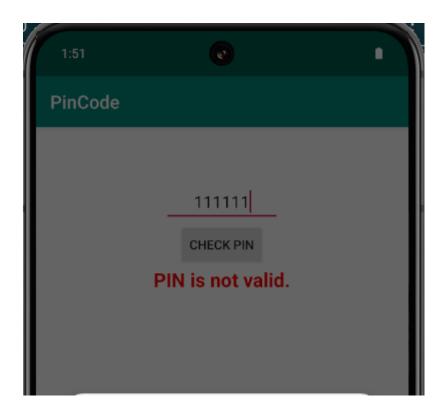
2-PinCode - reverse engineering

ده تاني تحدي معانا وهو ان احنا علشان نجيب flag لازم يكون pin صحيح واحنا مش عارفينه فلازم نروح نحلل app ونشوف ازاي pin بيتعملوا generation





1-open app with jadx-gui tool : decompile دي عبارة عن اداة بتحلل التطبيق من غير ما

jadx-gui pincode.apk

bypass activity or hack أول خاجة لازم نتاكد منها هو ملف ManiFest.xml ؟ علشان ممكن نستغله ان نحاول نعمل content provider or hacking broadcast receaiver or

2-open ManiFest.xml

بس مش هنلاقي في اي حاجة مهمة ممكن نستغلها

```
File View Navigation Tools Plugins Help
pincode.apk
                                     # AndroidManifest.xml ×
                                                                                                MainActivity ×
> 📭 Inputs
                                           <?xml version="1.0" encoding="utf-8"?>
Source code
                                           <manifest xmlns:android="http://schemas.android.com/apk/res/android"
android:versionCode="1"</pre>
  > 🖿 android
                                                android:versionName="1.0"
   > 🖿 androidx
                                                android:compileSdkVersion="28"
android:compileSdkVersionCodename="9"
  com.mobisec.pincode
     > @ BuildConfig
                                                package="com.mobisec.pincode"
platformBuildVersionCode="28"
     > 🧠 C0223R
     > 🕵 MainActivity
                                                platformBuildVersionName="9">
                                                <uses-sdk
      > © PinChecker
Resources
                                                android:targetSdkVersion="28"/>
<uses-permission android:name="android.permission.INTERNET"/>
  > META-INF
   > 🖿 res
                                                    android:theme="@style/AppTheme"
android:label="@string/app_name"
android:icon="@mipmap/ic_launcher'
android:debuggable="true"
AndroidManifest.xml
     dex
     aclasses2.dex
                                                     android:allowBackup="true"
android:supportsRtl="true"
   > 🖷 resources.arsc
   APK signature
                                                     android:roundIcon="@mipmap/ic_launcher_round"
android:appComponentFactory="android.support.v4.app.CoreComponentFactory">
<activity android:name="com.mobisec.pincode.MainActivity">

■ Summary

                                     22
23
24
26
23
22
                                                          <intent-filter>
                                                              <action android:name="android.intent.action.MAIN"/>
                                                          <category android:name="android.intent.category.LAUNCHER"/>
</intent-filter>
                                                     </activity>
                                           </application>
</manifest>
                                     13
```

3- open PinChecker

وده عبارة عن code اللي بيعمل code اللي المعمل

```
File View Navigation Tools Plugins Help
P Add files Pk
                             # AndroidManifest.xml ×
                                                         PinChecker >
                                                                            MainActivity
> Tilbaca
                                  package com.mobisec.pincode:
Source code
  > 🖿 android
                                  import android.content.Context;
                                  import android.util.Log;
  androidx
                                  import java.security.MessageDigest;

    com.mobisec.pincode

    > @ BuildConfig
                                   * loaded from: classes.dex */
    > 🧠 C0223R
                                  class PinChecker {
    MainActivity
                                      PinChecker() {
    PinChecker
 Resources
                             10
                                      public static boolean checkPin(Context ctx, String pin) {
  > META-INF
                             11
                                          if (pin.length() != 6) {
  > III res
                                              return false:
    AndroidManifest.xml
    _ classes.dex
                                             byte[] pinBytes = pin.getBytes();
                             17
    dex
                                             for (int i = 0; i < 25; i++) {
   for (int j = 0; j < 400; j++) }</pre>
  > 🖷 resources.arsc
                                                     MessageDigest md = MessageDigest.getInstance("MD5");
  APK signature
                                                     md.update(pinBytes);
byte[] digest = md.digest();

■ Summary

                             22
                             23
                                                     pinBytes = (byte[]) digest.clone();
                                             String hexPinBytes = toHexString(pinBytes);
                             26
                                              return hexPinBytes.equals("d04988522ddfed3133cc24fb6924eae9");
                                         } catch (Exception e) {
   Log.e("MOBISEC", "Exception while checking pin");
                             35
                                             return false;
                                      public static String toHexString(byte[] bytes) +
                             40
                                           tringBuilder hexString = new StringBuilder();
                             41
                                          for (byte b : bytes) {
                             44
                                                 ing hex = Integer.toHexString(b & 255);
                                              if (hex.length() == 1)
                             45
                                                 hexString.append('0');
                             46
                             48
                                             hexString.append(hex):
                                          return hexString.toString();
```

analyze code java

1- code length must be = 6

```
if (pin.length() != 6) {
    return false;
}
```

2- encode the PinCod

- convert code into byte --> pin.getBytes()
- use for loop (25 * 400) times
- identify the hash type --> MessageDigest.getInstance("MD5");
- update PinByte --> کبیر ان هی تجزءه byte کبیر ان می تجزءه
- apply md5 hash for PinByte --> digest.clone()
- get a clone after encode --> (byte[]) digest.clone()

الخلاصة هنا هو بيستخدم for loop 400*25 مرة وكل مرة بيعمل encode by md5 hash for new value يعني كل hash value يعني كل hash value

بعد كده اخر قيمة لل hash هنحولها ل hex باستخدام الفاتكشن اللي هو عملها create ويقارنها مع القيمة النهاءية اللي هي d04988522ddfed3133cc24fb6924eae9

```
try {
    byte[] pinBytes = pin.getBytes();
    for (int i = 0; i < 25; i++) {
        for (int j = 0; j < 400; j++) {
            MessageDigest md = MessageDigest.getInstance("MD5");
            md.update(pinBytes);
            byte[] digest = md.digest();
            pinBytes = (byte[]) digest.clone();
        }
    }
    String hexPinBytes = toHexString(pinBytes);
    return hexPinBytes.equals("d04988522ddfed3133cc24fb6924eae9");
}</pre>
```

دلوقتي بقي عرفنا ان قيمة hash لل PinCode المسح هي دي PinCode المسح هي دي d04988522ddfed3133cc24fb6924eae9

3-analyze toHexString function

هنا هو بياخد قمية byte وبيمشي علي كل byte ويحولها ل

```
public static String toHexString(byte[] bytes) {
    StringBuilder hexString = new StringBuilder();
    for (byte b : bytes) {
        String hex = Integer.toHexString(b & 255);
        if (hex.length() == 1) {
            hexString.append('0');
        }
        hexString.append(hex);
    }
    return hexString.toString();
}
```

هنا بقي خلينا نفهم method دي : دي بتاخد byte وتحوله ل integer وبعد كده بتقارنه مع 255 اللي هو علار عن and operation عشان لو دخلنا قيمة سالبة يحولها وهكذا

```
String hex = Integer.toHexString(b & 255)
```

Solution

دلوقتي احنا عارفين ان الرقم عبارة عن digit 6 ف احنا ننشيء script ان هو ينشء ارقام من 999999 : 000000 ونعمل نفس الخطوات بتاع تشفير كل رقم زي ما هو عملها وشنوف القيمة اللي هطلع لو بتساوي القيمة دي d04988522ddfed3133cc24fb6924eae9 يبقى فعلا هو ده الرقم الصح

create python script

```
import hashlib
import itertools
result="d04988522ddfed3133cc24fb6924eae9"
for code in itertools.product(range(10),repeat=6):
    pincode=""
    for x in code:
        pincode+=str(x)
    pincode=pincode.encode()
    for i in range(0,400*25):
        md5=hashlib.md5()
        md5.update(pincode)
        pincode=md5.digest()
    if pincode.hex()==result:
        print(code)
        break
```

correct pin is --> **703958**

check pin



Flag is ---> MOBISEC{local_checks_can_be_very_bad_for_security}