

BÁO CÁO THỰC HÀNH

Môn học: Bảo mật Web và Ứng dụng

Lab 4: Pentesting Android Applications

GVHD: Ngô Khánh Khoa

Nhóm: 6

1. THÔNG TIN CHUNG:

Lớp: NT213.P11.ANTT.2

STT	Họ và tên	MSSV	Email
1	Lại Quan Thiên	22521385	22521385@gm.uit.edu.vn
2	Mai Nguyễn Nam Phương	22521164	22521164@gm.uit.edu.vn
3	Hồ Diệp Huy	22520541	22520541@gm.uit.edu.vn
4	Nguyễn Phúc Nhi	22521041	22521041@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Nội dung	Tình Trạng	Thực hiện
1	Bài 1	100%	Phúc Nhi
2	Bài 2	100%	Nam Phương
3	Bài 3	100%	Quan Thiên
4	Bài 4	100%	Diệp Huy
5	Bài 5	100%	Quan Thiên
6	Bài 6	100%	Nam Phương
7	Bài 7	100%	Diệp Huy
8	Challenge 1 – Level 1	100%	Phúc Nhi
9	Challenge 1 – Level 2	100%	Nam Phương
10	Challenge 1 – Level 3	100%	Phúc Nhi
11	Challenge 1 – Level 4	100%	Quan Thiên
12	Challenge 1 – Level 5	100%	Diệp Huy
13	Challenge 1 – Level 6	100%	Quan Thiên

¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành

14	Challenge 1 – Level 7	100%	Phúc Nhi
15	Challenge 1 – Level 8	100%	Diệp Huy
16	Challenge 1 – Level 9	100%	Nam Phương
17	Challenge 1 – Level 10	100%	Diệp Huy
18	Challenge 1 – Level 11	100%	Phúc Nhi
19	Challenge 1 – Level 12	100%	Quan Thiên
20	Challenge 2 – Level 1	100%	Nam Phương
21	Challenge 2 – Level 2	100%	Diệp Huy
22	Challenge 2 – Level 3	100%	Phúc Nhi
23	Challenge 2 – Level 4	100%	Nam Phương
24	Challenge 2 – Level 5	100%	Quan Thiên

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

BÁO CÁO CHI TIẾT

BÀI TẬP CĂN BẢN

- Sau khi cài đặt và thiết lập môi trường của MobSF ta sẽ truy cập localhost:8000 và tải lên hoặc kéo thả tập tin InsecureBankv2.apk MobSF để tiến hành phân tích tĩnh.

- Ta thấy có một số vấn đề đang hiển thị trạng thái nguy hiểm (dangerous).

PERMISSION	STATUS	INFO	DESCRIPTION	CODE MAPPINGS
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network-based) location	Access coarse location sources, such as the mobile network database, to determine an approximate phone location, where available. Malicious applications can use this to determine approximately where you are.	
android.permission.ACCESS_NETWORK_STATE	normal	view network status	Allows an application to view the status of all networks.	
android.permission.GET_ACCOUNTS	dangerous	list accounts	Allows access to the list of accounts in the Accounts Service.	
android.permission.INTERNET	normal	full Internet access	Allows an application to create network sockets.	
android.permission.READ_CONTACTS	dangerous	read contact data	Allows an application to read all of the contact (address) data stored on your phone. Malicious applications can use this to send your data to other people.	
android.permission.READ_PROFILE	dangerous	read the user's personal profile data	Allows an application to read the user's personal profile data.	
android.permission.SEND_SMS	dangerous	send SMS messages	Allows application to send SMS messages. Malicious applications may cost you money by sending messages without your confirmation.	
android.permission.USE_CREDENTIALS	dangerous	use the authentication credentials of an account	Allows an application to request authentication tokens.	
android.permission.WRITE_EXTERNAL_STORAGE	dangerous	read/modify/delete external storage contents	Allows an application to write to external storage.	

- Ta thấy có một số vấn đề có mức độ nghiêm trọng cao (high).

- Kéo thả tập tin apk vào ByteCode Viewer để phân tích và hiển thị code

```

1 package com.android.insecurebankv2;
2
3 import android.app.Activity;
4 import android.content.Intent;
5 import android.content.SharedPreferences;
6 import android.net.Uri;
7 import android.preference.PreferenceManager;
8 import android.view.Menu;
9 import android.view.MenuItem;
10 import android.widget.Toast;
11 import java.io.BufferedReader;
12
13 public class DoLogin extends Activity {
14     public static final String MYREFS = "mySharedPreferences";
15     String password;
16     String protocol = "HTTP://";
17     BufferedReader reader;
18     String rememberme_password;
19     String rememberme_username;
20     String responseString = null;
21     String result;
22     SharedPreferences serverDetails;
23     String serverip = "";
24     String serverport = "";
25     String superSecurePassword;
26     String username;
27
28     public void callPreferences() {
29         this.startActivity(new Intent(this, FilePrefActivity.class));
30     }
31
32     protected void onCreate(Bundle var1) {
33         super.onCreate(var1);
34         this.setContentView(2130968602);
35         this.finish();
36         this.serverip = PreferenceManager.getDefaultSharedPreferences(this).getString("serverip", (String)null);
37         this.serverport = this.serverip.getString("serverport", (String)null);
38         if ((this.serverip != null & this.serverport != null) &
39             (Intent var2 = this.getIntent());
40             this.username = var2.getStringExtra("saved_username");
41             this.password = var2.getStringExtra("saved_password");
42             (new RequestTask(this)).execute(new String[]{this.username}));
43     }
44 }

```

- Sử dụng Malicious Code Scanner ở mục Plugins để quét các đoạn code có vẻ nguy hiểm.

```

public class com.android.insecurebankv2.DoLogin extends Activity {
    public static final String MYPREFS = "mySharedPreferences";
    ...
    public void callPreferences() {
        Intent var1 = new Intent(this, FilePrefActivity.class);
        ...
    }
}

```

- Sau khi scan ta nhìn thấy đoạn code lạ ở đường dẫn:
com/android/insecurebankv2/DoLogin\$RequestTask.class

```

public void postData(String var1) throws ClientProtocolException, IOException, JSONException, InvalidKeyException, NoSuchAlgorithmException, NoSuchPaddingException, {
    DefaultHttpClient var10 = new DefaultHttpClient();
    HttpPost var2 = new HttpPost(this.this$.protocol + this.this$.serverip + ":" + this.this$.serverport + "/login");
    HttpPost var5 = new HttpPost(this.this$.protocol + this.this$.serverip + ":" + this.this$.serverport + "/logout");
    ArrayList var3 = new ArrayList();
    var3.add(new BasicNameValuePair("username", this.this$.username));
    var3.add(new BasicNameValuePair("password", this.this$.password));
    HttpResponse var5;
    if (this.this$.username.equals("admin")) {
        var2.setEntity(new UrlEncodedFormEntity(var3));
        var6 = var4.execute(var5);
    } else {
        var2.setEntity(new UrlEncodedFormEntity(var3));
        var6 = var4.execute(var2);
    }
    InputStream var7 = var6.getEntity().getContent();
    this.this$.result = this.convertStreamToString(var7);
    this.this$.result = this.this$.result.replace(" ", "%20");
    if (this.this$.result != null) {
        Intent var8;
        if (this.this$.result.indexOf("correct credential") != -1) {
            Log.d("successful login", "accounts" + this.this$.username + ":" + this.this$.password);
            this.savecreds(this.this$.username, this.this$.password);
            this.trackUserLogins();
            var8 = new Intent(this.this$.getApplicationContext(), PostLogin.class);
            var8.putExtra("username", this.this$.username);
            this.this$.startActivity(var8);
        } else {
            var8 = new Intent(this.this$.getApplicationContext(), Wronglogin.class);
            this.this$.startActivity(var8);
        }
    }
}

```

Bài tập 1: Phân tích và chỉ ra điểm bất thường của đoạn code trên?

- Video thực hiện: <https://youtu.be/IppV92U2OJs>

Phân tích phương thức postData:

- Khởi tạo Http Client vàHttpPost request tới 2 url khác nhau
- Chuẩn bị Post data gồm các biến username và password
- Thực hiện gửi Post request, nếu username là devadmin thì Post request được gửi tới url .../devlogin các trường hợp còn lại gửi tới .../login.

=> Đây là một điểm bất thường trong đoạn code, khi mà username là devadmin thì request được gửi tới URL khác. Trong trường hợp này thì nếu ta nhập username là devadmin thì có thể login thẳng mà không cần tới password.

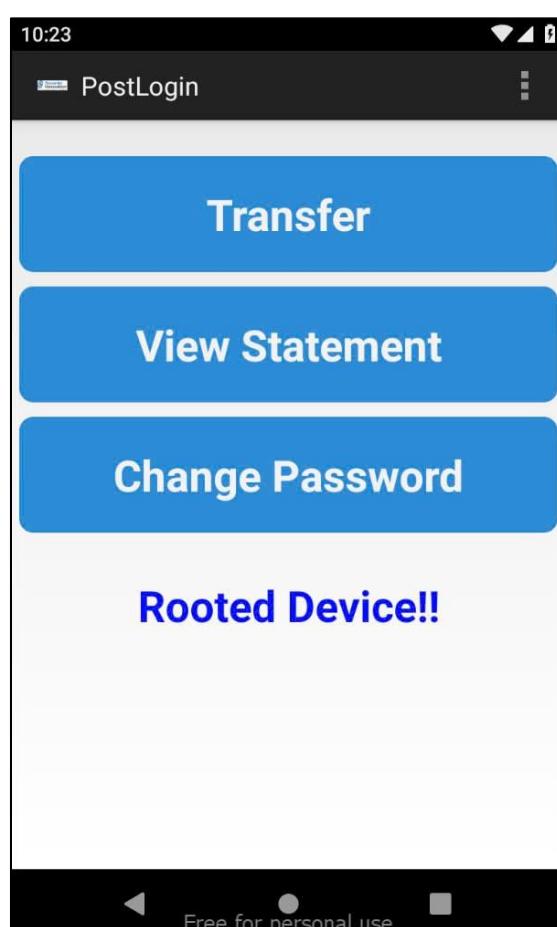
- Bên cạnh đó có thể thấy trong đoạn mã sử dụngHttpPost để post các request đăng nhập. Do phương thức HTTP không có mã hóa nên nếu bắt được gói thì username và password có thể bị đọc được.

Bài tập 2: Chỉ ra rằng dữ liệu lưu trữ có an toàn hay không?

- Video thực hiện: https://youtu.be/3gRFmc5p_bs

- Đầu tiên ta thực hiện đăng nhập vào ứng dụng bằng nhiều tài khoản có sẵn:

- + Đăng nhập vào tài khoản devadmin
- + Đăng nhập vào tài khoản dinesh





- Truy cập adb shell để truy cập vào giao diện dòng lệnh của thiết bị Android từ máy tính, thực hiện truy cập đến đường dẫn /data/data/com.android.insecurebankv2. Sau đó liên tục thực hiện các lệnh ls để kiểm tra các tệp và thư mục hiện có trong đường dẫn này thì ta sẽ thấy được các cơ sở dữ liệu của app ở đường dẫn /data/data/com.android.insecurebankv2/databases

```
C:\Users\namphuong>adb shell
genymotion:/ # cd /data/data/com.android.insecurebankv2
genymotion:/data/data/com.android.insecurebankv2 # ls
cache code_cache databases shared_prefs
genymotion:/data/data/com.android.insecurebankv2 # cd databases
genymotion:/data/data/com.android.insecurebankv2/databases # ls
mydb mydb-shm mydb-wal
```

- Thực hiện truy vấn các bảng bên trong csdl mydb, thấy bên trong có 2 table là names và android_metadata. Để thấy trong table names thì dữ liệu tên đăng nhập mà người dùng sử dụng trong quá trình đăng nhập được lưu lại dưới dạng plain text mà không được bảo vệ

=> Thông tin nhạy cảm lưu trữ không an toàn

```
genymotion:/data/data/com.android.insecurebankv2/databases # sqlite3 mydb
SQLite version 3.22.0 2019-09-03 18:36:11
Enter ".help" for usage hints.
sqlite> .database
main: /data/data/com.android.insecurebankv2/databases/mydb
sqlite> .table
android_metadata  names
sqlite> select * from names
...> ;
1|devadmin
2|devadmin
3|dinesh
4|devadmin
5|dinesh
6|dinesh
7|dinesh
8|devadmin
9|dinesh
sqlite> select * from android_metadata;
en_US
sqlite> _
```



Bài Tập 3: Kiểm tra xem thông tin nhạy cảm có lưu lại trên thiết bị hay không? Một số từ khóa: deviceId, userId, imei, deviceSerialNumber, devicePrint, phone, XDSN, mdn, IMSI, uuid...

Trả lời:

- Video thực hiện: <https://youtu.be/ehu3jP9jzHI>
- Dùng adb shell để vào command shell của điện thoại ảo sau đó di chuyển đến /data/data/com.android.insecurebankv2.
 - Chạy cấu trúc lệnh với Danh sách các từ khóa cần tìm kiếm: grep -r -E "deviceId|imei|userId|uuid|imei|deviceSerialNumber|devicePr|phone|XDSN|mdn|IMSI" \$(find)
 - Trong phần này thì ta không tìm được thông tin nhạy cảm nào cả.

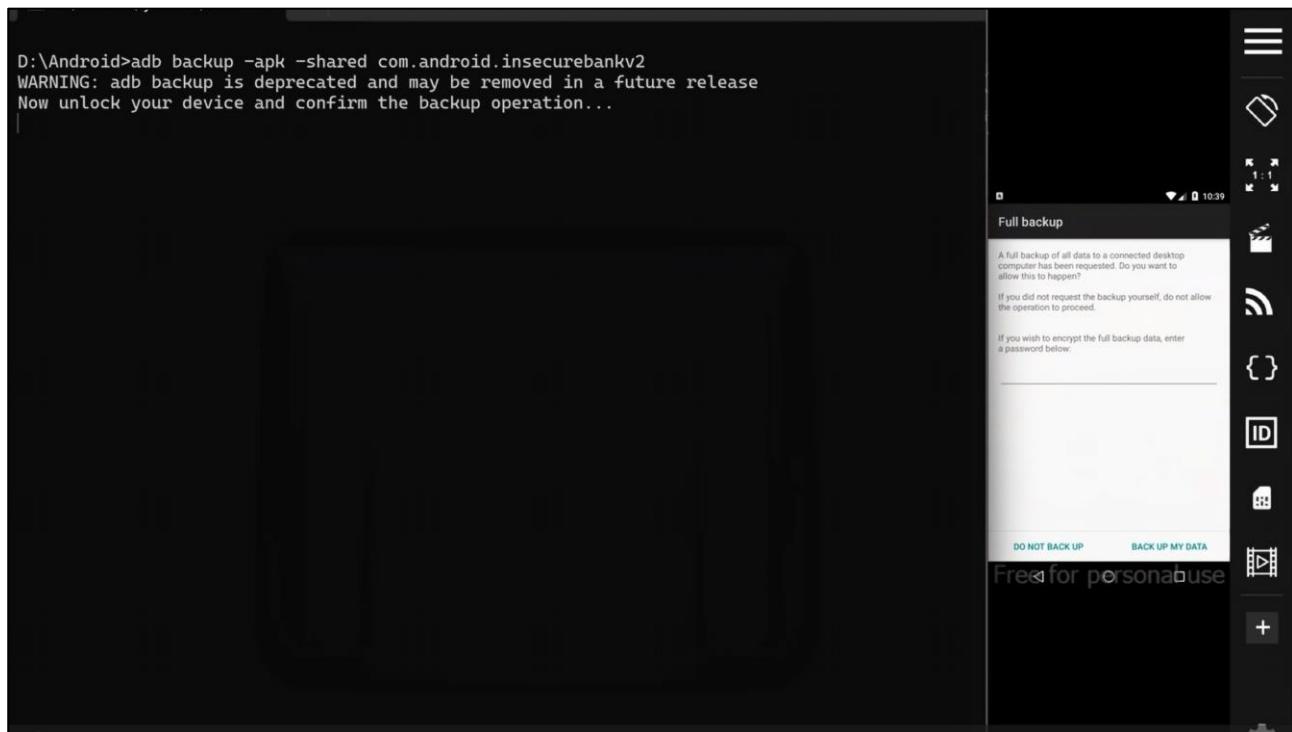
```
C:\Windows\System32\cmd.e + v
vbox86p:/data/data/com.android.insecurebankv2 # grep -r -E "deviceId|imei|userId|uuid|imei|deviceSerialNumber|devicePr>1|vbox86p:/data/data/com.android.insecurebankv2 # |
```

Bài tập 4: Theo bạn thư mục sao lưu chứa thông tin nào cần mã hoá, chỉ ra?

Trả lời:

- Video thực hiện: https://youtu.be/kNOb_PS8qj4

- Đầu tiên đăng nhập với tài khoản người dùng bình thường sau đó sử dụng lệnh adb backup -apk -shared com.android.insecurebankv2 để backup dữ liệu.



- Chuyển đổi tập tin sao lưu qua định dạng có thể đọc được (cài gói qpdf)

```
ubuntu@hohuy:~/Android$ cat backup.ab | (dd bs=24 count=0 skip=1; cat) | zlib-fl
ate -uncompress > backup_compressed.tar
0+0 records in
0+0 records out
0 bytes copied, 0.000236802 s, 0.0 kB/s
ubuntu@hohuy:~/Android$
```

- Kiểm tra các tập tin sao lưu sau khi giải nén, Ta có đoạn dữ liệu được mã hoá, ta có thể thấy thông tin của user đã được mã hóa.

```
ubuntu@hohuy:~/Android$ cd backup_compressed/
ubuntu@hohuy:~/Android/backup_compressed$ ls
apps shared
ubuntu@hohuy:~/Android/backup_compressed$ cd apps
ubuntu@hohuy:~/Android/backup_compressed/apps$ cd com.android.insecurebankv2/
ubuntu@hohuy:~/Android/backup_compressed/apps/com.android.insecurebankv2$ cd
a/ db/ sp/
ubuntu@hohuy:~/Android/backup_compressed/apps/com.android.insecurebankv2$ cd sp/
ubuntu@hohuy:~/Android/backup_compressed/apps/com.android.insecurebankv2/sp$ ls
com.android.insecurebankv2_preferences.xml mySharedPreferences.xml
ubuntu@hohuy:~/Android/backup_compressed/apps/com.android.insecurebankv2/sp$ ls -l
total 8
-rw-rw---- 1 ubuntu ubuntu 163 Thg 12 5 01:06 com.android.insecurebankv2_preferences.xml
-rw-rw---- 1 ubuntu ubuntu 221 Thg 12 5 01:45 mySharedPreferences.xml
ubuntu@hohuy:~/Android/backup_compressed/apps/com.android.insecurebankv2/sp$ cat mySharedPreferences.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="superSecurePassword">DTrW2VXjSoFdg0e61fHxJg==#10;      </string>
    <string name="EncryptedUsername">ZGluZXNo#13;&#10;      </string>
</map>
ubuntu@hohuy:~/Android/backup_compressed/apps/com.android.insecurebankv2/sp$ █
```

Bài tập 5: Viết chương trình giải mã đoạn dữ liệu mã hoá

Trả lời:

- Video thực hiện: <https://youtu.be/JRzytq6P4sY>

- Xem qua mã nguồn:

```
public class CryptoClass {
    String base64Text;
    byte[] cipherData;
    String cipherText;
    byte[] ivBytes = new byte[]{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
    String key = "This is the super secret key 123";
    String plainText;

    public static byte[] aes256decrypt(byte[] var0, byte[] var1, byte[] var2) throws UnsupportedEnc
        IvParameterSpec var4 = new IvParameterSpec(var0);
        SecretKeySpec var5 = new SecretKeySpec(var1, "AES");
        Cipher var3 = Cipher.getInstance("AES/CBC/PKCS5Padding");
        var3.init(2, var5, var4);
        return var3.doFinal(var2);
    }

    public static byte[] aes256encrypt(byte[] var0, byte[] var1, byte[] var2) throws UnsupportedEnc
        IvParameterSpec var4 = new IvParameterSpec(var0);
        SecretKeySpec var5 = new SecretKeySpec(var1, "AES");
        Cipher var3 = Cipher.getInstance("AES/CBC/PKCS5Padding");
        var3.init(1, var5, var4);
        return var3.doFinal(var2);
    }

    public String aesDecryptedString(String var1) throws UnsupportedEncodingException, InvalidKeyException
        byte[] var2 = this.key.getBytes("UTF-8");
        this.cipherData = aes256decrypt(this.ivBytes, var2, Base64.decode(var1.getBytes("UTF-8"), 0));
        this.plainText = new String(this.cipherData, "UTF-8");
        return this.plainText;
    }

    public String aesEncryptedString(String var1) throws UnsupportedEncodingException, InvalidKeyException
        byte[] var2 = this.key.getBytes("UTF-8");
        this.plainText = var1;
        this.cipherData = aes256encrypt(this.ivBytes, var2, this.plainText.getBytes("UTF-8"));
        this.cipherText = Base64.encodeToString(this.cipherData, 0);
        return this.cipherText;
    }
}
```

- Xem qua file được mã hoá:

```
[(kali㉿thinnnlinux)-[~/.../Lab_4/apps/com.android.insecurebankv2/sp]
$ cat mySharedPreferences.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="superSecurePassword">DTrW2VXjSoFdg0e61fHxJg===&#10;      </string>
    <string name="EncryptedUsername">ZGluZXNo&#13;&#10;      </string>
</map>
```

- Nhận xét: Đề bài đã cung cấp cho ta một đoạn mã Java dùng thuật toán AES-CBC-256 với đệm PKCS5Padding để mã hóa kèm theo đó là Key và IV 16 bytes 0 dùng để giải mã.

- Từ đó ta có thể viết một chương trình Python dùng để giải mã. Dưới đây là code:

```
from Crypto.Cipher import AES
from base64 import b64decode

def aes256_decrypt(iv, key, ciphertext):
    """
    Giải mã dữ liệu sử dụng AES-CBC-256 với padding PKCS5.
    """

    # ... (Implementation of AES-CBC-256 decryption with PKCS5 padding)
```

```

:param iv: Initialization vector (IV) 16 byte.
:param key: Secret key 32 byte.
:param ciphertext: Dữ liệu mã hóa dạng base64.
:return: Dữ liệu giải mã dạng chuỗi.

"""

# Decode dữ liệu ciphertext từ base64
ciphertext_bytes = b64decode(ciphertext)

# Tạo cipher với AES-CBC
cipher = AES.new(key.encode('utf-8'), AES.MODE_CBC, iv)

# Giải mã dữ liệu
decrypted_data = cipher.decrypt(ciphertext_bytes)

# Loại bỏ padding PKCS5
padding_length = decrypted_data[-1]
decrypted_data = decrypted_data[:-padding_length]

return decrypted_data.decode('utf-8')

# Dữ liệu từ mã Java
key = "This is the super secret key 123" # Chuỗi key 32 byte
iv = bytes([0] * 16) # IV 16 byte toàn số 0
encrypted_text = "DTrW2VXjSoFdg0e61fHxJg=="

# Thực hiện giải mã
try:
    decrypted_text = aes256_decrypt(iv, key, encrypted_text)
    print("Dữ liệu giải mã:", decrypted_text)
except Exception as e:
    print("Lỗi giải mã:", e)

```

- Kết quả giải mã thành công:

```

C:\Users\VanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4\Resources\AndroLabServer>"C:/Program Files/Python/Python310/python.exe" c
:/Users\VanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD/Thuc_hanh/Lab_4/task_5.py
Dữ liệu giải mã: Dinesh@123$


C:\Users\VanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4\Resources\AndroLabServer>

```

Bài tập 6: Sinh viên điều chỉnh mã nguồn ứng dụng sao cho luôn hiển thị trạng thái “Rooted Device!!” với bất kỳ trạng thái nào của thiết bị.

- Video thực hiện: <https://youtu.be/eQcOxibLtsM>
- Trước tiên thực hiện decompile file InsecureBankv2.apk bằng công cụ apktool

```
C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4>apktool d InsecureBankv2.apk
I: Using Apktool 2.10.0 on InsecureBankv2.apk with 16 thread(s).
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\namphuong\AppData\Local\apktool\framework\1.apk
I: Decoding values /* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

- Ta sẽ truy cập vào tập tin InsecureBankv2/smali/com/android/insecurebankv2/PostLogin.smali để sửa đổi tập tin, có thể thấy trong đoạn mã này thực hiện 1 lệnh if else để kiểm tra trạng thái của máy, sau đó dựa trên trạng thái đó đưa ra output

```

new 1
virt-install.sh
prepare_userdata.sh
backup.sh
ethercodes.txt
bmw.txt
requirements.txt
PostLogin.smali

445     move v0, v1
446
447     .line 88
448     .local v0, "isrooted":Z
449     :goto_0
450     if-ne v0, v1, :cond_2
451
452     .line 90
453     igeq-object v1, p0, Lcom/android/insecurebankv2/PostLogin;->root_status:Landroid/widget/TextView;
454
455     const-string v2, "Rooted Device!!"
456
457     invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
458
459     .line 96
460     :goto_1
461     return-void
462
463     .line 87
464     .end local v0      # "isrooted":Z
465     :cond_1
466     const/4 v0, 0x0
467
468     goto :goto_0
469
470     .line 94
471     .restart local v0      # "isrooted":Z
472     :cond_2
473     igeq-object v1, p0, Lcom/android/insecurebankv2/PostLogin;->root_status:Landroid/widget/TextView;
474
475     const-string v2, "Device not Rooted!!"
476
477     invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
478
479     goto :goto_1
480     .end method
481
482     .method protected viewStatement()V
483     .locals 3
484
485     .prologue
486     .line 142
487     new-instance v0, Landroid/content/Intent;
488
489     invoke-virtual {p0}, Lcom/android/insecurebankv2/PostLogin;->getApplicationContext()Landroid/content/Context;
490
491     move-result-object v1

```

- Ta sẽ thay đổi phần điều kiện máy chưa bị root, theo đó chuyển output đầu ra thành **Rooted Device (FAKE) !!** để dễ dàng phân biệt

```

442     if-eqz v2, :cond_1
443
444     :cond_0
445     move v0, v1
446
447     .line 88
448     .local v0, "isrooted":Z
449     :goto_0
450     if-ne v0, v1, :cond_2
451
452     .line 90
453     ige-object v1, p0, Lcom/android/insecurebankv2/PostLogin;->root_status:Landroid/widget/TextView;
454
455     const-string v2, "Rooted Device!!"
456
457     invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
458
459     .line 96
460     :goto_1
461     return-void
462
463     .line 87
464     .end local v0      # "isrooted":Z
465     :cond_1
466     const/4 v0, 0x0
467
468     goto :goto_0
469
470     .line 94
471     .restart local v0      # "isrooted":Z
472     :cond_2
473     ige-object v1, p0, Lcom/android/insecurebankv2/PostLogin;->root_status:Landroid/widget/TextView;
474
475     const-string v2, "Rooted Device (FAKE) !!"  
I
476
477     invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
478
479     goto :goto_1
480     .end method
481
482     .method protected viewStatement()V
483         .locals 3

```

- Sau khi chỉnh sửa xong tập tin PostLogin.smali thì lưu lại, sau đó ta sử dụng apktool để build lại một file apk mới có tên là SolvedInsecureBank

```

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4>apktool b InsecureBankv2 -o SolvedInsecureBank.apk
I: Using Apktool 2.10.0 with 16 thread(s).
I: Checking whether sources has changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources has changed...
I: Building resources...
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk into: SolvedInsecureBank.apk

```

- Android yêu cầu các tập tin APK đều phải được ký bằng một chứng chỉ trước khi được phép cài đặt trên thiết bị. Sau khi chỉnh sửa, tập tin APK sẽ không còn toàn vẹn như ban đầu nên cần phải được ký lại, vì thế ta sẽ tạo 1 key mới và kí vào file apk vừa tạo bằng jar signer

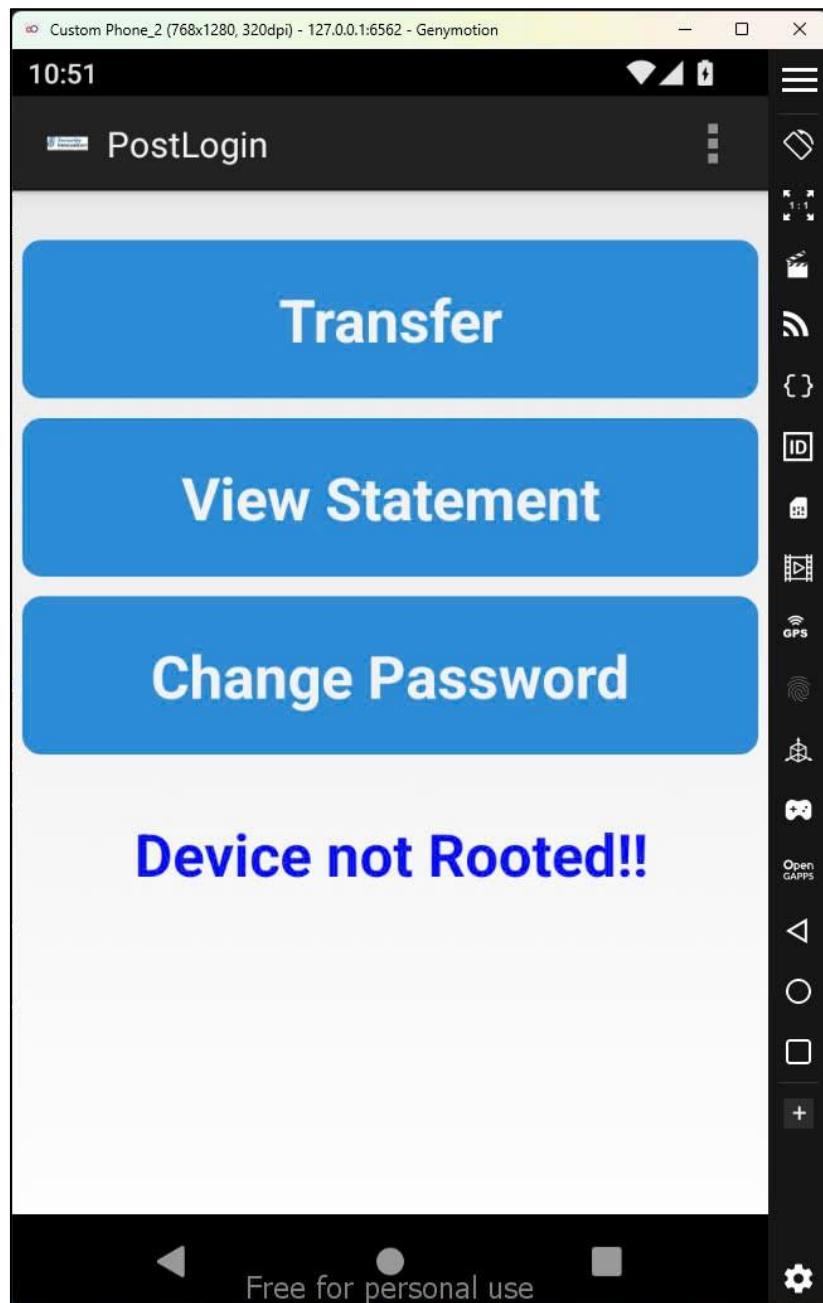
```

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4>jar signer -keystore "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4\key.keystore" -alias SolvedInsecureBank -keyalg RSA -keysize 2048 -vali
dity 1000
Enter keystore password:
Re-enter new password:
Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press ENTER to use the default value in braces.
What is your first and last name?
What is the name of your organizational unit?
What is the name of your organization?
What is the name of your City or locality?
What is the name of your State or Province?
What is the two-letter country code for this unit?
Is CN=Unknown, OU=Unknown, L=Unknown, C=Unknown correct?
Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press ENTER to use the default value in braces.
What is your first and last name?
What is the name of your organizational unit?
What is the name of your organization?
What is the name of your City or locality?
What is the name of your State or Province?
What is the two-letter country code for this unit?
Is CN=Unknown, OU=Unknown, L=Unknown, C=Unknown correct?
[INFO] Generating 2,048 bit RSA key pair and self-signed certificate (SHA384withRSA) with a validity of 1,000 days
    for: CN=Unknown, OU=Unknown, L=Unknown, ST=Unknown, C=Unknown
[Storing C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4\key.keystore]

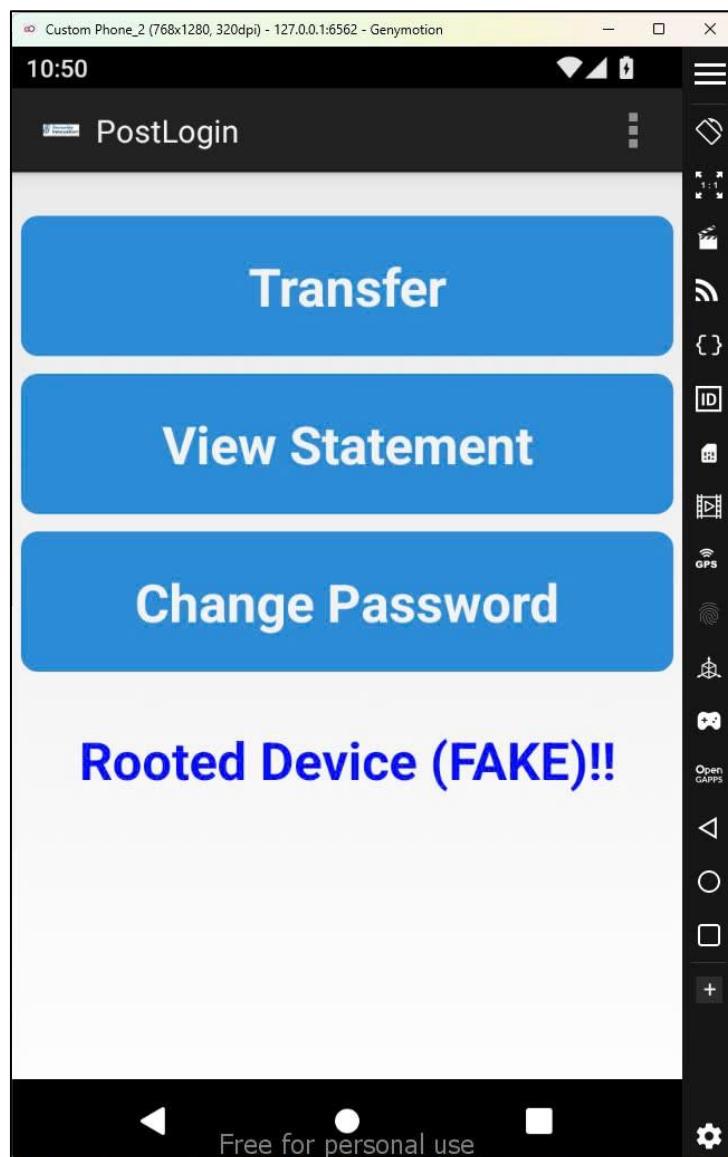
C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4>jar signer -keystore "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4\key.keystore" -storepass namphuong "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\LAB\W4\Lab4\SolvedInsecureBank.apk" SolvedInsecureBank
jar signed.
Warning:
The signer's certificate is self signed.

```

- Đầu tiên ta thực hiện kiểm tra kết quả nếu chưa qua chỉnh sửa trên một thiết bị chưa bị root sẽ có hiển thị như sau



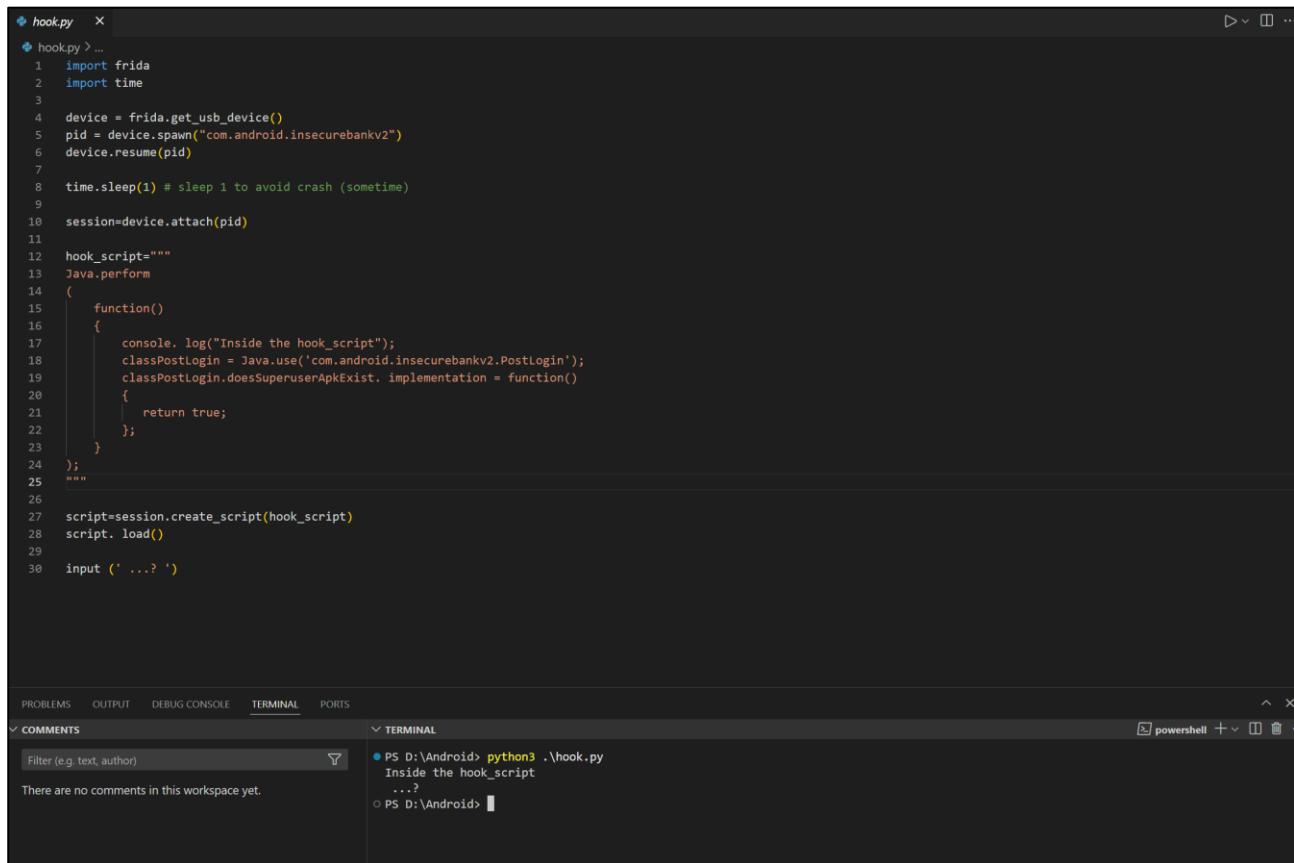
- Sau khi cài đặt lại file apk mới ta đã sửa thì được kết quả đúng như ta muốn đó là hiển thị “Rooted Device!!” trên bất kì thiết bị nào (phân biệt bằng cụm FAKE đã thêm)



Bài tập 7: Hoàn thiện đoạn code hook.py trên và demo.

Trả lời:

- Video thực hiện: <https://youtu.be/GVnfjni70CQ>
- Ta sẽ ghi đè method doesSuperuserApkExist() bằng cách sử dụng implementation.
- Khi doesSuperuserApkExist() == true thì root detected.

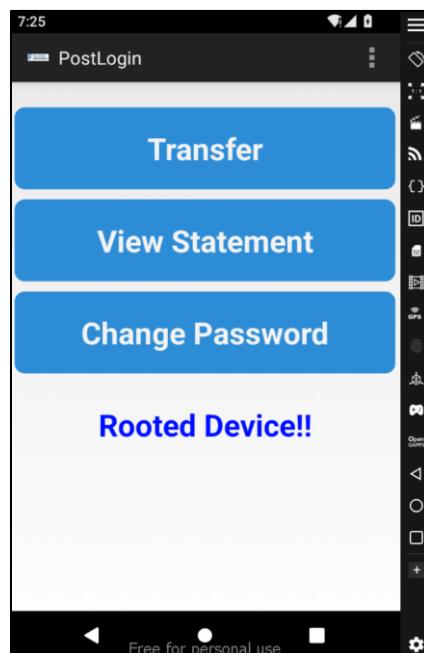


```

hook.py  x
hook.py > ...
1 import frida
2 import time
3
4 device = frida.get_usb_device()
5 pid = device.spawn("com.android.insecurebankv2")
6 device.resume(pid)
7
8 time.sleep(1) # sleep 1 to avoid crash (sometime)
9
10 session=device.attach(pid)
11
12 hook_script="""
13 Java.perform
14 (
15     function()
16     {
17         console.log("Inside the hook_script");
18         classPostLogin = Java.use('com.android.insecurebankv2.PostLogin');
19         classPostLogin.doesSuperuserApkExist_implementation = function()
20         {
21             return true;
22         };
23     }
24 );
25 """
26
27 script=session.create_script(hook_script)
28 script.load()
29
30 input ('...? ')

```

- Kết quả cho thấy ta root device thành công.



CHALLENGES 1**Level 1**

- Video thực hiện: https://youtu.be/a2SngSh_7fU
- Giao diện và hint của challenge (Debug Me), cho thấy secret key mà ta cần tìm đã được log lại, vì vậy lúc này ta chỉ cần đọc log của thiết bị thì sẽ có được giá trị cần tìm.



- Ta chạy lệnh `./adb shell ps` để tìm PID của ứng dụng (2447).

```

Windows PowerShell
+ + +
system      1720  367  13545116  87280  0          0 S com.genymotion.geny
system      1744  367  13610456  90124  0          0 S com.android.keychain
u0_a58     1763  367  13542672  83220  0          0 S com.android.quicksearchbox
u0_a36     1816  367  13545560  85748  0          0 S com.android.printspooler
system      1893  367  13543476  83868  0          0 S com.android.localtransport
u0_a56     1911  367  13568296  96472  0          0 S com.android.deskclock
u0_a19     1944  367  13541368  86132  0          0 S com.android.externalstorage
u0_a11     1973  367  13548600  94672  0          0 S android.process.media
u0_a45     1995  367  13552900  92272  0          0 S com.android.traceur
u0_a60     2023  367  13544304  83692  0          0 S com.android.calendar
u0_a64     2041  367  13923816  87166  0          0 S com.android.camera2
u0_a48     2046  367  13548788  98796  0          0 S com.android.imsserviceentitlement
u0_a50     2084  367  13550952  89704  0          0 S com.android.contacts
system      2108  367  13541464  81732  0          0 S com.android.dnsystem
system      2128  367  13542560  84784  0          0 S com.android.dnsystem:dnsystem
u0_a26     2143  367  13547236  86988  0          0 S com.android.managedprovisioning
u0_a53     2162  367  13554628  105456  0          0 S com.android.messaging
u0_a47     2185  367  13541192  82264  0          0 S com.android.onetimeinitializer
u0_a27     2283  367  13545260  86140  0          0 S com.android.packageinstaller
u0_a80     2219  367  13566864  102088  0          0 S com.android.permissioncontroller
u0_a20     2242  367  13544096  91380  0          0 S com.android.providers.calendar
shell      2261  367  13543836  85644  0          0 S com.android.shell
u0_a24     2278  367  13552896  94508  0          0 S com.android.statementservice
shell      2383  506  10887788  4188  0          0 S abb
u0_a63     2396  367  13875540  90368  0          0 S com.android.gallery3d
u0_a88     2407  367  13675988  146724  0          0 S com.revo.evabs
root      2544  2    0    0   0          0 I [kworker/u8:0-phy0]
root      2570  2    0    0   0          0 I [kworker/1:0-events]
shell      2581  506  10792836  3712  0          0 R ps
PS D:\BWMapp\platform-tools> |

```

- Chạy lệnh `./adb logcat --pid=2447`, nhấn vào button LOG THE KEY trên giao diện app và quay trở lại shell để tìm flag vừa xuất hiện.

Lab 4: Pentesting Android Applications

```

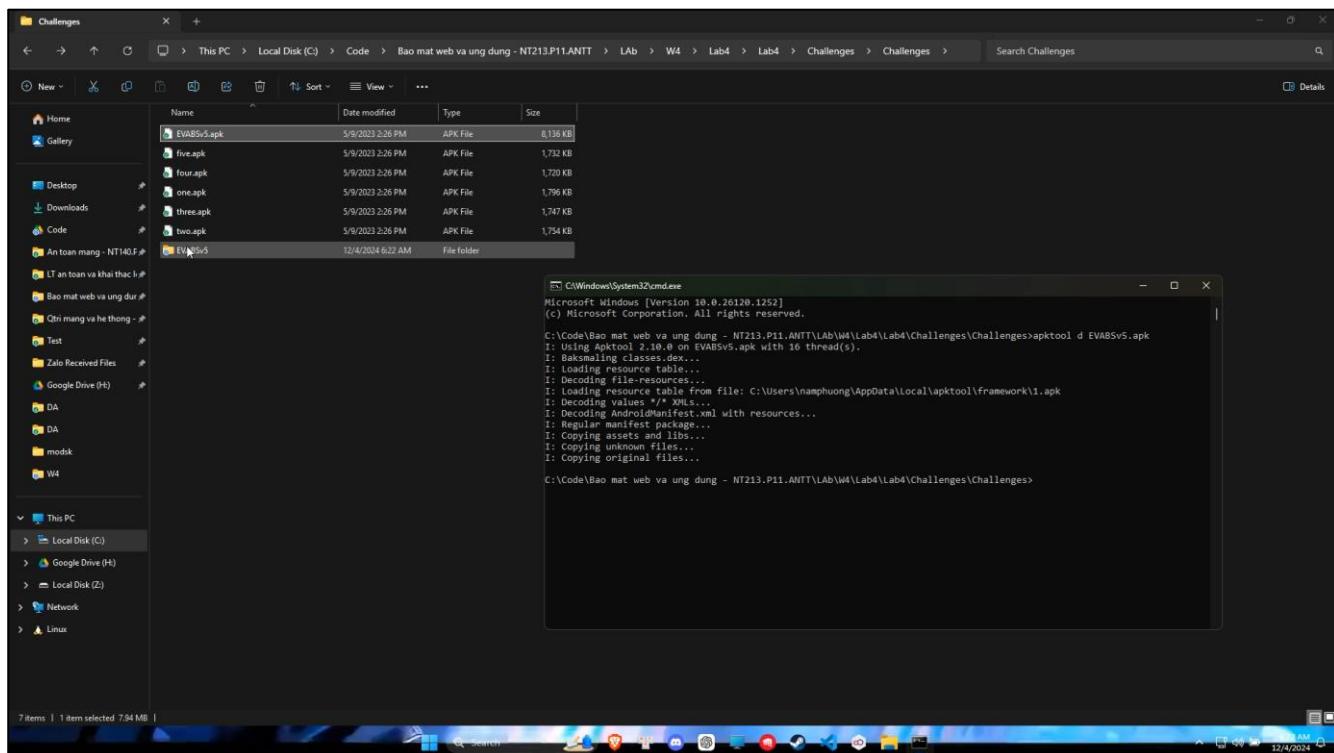
Windows PowerShell

2447 2447 D CompatibilityChangeReporter: Compat change id reported: 171228096; UID 10088; state: ENABLED
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 W Parcel : Expecting binder but got null!
2447 2469 D EGL_emulation: app_time_stats: avg=32.59ms min=8.41ms max=381.55ms count=28
2447 2469 D EGL_emulation: app_time_stats: avg=419.69ms min=59.60ms max=960.98ms count=3
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 W Parcel : Expecting binder but got null!
2447 2469 D EGL_emulation: app_time_stats: avg=2018.59ms min=23.37ms max=4013.81ms count=2
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=337.31ms min=7.95ms max=3244.92ms count=10
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=148079.69ms min=148079.69ms max=148079.69ms count=1
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 D EGL_emulation: app_time_stats: avg=455.73ms min=16.61ms max=1161.21ms count=3
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 W Parcel : Expecting binder but got null!
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=748.34ms min=38.31ms max=1458.37ms count=2
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=4683.57ms min=4683.57ms max=4683.57ms count=1
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 W Parcel : Expecting binder but got null!
2447 2469 D EGL_emulation: app_time_stats: avg=459.87ms min=29.08ms max=1174.85ms count=3
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=19445.36ms min=32.50ms max=38858.21ms count=2
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=42827.51ms min=9.34ms max=428142.03ms count=10
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 D EGL_emulation: app_time_stats: avg=430.53ms min=24.02ms max=1042.93ms count=3
2447 2469 E OpenGLRenderer: Unable to match the desired swap behavior.
2447 2469 W Parcel : Expecting binder but got null!
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=592.69ms min=38.30ms max=1147.08ms count=2
2447 2469 D EGL_emulation: app_time_stats: avg=657.28ms min=10.25ms max=6435.94ms count=10
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=2188.93ms min=11.70ms max=21740.09ms count=10
2447 2447 D ** SYS_CTRL **: EVABS{logging_info_never_safe}
2447 2469 D EGL_emulation: app_time_stats: avg=1164.92ms min=1164.92ms max=1164.92ms count=1

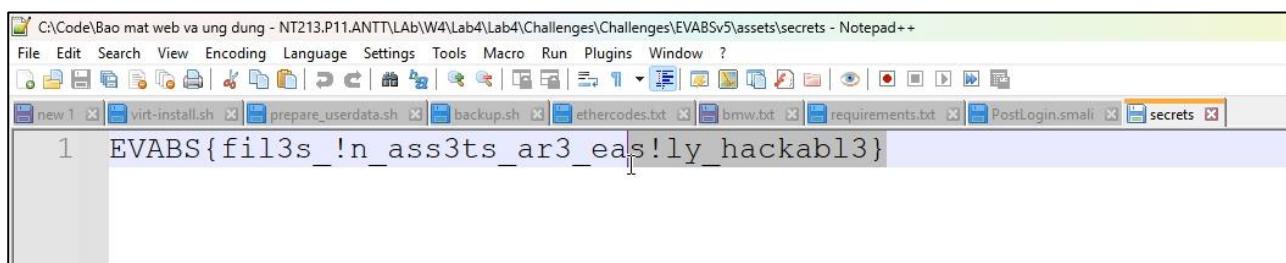
```

Level 2

- Video thực hiện: <https://youtu.be/ssKkd2n-j8I>
- Flag: EVABS{fil3s_!n_ass3ts_ar3_eas!ly_hackabl3}
- Dựa trên gợi ý của Level, ta sẽ sử dụng file apk để tìm folder asset, tức ta sẽ sử dụng apktool để decompile file apk gốc của Challenge

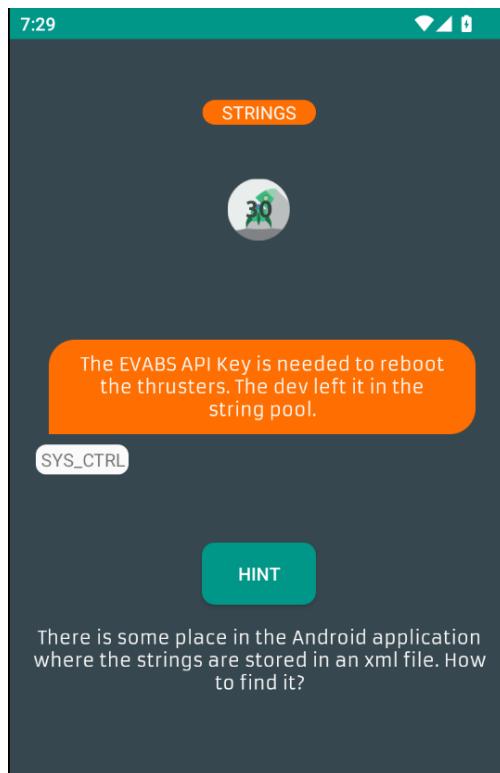


- Truy cập vào folder được compile ra, ta sẽ thấy thư mục asset bên trong và cả 1 tập tin secrets bên trong nó chính là flag của level này



Level 3

- Video thực hiện: <https://youtu.be/87X1YZWS6IA>
- Dựa vào hint của challenge (Strings), flag cần tìm là 1 API KEY và được lập trình viên lưu như một chuỗi kí tự, và biết rằng các chuỗi sử dụng trong apk lưu tại file string.xml.



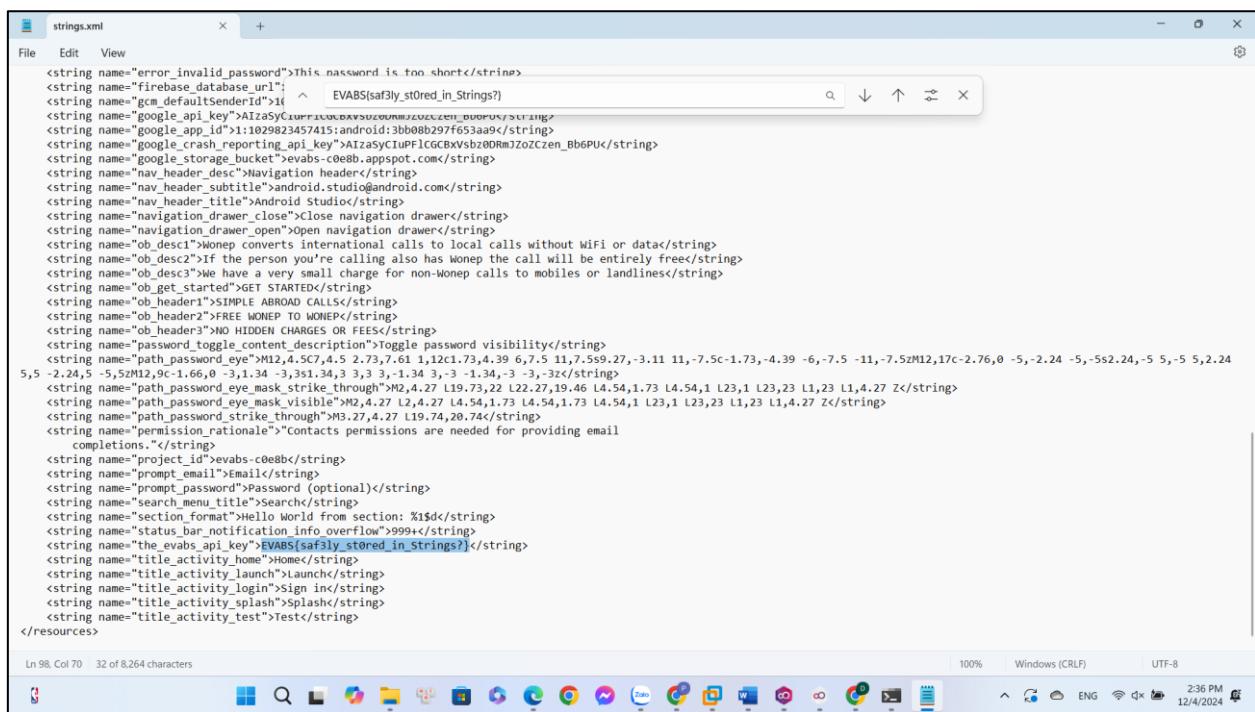
- Tiến hành dịch ngược file apk EVABSV5.apk.

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\nguye>apktool d D:\BMWapp\platform-tools\EVABSV5.apk -o D:\BMWapp\platform-tools\EVABSV5out
I: Using Apktool 2.10.0 on EVABSV5.apk with 16 thread(s).
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\nguye\AppData\Local\apktool\framework\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...

C:\Users\nguye>
```

- Truy cập vào đường dẫn EVABSV5out\res\values\strings.xml và tìm kiếm chuỗi EVABS trong tập tin strings.xml cho đến khi tìm được flag.



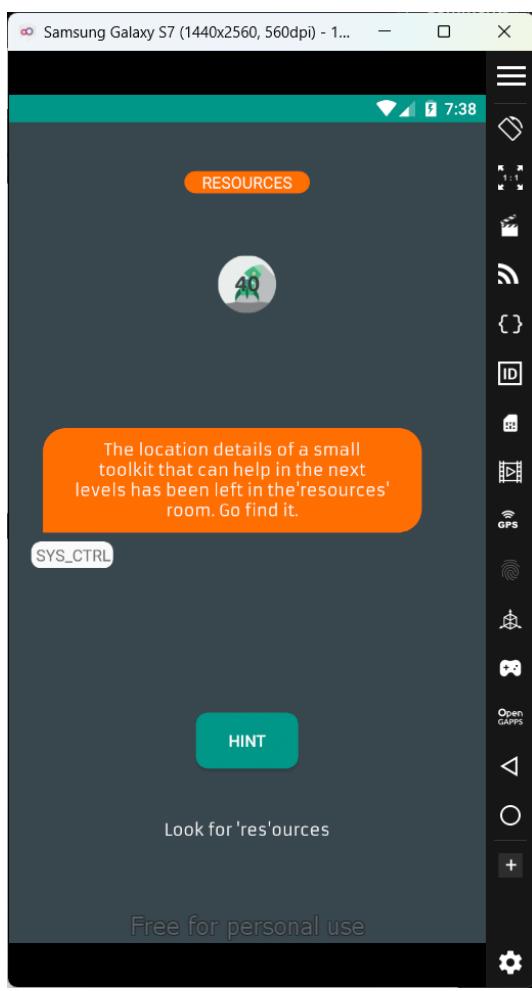
```

strings.xml
File Edit View
<string name="error_invalid_password">This password is too short</string>
<string name="firebase_database_url">^ EVABS(saf3ly_st0red_in_Strings)?</string>
<string name="gcm_defaultSenderId">id ^ EVABS(saf3ly_st0red_in_Strings)?</string>
<string name="google_api_key">AIzaSyC1upr1c0oxwszeumwzozCzen_Bb6PUc</string>
<string name="google_app_id">x1:1029823457415:android:3bb08b297f63aa9</string>
<string name="google_crash_reporting_api_key">AIzaSyC1upF1GGBxvzb0DRmJz0ZCzen_Bb6PUc</string>
<string name="google_storage_bucket">evabs-c0e8b.appspot.com</string>
<string name="nav_header_desc">Navigation header</string>
<string name="nav_header_subtitle">android.studio@android.com</string>
<string name="nav_header_title">Android Studio</string>
<string name="navigation_drawer_close">Close navigation drawer</string>
<string name="navigation_drawer_open">Open navigation drawer</string>
<string name="ob_desc1">Wonep converts international calls to local calls without WiFi or data</string>
<string name="ob_desc2">If the person you're calling also has wonep the call will be entirely free</string>
<string name="ob_desc3">We have a very small charge for non-Wonep calls to mobiles or landlines</string>
<string name="ob_get_started">GET STARTED</string>
<string name="ob_header1">SIMPLE ABROAD CALLS</string>
<string name="ob_header2">FREE WONEP TO WONEP</string>
<string name="ob_header3">NO HIDDEN CHARGES OR FEES</string>
<string name="password_toggle_content_description">Toggle password visibility</string>
<string name="path_password">M12,4.5C7,4.5 2.73,7.61 1,12c1.73,4.39 6,7.5 11,7.599,27,-3.11 11,-7.5c-1.73,-4.39 -6,-7.5 -11,-7.5zM12,17c-2.76,0 -5,-2.24 -5,-5s2.24,-5 5,-5 5,2.24
5, -2,24,5 -5,s2M12,9C-1.66,0 -3,1.34 -3,3s1.34,3 3,3 -1.34,3,-3 -1.34,-3 -3,-3z</string>
<string name="path_password_eye_mask_strike_through">M2,4.27 L19,73,22 L22,27,19,46 L4,54,1,73 L4,54,1 L23,1 L23,23 L1,23 L1,4,27 Z</string>
<string name="path_password_eye_mask_visible">M2,4.27 L2,4,27 L4,54,1,73 L4,54,1 L23,1 L23,23 L1,23 L1,4,27 Z</string>
<string name="path_password_strike_through">M3,27,4,27 L19,74,20,74</string>
<string name="permission_rationale">"Contacts permissions are needed for providing email
completions."</string>
<string name="project_id">evabs-c0e8b</string>
<string name="prompt_email">Email</string>
<string name="prompt_password">Password (optional)</string>
<string name="search_menu_title">Search</string>
<string name="section_format">Hello World from section: %1$sd</string>
<string name="status_bar_notification_info_overflow">999</string>
<string name="the_evabs_api_key">EVABS(safely_stored_in_Strings)?</string>
<string name="title_activity_home">Home</string>
<string name="title_activity_launch">Launch</string>
<string name="title_activity_login">Sign In</string>
<string name="title_activity_splash">Splash</string>
<string name="title_activity_test">Test</string>
</resources>
Ln 98 Col 70 32 of 8,264 characters
100% Windows (CRLF) UTF-8
2:36 PM 12/4/2024

```

Level 4

- Video thực hiện: <https://youtu.be/NEVyXqjMQ5g>

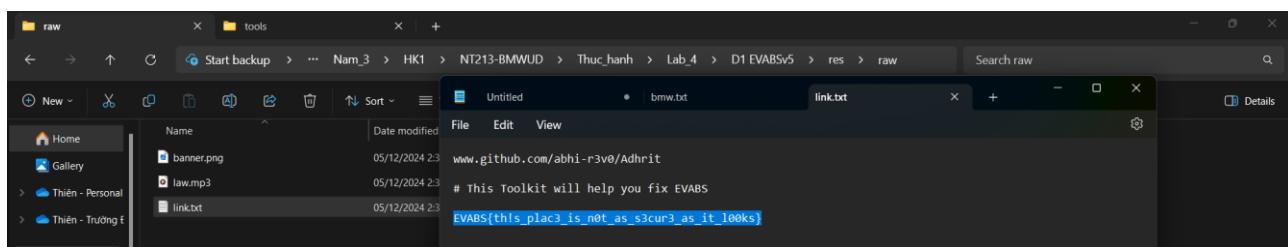


- Ta sử dụng apktool để giải mã file APK thành mã nguồn có thể đọc được (XML, Smali code)

```
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>apktool.bat "C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4\Resources\D1 EVABSV5.apk"
I: Using Apktool 2.10.0 on D1 EVABSV5.apk with 12 thread(s).
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\WanThinnn\AppData\Local\apktool\framework\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

- Thông qua quá trình tìm kiếm trong /res/raw, ta thu được flag:

EVABS{th!s_plac3_is_n0t_as_s3cur3_as_it_100ks}



- Tuy nhiên, cách tìm như vậy sẽ khá lâu, dưới đây là đoạn code python để tìm flag dựa theo gợi ý của bài:

```
import os
import re

def find_flag_in_res(directory):
    # Biểu thức chính quy để tìm flag dạng EVABS{...}
    flag_pattern = re.compile(r'EVABS\{.*?\}')
    flags_found = []

    # Duyệt qua tất cả các file trong thư mục "res" và các thư mục con
    for root, dirs, files in os.walk(directory):
        for file in files:
            file_path = os.path.join(root, file)
            try:
                # Mở và đọc nội dung file
                with open(file_path, 'r', encoding='utf-8') as f:
                    content = f.read()
                    # Tìm tất cả flag khớp với biểu thức chính quy
                    flags = flag_pattern.findall(content)
                    if flags:
                        flags_found.extend(flags)
            except (UnicodeDecodeError, FileNotFoundError):
                # Bỏ qua các file không thể đọc
                pass

    if flags_found:
        print("Flags found:")
        for flag in flags_found:
            print(flag)
    else:
        print("No flags found in 'res' directory.")

# Đường dẫn đến thư mục "res"
res_directory = r"C:\\\\Users\\\\WanThinnn\\\\Documents\\\\UIT\\\\Nam_3\\\\HK1\\\\NT213-BMWUD\\\\Thuc_hanh\\\\Lab_4\\\\D1_EVABSV5\\\\res"
find_flag_in_res(res_directory)
```

- Kết quả thu được thêm 1 flag nữa:

```
C:\\\\Users\\\\WanThinnn\\\\Documents\\\\UIT\\\\Nam_3\\\\HK1\\\\NT213-BMWUD\\\\Thuc_hanh\\\\Lab_4\\\\Resources\\\\AndroLabServer\\\\AndroLabServer\\\\\"C:/Program Files/Python/Python310/python.exe\" c:\\\\Users\\\\WanThinnn\\\\Documents\\\\UIT\\\\Nam_3\\\\HK1\\\\NT213-BMWUD\\\\Thuc_hanh\\\\Lab_4\\\\challenge-1-level-5.py
Flags found:
EVABS{}
EVABS{some_t3xt_here}
EVABS{}
EVABS{some_t3xt_here}
EVABS{this_plac3_is_n0t_as_s3cur3_as_it_l00ks}
EVABS{saft3ly_st0red_in_strings?}
```

Level 5

- Video thực hiện: <https://youtu.be/2U9aOLuh5uU>
- Sử dụng adb shell để truy cập vào hệ thống android bằng lệnh:

```
cd /data/data/com.revo.evabs/shared_prefs
```

- Chạy lệnh grep -r "EVABS" * để tìm flag giấu trong các file xml.

```
D:\Android\Challenges>ls
EVABSV5  EVABSV5.apk  five.apk  four.apk  mtest  one.apk  three.apk  two  two.apk

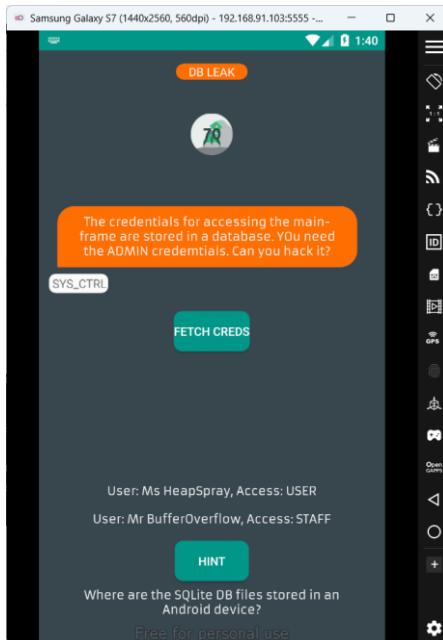
D:\Android\Challenges>adb shell
genymotion:/ # cd /data/data/com.revo.evabs/shared_prefs
genymotion:/data/data/com.revo.evabs/shared_prefs # grep -r "EVABS" *
DETAILS.xml:    <string name="password">EVABS{shar3d_pr3fs_c0uld_be_c0mpromiz3ds}</string>
genymotion:/data/data/com.revo.evabs/shared_prefs # |
```

- Ta tìm được flag: EVABS{shar3d_pr3fs_c0uld_be_c0mpromiz3ds}

Level 6

- Video thực hiện: <https://youtu.be/3wnHEMSKvHk>

- Dựa vào gợi ý, ta biết rằng các ứng dụng Android có lưu trữ dữ liệu local. Một trong những nơi ứng dụng sử dụng để lưu trữ dữ liệu là SQLite DB. Các database này luôn nằm tại /data/data/<package-name>/databases



- Sau khi click button "FETCH CREDS" thì sẽ có DB được sinh ra trong Local Storage.

- Tiếp theo, ta chạy lệnh: adb -s 192.168.91.103 shell "ls /data/data/com.revo.evabs/databases" để kiểm tra xem có những db nào. Ở đây có 2 databases là MAINFRAME_ACCESS và MAINFRAME_ACCESS-journal.

```
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>adb -s 192.168.91.103 shell "ls /data/data/com.revo.evabs/databases"
MAINFRAME_ACCESS
MAINFRAME_ACCESS-journal
```

- Ta pull DB đó về máy thật để mở bằng SQLite browser bằng lệnh adb -s 192.168.91.103 pull "/data/data/com.revo.evabs/databases/MAINFRAME_ACCESS".

```
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>adb -s 192.168.91.103 pull "/data/data/com.revo.evabs/databases/MAINFRAME_ACCESS"
/data/data/com.revo.evabs/databases/MAINFRAME_ACCESS: 1 file pulled. 1.2 MB/s (16384 bytes in 0.013s)
```

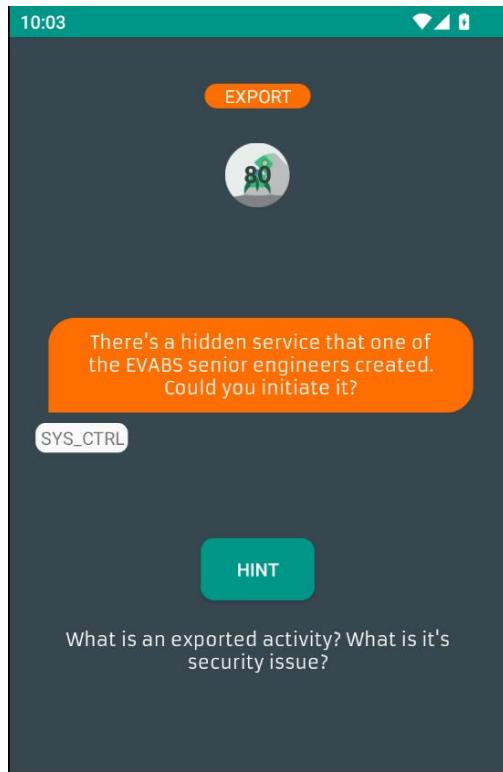
- Xem các bảng, ta tìm được flag là password của user Dr.l33t có role admin.

Table: CREDS		
admin	pass	access
1 Dr.l33t	EVABS(sqlite_is_not_safe)E	ADMIN
2 Mr_BufferOverflow	0xNotSecureSQLite_	STAFF
3 Ms_HeapSpray	SQLite_exploit	USER

- Với database MAINFRAME_ACCESS-journal thì dung lượng sau khi pull về 0kb, ta không tìm được gì ở database này.

Level 7

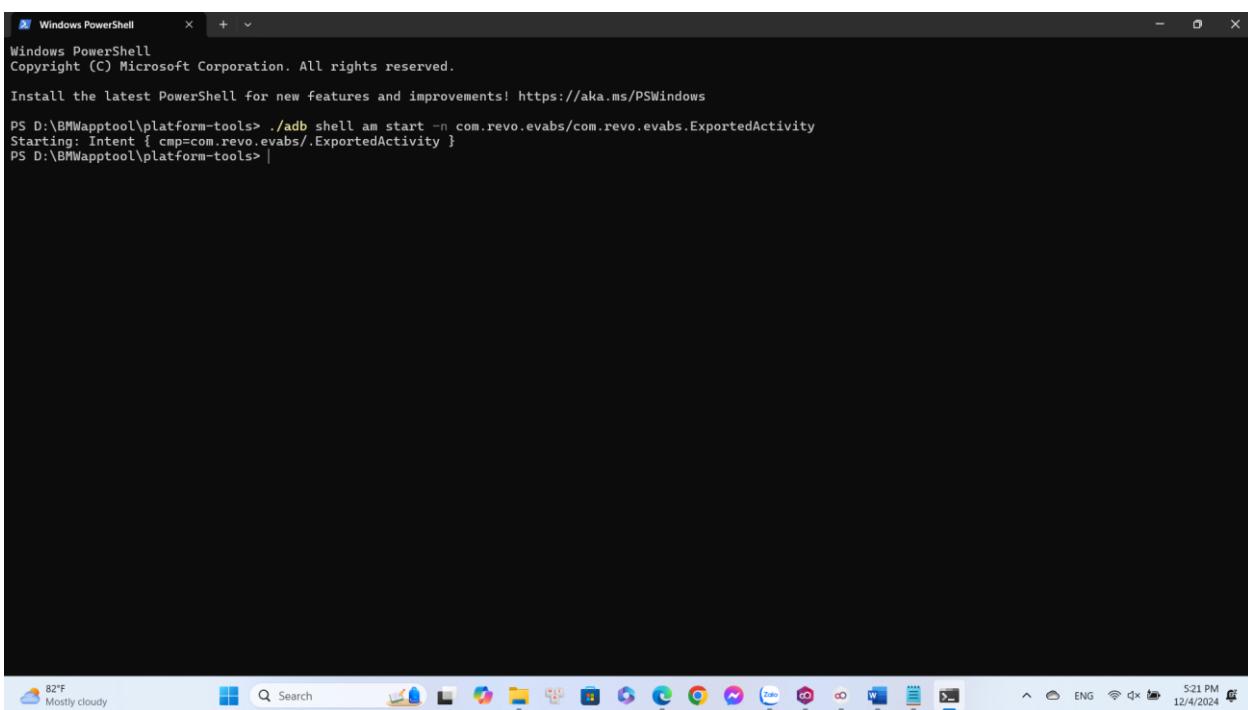
- Video thực hiện: <https://youtu.be/058EK7E9Ha4>
 - Dựa vào hint challenge (Export), ta cần tìm exported activity.



- Exported activity của ứng dụng android là hàm có thể được sử dụng bởi các ứng dụng Android khác. Nó được khai báo trong file AndroidManifest.xml với trường android:exported="true"

```
<?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.revo.evabs">
<uses-permission android:name="android.permission.INTERNET"/>
<application android:allowBackup="false" android:debuggable="true" android:icon="@mipmap/ic_launcher" android:label="@string/app_name" android:roundIcon="@mipmap/ic_launcher_round"
    android:supportRtl="true" android:theme="@style/AppTheme">
    <activity android:exported="true" android:name=".com.revo.evabs.ExportedActivity"/>
    <activity android:name=".com.revo.evabs.Frida"/>
    <activity android:name=".com.revo.evabs.FileRead"/>
    <activity android:name=".com.revo.evabs.DebugMe"/>
    <activity android:name=".com.revo.evabs.Welcome"/>
    <activity android:name=".com.revo.evabs.ChallengeList" android:parentActivityName=".com.revo.evabs.Launch">
        <meta-data android:name="android.support.PARENT_ACTIVITY" android:value=".Launch"/>
    </activity>
    <activity android:name=".com.revo.evabs.ExportedInfo"/>
    <activity android:name=".com.revo.evabs.SmallInject"/>
    <activity android:name=".com.revo.evabs.StringsSecrets"/>
    <activity android:name=".com.revo.evabs.SharedBreach"/>
    <activity android:name=".com.revo.evabs.Decode"/>
    <activity android:name=".com.revo.evabs.BadComm"/>
    <activity android:name=".com.revo.evabs.DBLreak"/>
    <activity android:name=".com.revo.evabs.CustomAccess">
        <intent-filter>
            <action android:name=".com.revo.evabs.actionSENSOR_KEY"/>
            <category android:name="android.intent.category.DEFAULT"/>
            <data android:mimeType="text/plain"/>
        </intent-filter>
    </activity>
    <activity android:name=".com.revo.evabs.Res_raw"/>
    <meta-data android:name="preloaded_fonts" android:resource="@array/preloaded_fonts"/>
    <activity android:name=".com.revo.evabs.Welcome0"/>
    <activity android:name=".com.revo.evabs.Splash">
        <intent-filter>
            <action android:name="android.intent.action.MAIN"/>
            <category android:name="android.intent.category.LAUNCHER"/>
        </intent-filter>
    </activity>
    <activity android:name=".com.revo.evabs.Login"/>
    <activity android:name=".com.revo.evabs.Checker"/>
</application>
</manifest>
```

- Ta thực thi lệnh activity trong shell sử dụng lệnh ./adb shell am start -n com.revo.evabs/com.revo.evabs.ExportedActivity



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\BWWapptool\platform-tools> ./adb shell am start -n com.revo.evabs/com.revo.evabs.ExportedActivity
Starting: Intent { cmp=com.revo.evabs/.ExportedActivity }
PS D:\BWWapptool\platform-tools> |
```

- Sau đó, giao diện điện thoại ảo xuất hiện với flag cần tìm.



Level 8

- Video thực hiện: https://youtu.be/_SZAce4VzZk
- Reverse sang code java bằng Bytecode viewer, sau đó mở file Decode.class - file code java cho level 8 thấy ngay 3 đoạn text hardcoded.

```

1 package com.revo.evabs;
2
3 import android.os.Bundle;
4 import android.support.v7.app.AppCompatActivity;
5 import android.widget.Button;
6 import android.widget.TextView;
7
8 public class Decode extends AppCompatActivity {
9     protected void onCreate(Bundle var1) {
10         super.onCreate(var1);
11         this.setContentView(2131492896);
12         StringVar2 = new StringBuilder();
13         var2.append("RVZBQN7bmV2K9fF5Quewu");
14         var2.append("X93aenWd01MjPw | 0y |");
15         var2.append("XxFuXddI1szMMy72vjhR1");
16         var2.toString();
17         ((Button)this.findViewById(2131361842)).setOnClickListener(new 1(this, (TextView)this.findViewById(2131362094)));
18     }
19 }

```

- Decode B64 với 3 string này ta có được flag:

EVACS{nev3r_st0re_s3ns!tmv3_data_1n_7h3_s0urce0du}

Decode from Base64 format

Simply enter your data then push the decode button.

```
RVZBQ1N7bmV2M3Jfc3QwcmU=
```

- ⓘ For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

Source character set.

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

```
EVACS{nev3r_st0re
```

Decode from Base64 format

Simply enter your data then push the decode button.

```
X3MzbnMhdG12M19kYXRh
```

- i** For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

```
_s3ns!tmv3_data
```

Decode from Base64 format

Simply enter your data then push the decode button.

```
XzFuXzdoM19zMHVyY2VjMGR1
```

- ⓘ For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

```
_1n_7h3_s0urcece0du
```

Level 9

- Video thực hiện: <https://youtu.be/cKv8Uf9dOg8>
- Flag: EVABS{smali_inj_is_l3thals}
- Sử dụng apktool để decompile file apk gốc, tiến hành thực hành tìm kiếm mã nguồn của level này ta sẽ tìm thấy nó ở đường dẫn "~\EVABSV5\smali\com\revo\evabs\SmaliInject.smali"
- Xem mã nguồn ta sẽ thấy nó có điều kiện if else, cụ thể là 1 trường hợp sẽ thực hiện file SmaliInject\$1.smali, trường hợp còn lại sẽ thực hiện SmaliInject\$2.smali

```

94     invoke-virtual {p0, v4}, Lcom/revo/evabs/SmaliInject;->findViewById(I)Landroid/view/View;
95
96     move-result-object v4
97
98     check-cast v4, Landroid/widget/TextView;
99
100    .line 24
101    .local v4, "tvlaboff":Landroid/widget/TextView;
102    const v5, 0x7f0a127
103
104    invoke-virtual {p0, v5}, Lcom/revo/evabs/SmaliInject;->findViewById(I)Landroid/view/View;
105
106    move-result-object v5
107
108    check-cast v5, Landroid/widget/TextView;
109
110    .line 26
111    .local v5, "tvflag":Landroid/widget/TextView;
112    new-instance v6, Lcom/revo/evabs/SmaliInject$1;
113
114    invoke-direct {v6, p0, v3}, Lcom/revo/evabs/SmaliInject$1;-><init>(Lcom/revo/evabs/SmaliInject;Landroid/widget/TextView;)V
115
116    invoke-virtual {v1, v6}, Landroid/widget/Button;->setOnClickListener(Landroid/view/View$OnClickListener;)V
117
118    .line 34
119    new-instance v6, Lcom/revo/evabs/SmaliInject$2;
120    |
121    invoke-direct {v6, p0, v4, v2, v5}, Lcom/revo/evabs/SmaliInject$2;-><init>(Lcom/revo/evabs/SmaliInject;Landroid/widget/TextView;Landroid/widget/TextView;Landroid/widget/TextView;)V
122
123    invoke-virtual {v0, v6}, Landroid/widget/Button;->setOnClickListener(Landroid/view/View$OnClickListener;)V
124
125    .line 52
126    return-void
127
128    .end method
129
130    .method public native stringFromSmali()Ljava/lang/String;
131    .end method

```

- Lúc này ta sẽ thực hiện điều chỉnh đoạn code cho nó thực hiện SmaliInject\$2.smali ở bất kì điều kiện nào

```

94     invoke-virtual {p0, v4}, Lcom/revo/evabs/SmaliInject;->findViewById(I)Landroid/view/View;
95
96     move-result-object v4
97
98     check-cast v4, Landroid/widget/TextView;
99
100    .line 24
101    .local v4, "tvlaboff":Landroid/widget/TextView;
102    const v5, 0x7f0a127
103
104    invoke-virtual {p0, v5}, Lcom/revo/evabs/SmaliInject;->findViewById(I)Landroid/view/View;
105
106    move-result-object v5
107
108    check-cast v5, Landroid/widget/TextView;
109
110    .line 26
111    .local v5, "tvflag":Landroid/widget/TextView;
112    new-instance v6, Lcom/revo/evabs/SmaliInject$2;
113
114    invoke-direct {v6, p0, v4, v2, v5}, Lcom/revo/evabs/SmaliInject$2;-><init>(Lcom/revo/evabs/SmaliInject;Landroid/widget/TextView;Landroid/widget/TextView;Landroid/widget/TextView;)V
115
116    invoke-virtual {v0, v6}, Landroid/widget/Button;->setOnClickListener(Landroid/view/View$OnClickListener;)V
117
118    .line 34
119    new-instance v6, Lcom/revo/evabs/SmaliInject$2;
120
121    invoke-direct {v6, p0, v4, v2, v5}, Lcom/revo/evabs/SmaliInject$2;-><init>(Lcom/revo/evabs/SmaliInject;Landroid/widget/TextView;Landroid/widget/TextView;Landroid/widget/TextView;)V
122
123    invoke-virtual {v0, v6}, Landroid/widget/Button;->setOnClickListener(Landroid/view/View$OnClickListener;)V
124
125    .line 52
126    return-void
127
128    .end method
129
130    .method public native stringFromSmali()Ljava/lang/String;
131    .end method

```

- Thực hiện vào xem mã nguồn của SmaliInject\$2.smali, có 2 điểm đáng chú ý sau (các phần được tô đen)

```

SmaliInject.smali SmaliInject$2.smali
13     invoke-direct {p0}, Ljava/lang/Object;-><init>()V
14
15     return-void
16 .end method
17
18
19 # virtual methods
20 .method public onClick(Landroid/view/View;)V
21     .locals 4
22     .param p1, "view"    # Landroid/view/View;
23
24     .line 38
25     igure-object v0, p0, Lcom/revo/evabs/SmaliInject$2;->this$0:Lcom/revo/evabs/SmaliInject;
26
27     invoke-virtual {v0}, Lcom/revo/evabs/SmaliInject;->stringFromSmali()Ljava/lang/String;
28
29     move-result-object v0
30
31
32     .line 40
33     .local v0, "ctrl":Ljava/lang/String;
34     igure-object v1, p0, Lcom/revo/evabs/SmaliInject$2;->this$0:Lcom/revo/evabs/SmaliInject;
35
36     igure-object v1, v1, Lcom/revo/evabs/SmaliInject;->SIGNAL:Ljava/lang/String;
37
38     const-string v2, "LAB_ON"
39
40     invoke-virtual {v1, v2}, Ljava/lang/String;->equals(Ljava/lang/Object;)Z
41
42     move-result v1
43
44     if-eqz v1, :cond_0
45
46

```

```

SmaliInject.smali SmaliInject$2.smali
91     igure-object v1, p0, Lcom/revo/evabs/SmaliInject$2;->val$tvflag:Landroid/widget/TextView;
92
93     new-instance v2, Ljava/lang/StringBuilder;
94
95     invoke-direct {v2}, Ljava/lang/StringBuilder;-><init>()V
96
97     const-string v3, "EVABS{"
98
99     invoke-virtual {v2, v3}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
100
101    invoke-virtual {v2, v0}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
102
103    const-string v3, "}"
104
105    invoke-virtual {v2, v3}, Ljava/lang/StringBuilder;->append(Ljava/lang/String;)Ljava/lang/StringBuilder;
106
107    invoke-virtual {v2}, Ljava/lang/StringBuilder;->toString()Ljava/lang/String;
108
109    move-result-object v2
110
111    invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
112
113    goto :goto_0
114
115     .line 46
116     :cond_0
117     igure-object v1, p0, Lcom/revo/evabs/SmaliInject$2;->val$tvlaboff:Landroid/widget/TextView;
118
119     const-string v2, "SYS_CTRL_CODE: LAB_OFF"
120
121     invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
122
123     .line 47
124     igure-object v1, p0, Lcom/revo/evabs/SmaliInject$2;->val$labstat:Landroid/widget/TextView;
125
126     const-string v2, "SYS_CTRL: ACCESS_DENIED"
127
128     invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V
129
130     .line 49
131     :goto_0
132     return-void
133 .end method
134

```

=> Phần này sử dụng SIGNAL được kiểm tra để quyết định hiển thị trạng thái, tức Nếu SIGNAL == "LAB_ON", hiển thị trạng thái ACCESS_GRANTED, Nếu SIGNAL != "LAB_ON", hiển thị trạng thái ACCESS_DENIED

- Thực hiện xóa 2 phần được tô đen và 1 số điều kiện liên quan, khi đó trạng thái ACCESS_GRANTED được áp dụng mặc định mà không có kiểm tra signal
- Thực hiện lưu lại các file sau khi chỉnh sửa, tạo key mới và kí lại rồi build lại file apk mới

```
C:\Windows\System32\cmd.exe
C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\apktool d EVABSV5.apk
I: Using Apktool 2.10.0 on EVABSV5.apk with 16 thread(s).
I: Baksmaling classes...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\namphuong\AppData\Local\apktool\framework\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Resources XMLs...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\key.keystore>keytool -genkeypair -v -keystore "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\key.keystore" -alias CTF9 -keyalg RSA -keysize 2048 -validity 1000
Enter keystore password:
Re-enter new password:
Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press ENTER to use the default value in braces.
What is your first and last name?
What is the name of your organizational unit?
What is the name of your organization?
What is the name of your City or Locality?
What is the name of your State or Province?
What is the two-letter country code for this unit?
Is CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, S=Unknown, C=Unknown correct?
[no]: yes
Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 1,000 days
[Storing C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\key.keystore]

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\apktool b EVABSV5 -o ctf9.apk
I: Using Apktool 2.10.0 with 16 thread(s).
I: Checking for updates...
I: Sealing small folder into classes.dex...
I: Checking whether resources has changed...
I: Building resources...
I: Copying libs into /lib/
I: Building signed file...
I: Copying unknown files</dir>...
I: Built apk into: ctf9.apk

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\jarsigner -keystore "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\key.keystore" -storepass namphuong "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\ctf9.apk" CTF9
jar signed.

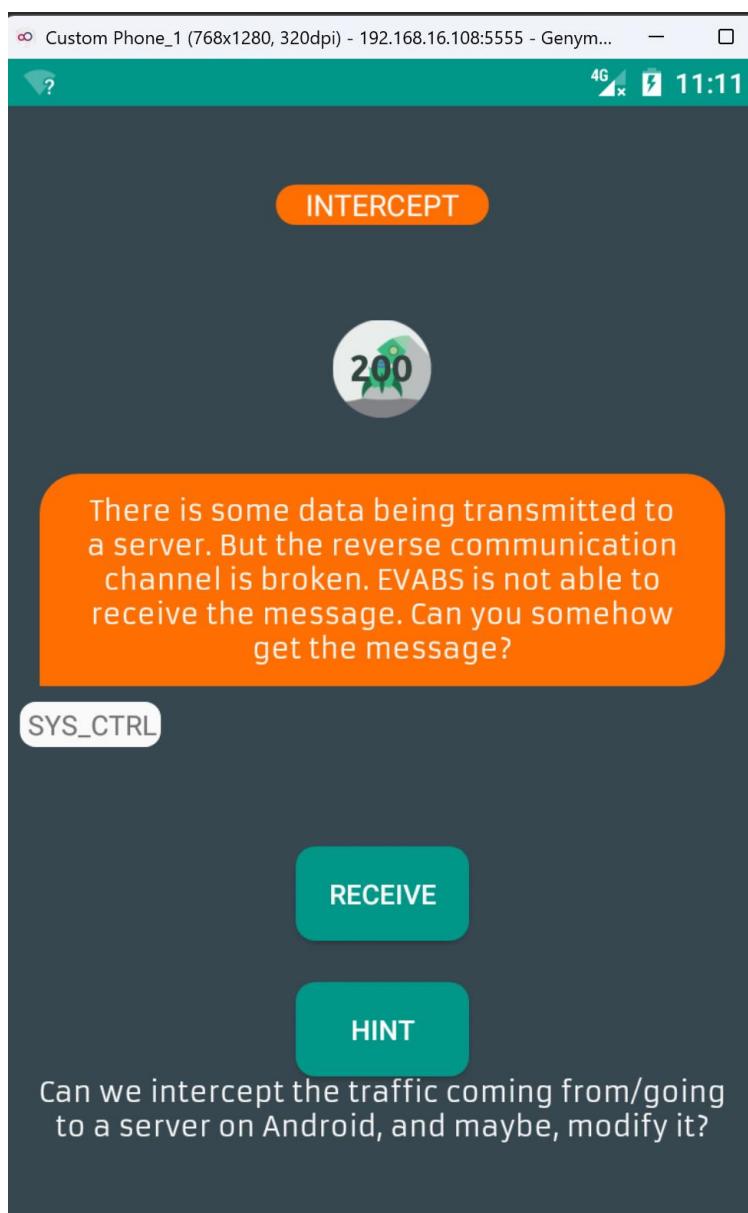
Warning:
The signer's certificate is self-signed.

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges>adb install ctf9.apk
Performing Streamed Install
```

- Kiểm tra kết quả trong file APK mới:



Level 10



- Video thực hiện: <https://youtu.be/ug0kTbCFmoo>
- Ta sẽ sử dụng OWASP ZAP để bắt gói tin
- Ta thực hiện các bước tạo cer, và kết nối OWASP với thiết bị Android
- Sau đó ta thực hiện bắt gói tin

104	Proxy	4/04/21 12:50:22 AM	GET	https://www.google.com/xjs/_/js/k=xjs.qs.es_4...	200	OK	249milisegun...	144.519bytes
111	Proxy	4/04/21 12:50:23 AM	GET	https://www.google.com/xjs/_/js/k=xjs.qs.es_4...	200	OK	124milisegun...	12.641bytes
119	Proxy	4/04/21 12:50:23 AM	POST	https://www.google.com/gen_204?atyp=i&ei=...	204	No Content	170milisegun...	0bytes
120	Proxy	4/04/21 12:50:23 AM	POST	https://www.google.com/gen_204?atyp=csi&ei=...	204	No Content	112milisegun...	0bytes
121	Proxy	4/04/21 12:50:23 AM	GET	https://www.google.com/client_204?cs=0	204	No Content	198milisegun...	0bytes
123	Proxy	4/04/21 12:50:23 AM	GET	https://www.google.com/async/bgasy?ei=mVN...	200	OK	230milisegun...	5.517bytes
131	Proxy	4/04/21 12:50:23 AM	GET	https://www.google.com/xjs/_/js/k=xjs.qs.es_4...	200	OK	125milisegun...	12.737bytes
146	Proxy	4/04/21 12:50:23 AM	GET	https://www.google.com/xjs/_/js/k=xjs.qs.es_4...	200	OK	121milisegun...	7.538bytes
153	Proxy	4/04/21 12:50:25 AM	POST	https://www.google.com/gen_204?atyp=csi&ei=...	204	No Content	107milisegun...	0bytes
154	Proxy	4/04/21 12:50:25 AM	GET	https://adservice.google.com/adsid/google/ui	204	No Content	156milisegun...	0bytes
155	Proxy	4/04/21 12:50:25 AM	POST	https://www.google.com/gen_204?atyp=i&ei=...	204	No Content	98milisegund...	0bytes
156	Proxy	4/04/21 12:50:43 AM	POST	https://evabstfao.000webhostapp.com/reboot.p...	200	OK	501milisegun...	33bytes

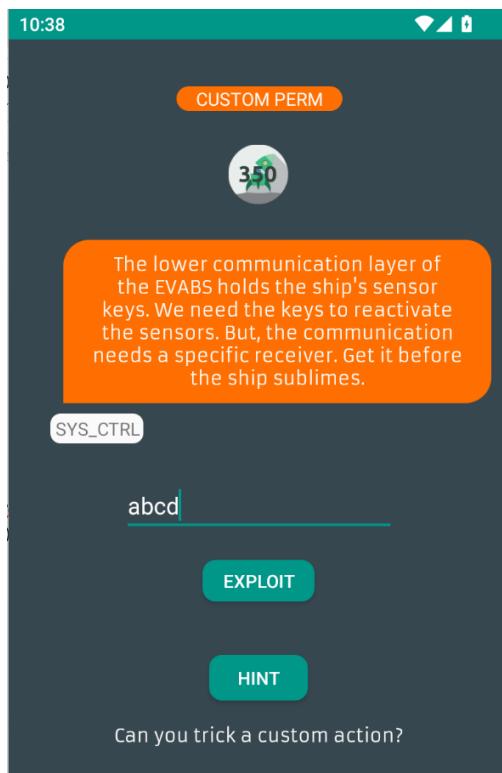
```
HTTP/1.1 200 OK
Date: Sun, 04 Apr 2021 05:50:41 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
Server: awex
X-Xss-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Request-ID: 8b92b47716a7ee6e2c9f3b8fdb1bdab6

EVABS{Always_p!n_SSL_C3rtificate}
```

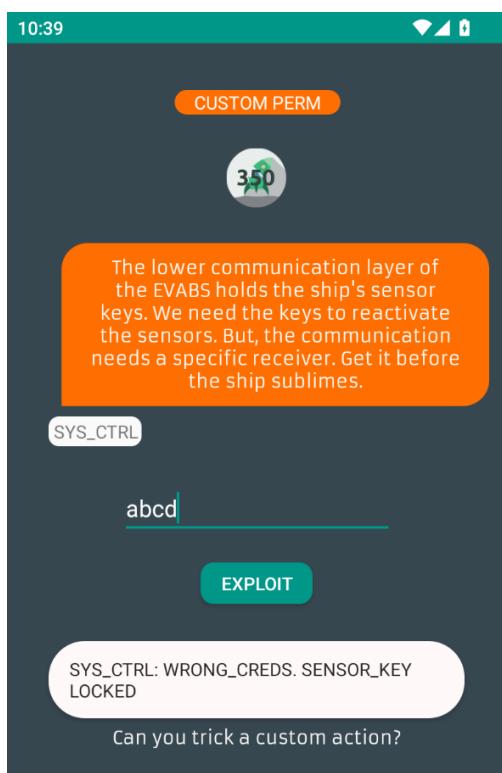
- Flag: EVABS{Always_p!n_SSL_C3rtificate}

Level 11

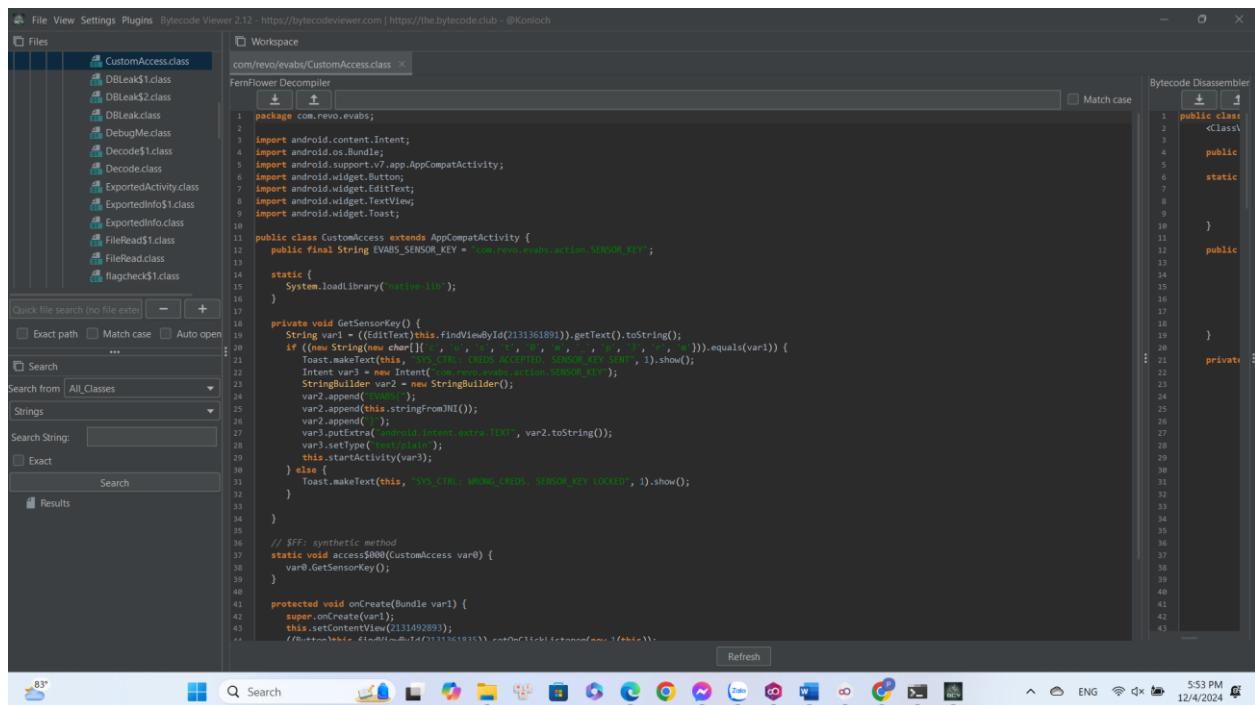
- Video thực hiện: https://youtu.be/2NiZO7_ELJI
- Nhập thử một chuỗi bất kì (ví dụ là abcd) và nhấn nút EXPLOIT.



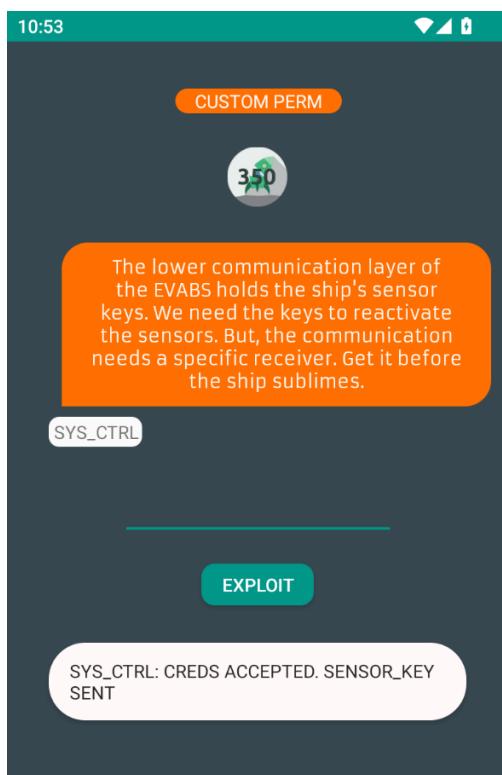
- Kết quả cho thấy đã nhập sai chuỗi.



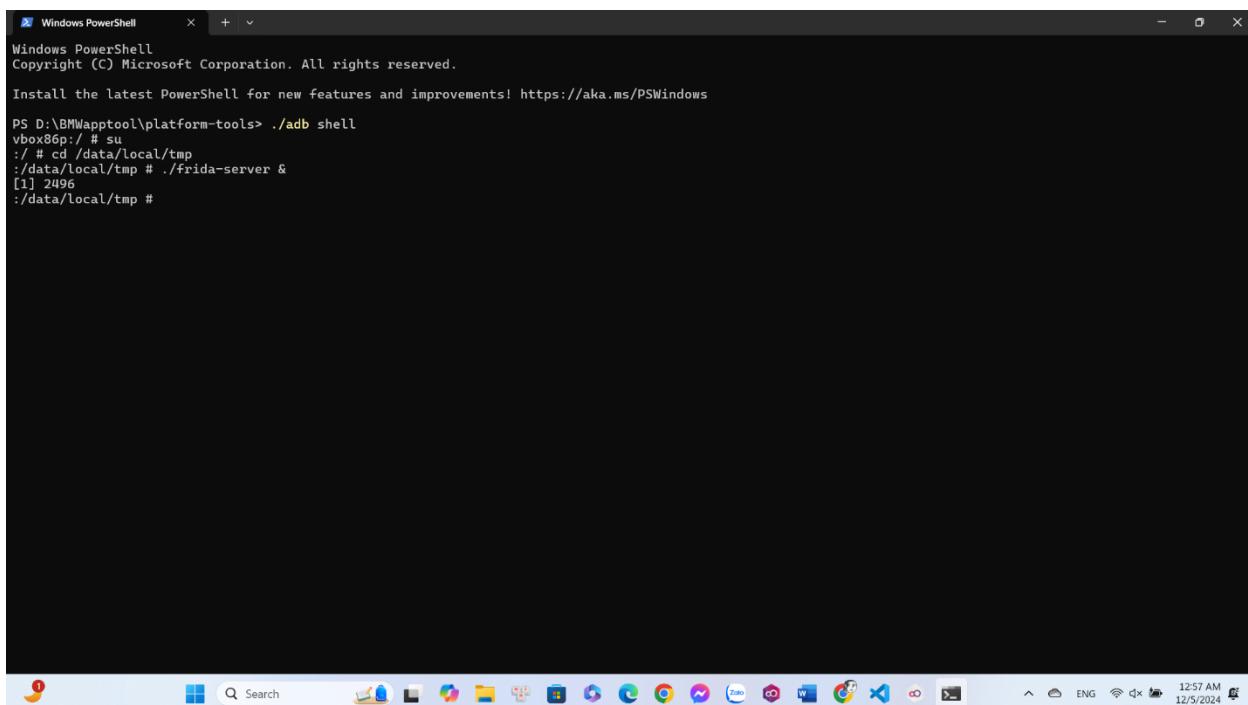
- Sử dụng ByteCode Viewer để phân tích tập tin EVABSV5.apk, mở tập tin CustomAccess.class và tiến hành xem xét.



- Ta thấy input là chuỗi “cust0m_p3rm”, nếu nhập đúng input thì flag sẽ được tạo ra và truyền vào intent com.revo.evabs.actionSENSOR KEY bằng hàm putExtra().



- Cài đặt Frida và chạy frida-Server



```

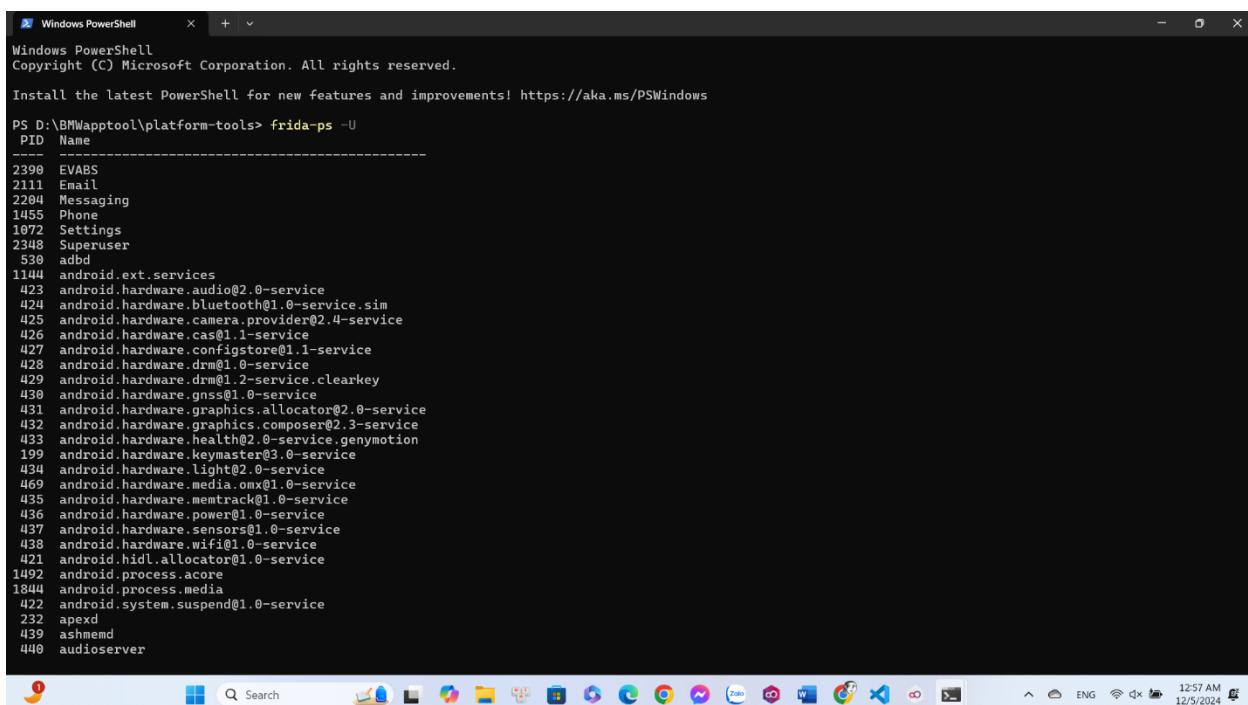
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\BMWapp\platform-tools> ./adb shell
vbox86p:/ # su
:/ # cd /data/local/tmp
:/data/local/tmp # ./frida-server &
[1] 2496
:/data/local/tmp #

```

- Kiểm tra cài đặt thành công hay chưa.



```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\BMWapp\platform-tools> frida-ps -U
PID Name
-----
2390 EVABS
2111 Email
2204 Messaging
1455 Phone
1072 Settings
2348 Superuser
538 adb
1144 android.ext.services
423 android.hardware.audio@2.0-service
424 android.hardware.bluetooth@1.0-service.sim
425 android.hardware.camera.provider@2.4-service
426 android.hardware.cas@1.1-service
427 android.hardware.configstore@1.1-service
428 android.hardware.drm@1.0-service
429 android.hardware.drm@1.2-service.clearkey
430 android.hardware.gnss@1.0-service
431 android.hardware.graphicsallocator@2.0-service
432 android.hardware.graphics.composer@2.3-service
433 android.hardware.health@2.0-service.genymotion
199 android.hardware.keymaster@3.0-service
434 android.hardware.light@2.0-service
469 android.hardware.media.omx@1.0-service
435 android.hardware.memtrack@1.0-service
436 android.hardware.power@1.0-service
427 android.hardware.sensors@1.0-service
438 android.hardware.wifi@1.0-service
421 android.hidl.allocator@1.0-service
1492 android.process.acore
1844 android.process.media
422 android.system.suspend@1.0-service
232 apexd
439 ashmemd
440 audioserver

```

- Hook hàm putExtra() bằng Frida để in ra flag cho mình với file hook.py.

```

import frida
import sys

def onMessage(message, data):
    print(message)

package = "com.revo.evabs"

```

```

jscode = """
Java.perform(function () {
    send("[-] Starting hooks android.content.Intent.putExtra");
    var intent = Java.use("android.content.Intent");
    intent.putExtra.overload("java.lang.String", "java.lang.String").implementation =
function(var_1, var_2) {
    send("[+] Flag: " + var_2);
};

});

"""
process = frida.get_usb_device().attach(package)
script = process.create_script(jscode)
script.on("message", onMessage)
print("[*] Hooking", package)
script.load()
sys.stdin.read()

```

- Sau đó, ta chạy tập tin hook.py sau khi đã chạy frida-server. Trở về app và nhập lại chuỗi cust0m_p3rm sau đó nhấn nút EXPLOIT. Thông tin xuất hiện trên terminal chứa flag cần tìm.

```

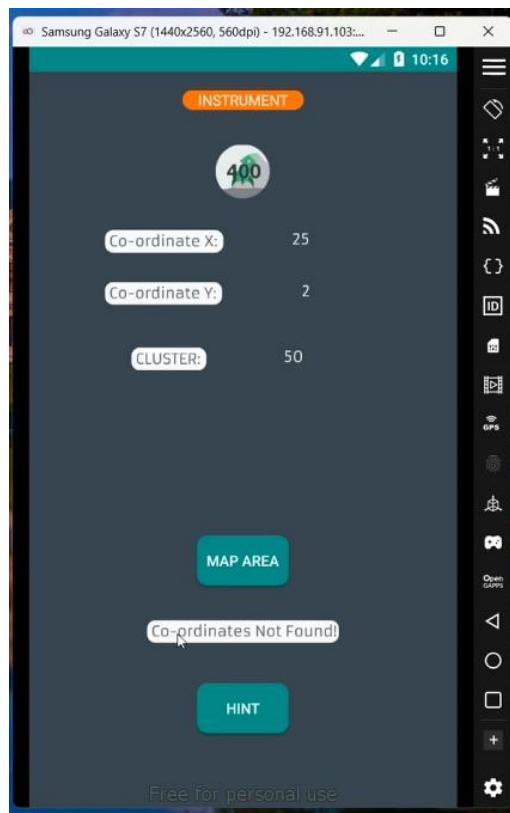
PROBLEMS OUTPUT DEBUG CONSOLE PORTS GITLENS COMMENTS TERMINAL

PS C:\Users\ADMIN\Desktop> python .\hook.py
Error: Unable to find process with name 'com.revo.evabs'
PS C:\Users\ADMIN\Desktop> python .\hook.py
+ Hooking com.revo.evabs
{'type': 'send', 'payload': '[+] Flag: EVABS{always_ver1fy_packag3sa}'}

```

Level 12

- Video thực hiện: <https://youtu.be/9-3OlkRTVcg>
- Khi click "MAP AREA" thì sẽ xuất hiện 2 tọa độ x và y cùng với 1 giá trị bằng $x * y$.



- Ta xem source code của level 12 trong file frida1.class:

```

11  public class Frida1 extends AppCompatActivity implements View.OnClickListener {
12      int a = 25;
13      int b = 2;
14      int x;
15
16      static {
17          System.loadLibrary("native-lib");
18      }
19
20      public void onClick(View var1) {
21          TextView var5 = (TextView)this.findViewById(2131361996);
22          TextView var3 = (TextView)this.findViewById(2131362132);
23          TextView var4 = (TextView)this.findViewById(2131362134);
24          TextView var6 = (TextView)this.findViewById(2131362142);
25          var3.setText(String.valueOf(this.a));
26          var4.setText(String.valueOf(this.b));
27          this.x = this.a * this.b;
28          int var2 = (new Random()).nextInt(70);
29          var6.setText(String.valueOf(this.x));
30          if (this.x > var2 + 150) {
31              var5.setText("VIBRAN IS READY TO FLY! YOU ARE GOING HOME!");
32              Log.d("CONGRATZ!", this.stringFromJNI());
33          } else {
34              var5.setText("Co-ordinates Not Found!");
35          }
36      }
37
38      protected void onCreate(Bundle var1) {
39          super.onCreate(var1);
40          this.setContentView(2131492901);
41          ((Button)this.findViewById(2131361902)).setOnClickListener(this);
42          ((Button)this.findViewById(2131361844)).setOnClickListener(new 1(this, (TextView)this.findViewById(2131362093)));
43      }
44
45      public native String stringFromJNI();
46  }

```

- Nhận xét:

+ Logic so sánh của bài này như sau, flag sẽ được in ra nếu $(x = a * b) > \text{var5} + 150$ với var5 là 1 số int random trong khoảng 0 -> 70.

+ Vì a b cố định, nên khiến x luôn là 50, như thế thì var5 random kiểu gì thì x cũng không thỏa mãn được.

+ Vậy thì ta chỉ cần hook và sửa lại hàm nextInt(int) cho return -150 là được.

- Code python để xử lý:

```
import frida
import sys
import time

def onMessage(message, data):
    print(message)

# Đặt tên package của ứng dụng
package = "com.revo.evabs"

# Kết nối tới thiết bị Android qua USB
device = frida.get_usb_device()

# Lấy danh sách tiến trình và attach vào PID của ứng dụng
try:
    # Nếu ứng dụng đã chạy, attach trực tiếp
    process = device.attach(package)
except frida.ProcessNotFoundError:
    # Nếu chưa chạy, khởi động ứng dụng trước khi attach
    print(f"[!] Waiting for Package '{package}'")
    pid = device.spawn([package])
    device.resume(pid)
    time.sleep(1) # Đợi ứng dụng khởi động
    process = device.attach(pid)

# Mã JavaScript để hook
jscode = """
Java.perform(function () {
    send("[ - ] Starting hooks java.util.Random.nextInt");
    var random = Java.use("java.util.Random");
    random.nextInt.overload("int").implementation = function(var_1) {
        return -150;
    };
});
"""

# Tạo script và load nó
script = process.create_script(jscode)
script.on("message", onMessage)
print("[*] Hooking", package)
script.load()
```

```
# Đợi người dùng nhập
sys.stdin.read()
```

- Giải thích:

+ **Kết nối đến điện thoại Android:** Sử dụng Frida để kết nối qua cáp USB.

+ **Tìm và gắn vào ứng dụng:**

- Nếu ứng dụng đang chạy, Frida sẽ gắn (attach) vào.
- Nếu chưa chạy, Frida sẽ khởi động ứng dụng và đợi nó khởi động xong.

+ **Inject đoạn script JavaScript:** Script này thay đổi cách hoạt động của hàm java.util.Random.nextInt, khiến hàm luôn trả về -150.

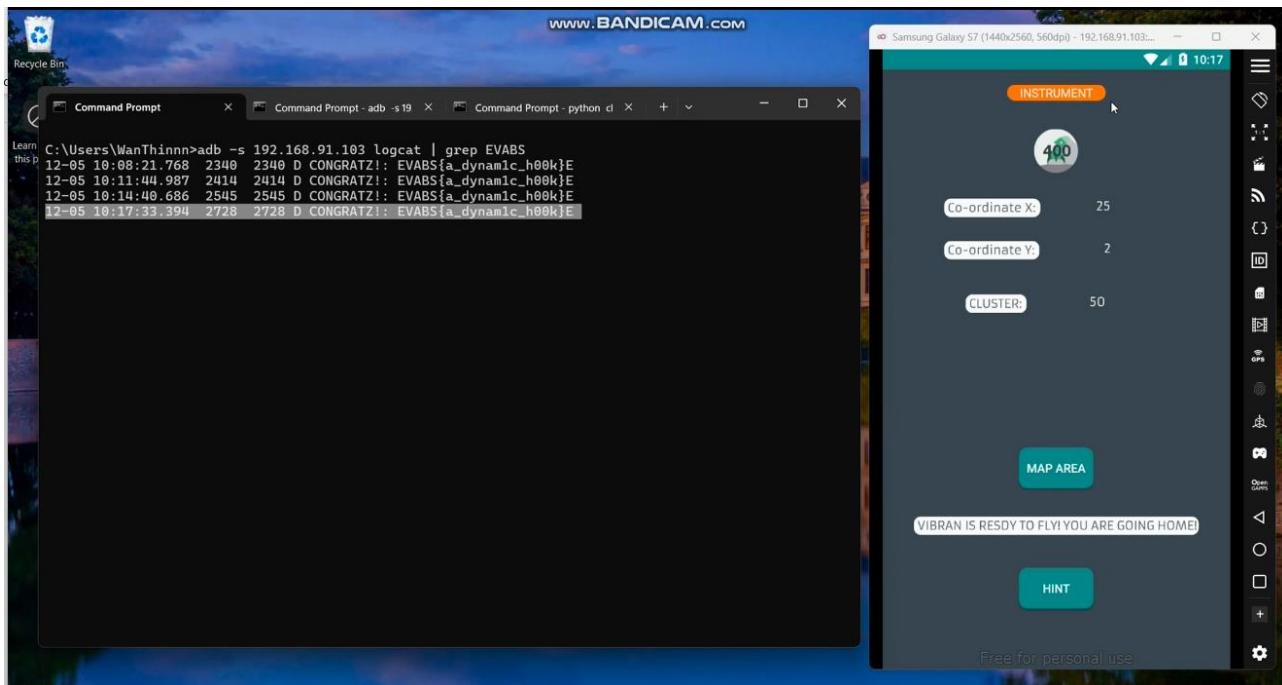
+ **In log ra màn hình:** Những gì script phát hiện hoặc thay đổi sẽ được in ra.

+ **Giữ chương trình chạy:** Frida tiếp tục hoạt động cho đến khi ta dừng nó.

- Chạy chương trình:

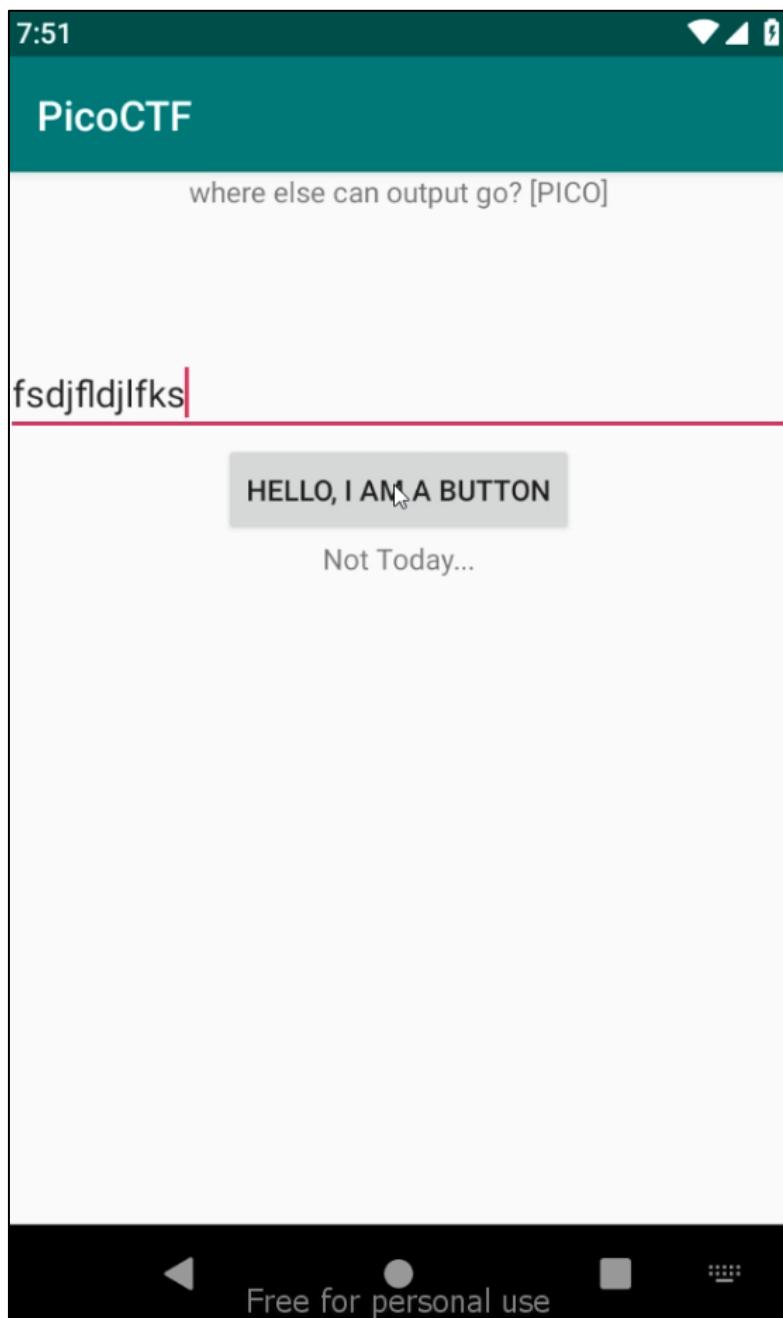
```
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>python challenge-1-level-12.py
[!] Waiting for Package 'com.revo.evabs'
[*] Hooking com.revo.evabs
{'type': 'send', 'payload': '[-] Starting hooks java.util.Random.nextInt'}
```

- Kết quả:



CHALLENGES 2**Level 1**

- Video thực hiện: <https://youtu.be/pcx9J0DgwXg>
- Flag: picoCTF{a.moose.once.bit.my.sister}
- Thực hiện vào chương trình, nhập input đưa vào chương trình rồi ấn vào button để kích hoạt sự kiện của button



- Truy cập vào xem log của chương trình

```
[4] Command Prompt - adb logcat
2-04 19:51:00.092    727 1944 E libEGL : load_driver(/vendor/lib/egl/libGLES_emulation.so): dlopen failed: library "/vendor/lib/egl/libGLES_emulation.so" not found
2-04 19:51:00.092    727 1944 D vndksupport: Loading /vendor/lib/egl/libGLES_emulation.so from current namespace instead of sphal namespace.
2-04 19:51:00.094    727 1944 D libEGL : loaded /vendor/lib/egl/libGLES_emulation.so
2-04 19:51:00.094    727 1944 D vndksupport: Loading /vendor/lib/egl/libGLESv1_CM_emulation.so from current namespace instead of sphal namespace.
2-04 19:51:00.094    727 1944 D libEGL : loaded /vendor/lib/egl/libGLESv1_CM_emulation.so
2-04 19:51:00.096    520 520 E : open_verbose:32: Could not open '/dev/goldfish_pipe': No such file or directory
2-04 19:51:00.096    520 520 D gralloc_ranchu: gralloc_alloc: Creating ashmem region of size 3788800
2-04 19:51:00.101    520 520 E : open_verbose:32: Could not open '/dev/goldfish_pipe': No such file or directory
2-04 19:51:00.102    520 520 D gralloc_ranchu: gralloc_alloc: Creating ashmem region of size 3788800
2-04 19:51:00.105    727 1944 D vndksupport: Loading /vendor/lib/egl/libGLESv2_emulation.so from current namespace instead of sphal namespace.
2-04 19:51:00.106    727 1944 D libEGL : loaded /vendor/lib/egl/libGLESv2_emulation.so
2-04 19:51:00.107    520 520 E : open_verbose:32: Could not open '/dev/goldfish_pipe': No such file or directory
2-04 19:51:00.112    727 727 I RenderThread: type=1400 audit(0.0:813): avc: denied { write } for name="local_opengl" dev="/tmpfs" ino=13372 scontext=u:r:untrusted_app_25:s0:c512,c768 tcontext=u:object_r:socket_file_t tperm=000
2-04 19:51:00.112    727 727 I RenderThread: type=1400 audit(0.0:814): avc: denied { connectto } for path="/dev/socket/local_opengl" scontext=u:r:untrusted_app_25:s0:c512,c768 tcontext=u:object_r:socket_file_t tperm=000
2-04 19:51:00.119    727 1944 D HostConnection: HostConnection::get() New Host Connection established 0xde585840, pid 727, tid 1944
2-04 19:51:00.121    471 1945 D local_opengl: Sending id 19 to host
2-04 19:51:00.122    727 1944 D HostConnection: HostComposition ext ANDROID_EMU_host_composition_v1 ANDROID_EMU_host_composition_v2 ANDROID_EMU_async_unmap_buffer ANDROID_EMU_sync_buffer_external_ess3 GL_OES_vertex_array_object GL_KHR_texture_compression_astc_ldr ANDROID_EMU_host_side_tracing ANDROID_EMU_async_frame_commands ANDROID_EMU_gles_max_version_3_1
2-04 19:51:00.122    727 1944 E : open_verbose:32: Could not open '/dev/goldfish_pipe': No such file or directory
2-04 19:51:00.122    727 1944 W : Process pipe failed
2-04 19:51:00.129    727 1944 I ConfigStore: android::hardware::configstore::V1_0::ISurfaceFlingerConfigs::hasWideColorDisplay retrieved: 0
2-04 19:51:00.129    727 1944 I ConfigStore: android::hardware::configstore::V1_0::ISurfaceFlingerConfigs::hasHDRDisplay retrieved: 0
2-04 19:51:00.129    727 1944 I OpenGLRenderer: Initialized EGL, version 1.4
2-04 19:51:00.129    727 1944 D OpenGLRenderer: Swap behavior 1
2-04 19:51:00.130    727 1944 W OpenGLRenderer: Failed to choose config with EGL_SWAP_BEHAVIOR_PRESERVED, retrying without...
2-04 19:51:00.130    727 1944 D OpenGLRenderer: Swap behavior 0
2-04 19:51:00.139    727 1944 D EGL_emulation: eglCreateContext: 0xde585800: maj 3 min 1 rcv 4
2-04 19:51:00.154    727 1944 E : open_verbose:32: Could not open '/dev/goldfish_pipe': No such file or directory
2-04 19:51:00.155    727 1944 D EGL_emulation: eglMakeCurrent: 0xde585800: ver 3 1 (tinfo 0xde583fa0) (first time)
2-04 19:51:00.208    727 1944 D vndksupport: Loading /vendor/lib/hw/android.hardware.graphics.mapper@2.0-impl.so from current namespace instead of sphal namespace.
2-04 19:51:00.208    727 1944 D vndksupport: Loading /vendor/lib/hw/gralloc.ranchu.so from current namespace instead of sphal namespace.
2-04 19:51:00.209    727 1944 D HostConnection: createUnique: call
2-04 19:51:00.209    727 1944 D HostConnection: HostConnection::get() New Host Connection established 0xde585d80, pid 727, tid 1944
2-04 19:51:00.209    471 1946 D local_opengl: Sending id 20 to host
2-04 19:51:00.203    727 1944 D HostConnection: HostComposition ext ANDROID_EMU_host_composition_v1 ANDROID_EMU_host_composition_v2 ANDROID_EMU_async_unmap_buffer ANDROID_EMU_sync_buffer_external_ess3 GL_OES_vertex_array_object GL_KHR_texture_compression_astc_ldr ANDROID_EMU_host_side_tracing ANDROID_EMU_async_frame_commands ANDROID_EMU_gles_max_version_3_1
2-04 19:51:00.992    537 537 I vinput : type=1400 audit(0.0:815): avc: denied { read } for path="socket:[13579]" dev="sockfs" ino=13579 scontext=u:r:vinput:s0 tcontext=u:r:vinput:s0 tperm=000
2-04 19:51:01.170    514 514 W genymotion_audio: Not supplying enough data to HAL, expected position 771197 , only wrote 618480
2-04 19:51:01.229    1921 1921 I PICO : picoTf(a.moose.once.my.sister)
2-04 19:51:01.231    514 514 W genymotion_audio: Not supplying enough data to HAL, expected position 618538 , only wrote 618480
2-04 19:51:01.484    527 527 I wifi@0.0-service: type=1400 audit(0:0:816): avc: denied { write } for scontext=u:r:hal_wifi_default:s0 tcontext=u:r:hal_wifi_default:s0 tclass=netlink_route_
2-04 19:51:01.484    527 527 I wifi@0.0-service: type=1400 audit(0:0:817): avc: denied { nlmsg_read } for scontext=u:r:hal_wifi_default:s0 tcontext=u:r:hal_wifi_default:s0 tclass=netlink_
2-04 19:51:01.484    527 527 I wifi@0.0-service: type=1400 audit(0:0:818): avc: denied { read } for scontext=u:r:hal_wifi_default:s0 tcontext=u:r:hal_wifi_default:s0 tclass=netlink_route_
2-04 19:51:01.441    514 514 W genymotion_audio: Not supplying enough data to HAL, expected position 824575 , only wrote 775696
```

- Thực hiện tìm kiếm thông tin liên quan đến cụm từ picoCTF để thấy được flag cần tìm

```
C:\Users\namphuong>adb logcat | grep picoCTF  
12-04 19:51:01.229 1921 1921 I PICO    : picoCTF{a.moose.once.bit.my.sister}  
12-04 19:51:15.377 1921 1921 T PICO    : picoCTF{a moose once bit my sister}
```

Level 2

- Video thực hiện: https://youtu.be/52Qc_dHoXx4

- Đầu tiên ta decompile file two.apk

```
D:\Android\Challenges>ls
EVABSV5.apk five.apk four.apk mtest one.apk three.apk two.apk

D:\Android\Challenges>apktool d two.apk
I: Using Apktool 2.10.0 on two.apk with 22 thread(s).
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\PC\AppData\Local\apktool\framework\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...

D:\Android\Challenges>ls
EVABSV5.apk five.apk four.apk mtest one.apk three.apk two two.apk
```

- Ta thấy thư mục two xuất hiện, tìm trong thư mục ta thấy địa chỉ đáng ngờ

You have Docker installed on your system. Do you want to install the recommended 'Dev Containers' extension from Microsoft? [Yes] [No]

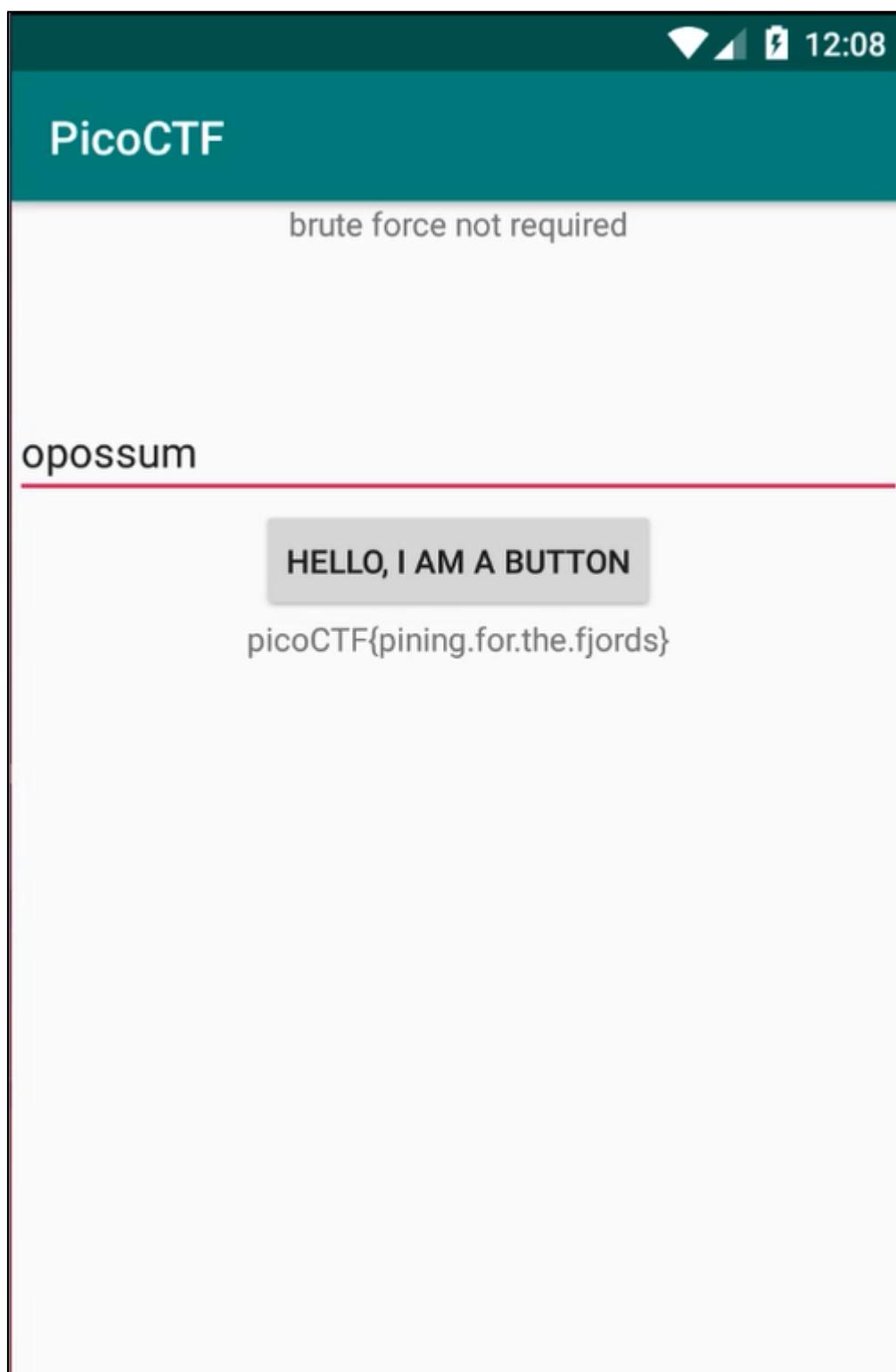
- Thực hiện tìm kiếm ta thấy nó được gắn liền với name password\

```
ubuntu@hohuy:~/Android/two$ grep "opossum"
^C
ubuntu@hohuy:~/Android/two$ grep -r "0x7f0b002f"
smali/com/hellocmu/picoctf/FlagstaffHill.smali:    const v0, 0x7f0b002f
smali/com/hellocmu/picoctf/R$string.smali:.field public static final password:I
= 0x7f0b002f
res/values/public.xml:      <public type="string" name="password" id="0x7f0b002f"
/>
```

- Tiếp tục tìm kiếm “password” ta tìm được key là “opossum”

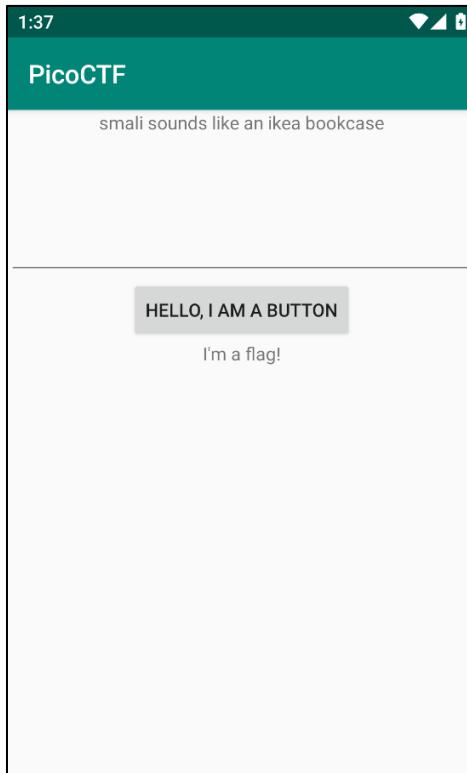
```
ubuntu@hohuy:~/Android/two$ grep -r "password"
smali/com/hellocmu/picoctf/FlagstaffHill.smali:    .local v0, "password":Ljava/lang/String;
smali/com/hellocmu/picoctf/R$string.smali:.field public static final password:I
= 0x7f0b002f
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:    .pa
ram p1, "password"    # Z
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:    con
st-string v2, ";" password: "
res/values/public.xml:    <public type="string" name="password" id="0x7f0b002f"
/>
res/values/strings.xml:    <string name="password">opossum</string>
ubuntu@hohuy:~/Android/two$ █
```

- Lấy được flag:

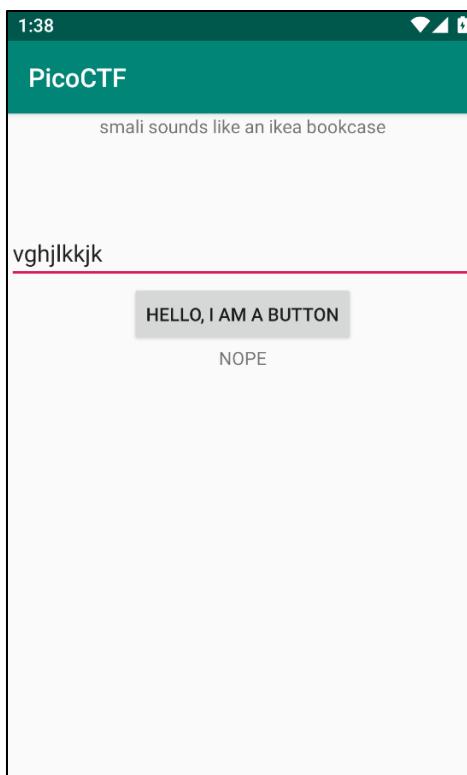


Level 3

- Video thực hiện: <https://youtu.be/63HxR0RTrgM>
- Giao diện ban đầu khi mở ứng dụng PicoCTF (file three.apk).



- Giao diện sau khi nhập một chuỗi ngẫu nhiên vào ô nhập input và bấm nút HELLO, I AM A BUTTON.



- Dùng Bytecode Viewer mở three.apk, ta truy cập vào đường dẫn và xem nội dung file com/hellocmu/picoctf/MainActivity.class.

Lab 4: Pentesting Android Applications

```

1 package com.hellocmu.picocft;
2
3 import android.content.Context;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import android.widget.EditText;
8 import android.widget.TextView;
9 import androidx.appcompat.app.AppCompatActivity;
10
11 public class MainActivity extends AppCompatActivity {
12     Button button;
13     Context ctx;
14     TextView text_bottom;
15     EditText text_input;
16     TextView text_top;
17
18     public void onClick(View var1) {
19         String var2 = this.text_input.getText().toString();
20         this.text_bottom.setText(FlagstaffHill.getFlag(var2, this.ctx));
21     }
22
23     protected void onCreate(Bundle var1) {
24         super.onCreate(var1);
25         this.setContentView(2131296284);
26         this.text_top = (TextView)this.findViewById(2131165322);
27         this.text_bottom = (TextView)this.findViewById(2131165320);
28         this.text_input = (EditText)this.findViewById(2131165321);
29         this.ctx = this.getApplicationContext();
30         System.loadLibrary("hell");
31         this.text_top.setText(2131427368);
32     }
33 }

```

- Ta thấy dòng code thứ 20 trong tập tin MainActivity.class có đề cập đến getFlag() và FlagstaffHill, tiến hành xem file FlagstaffHill.class.

```

1 package com.hellocmu.picocft;
2
3 import android.content.Context;
4
5 public class FlagstaffHill {
6     public static String getFlag(String var0, Context var1) {
7         String[] var6 = new String[]{"weather", "ogg", "onclick", "nitr", "aching", "dismess"};
8         int var2 = 3 - 3;
9         int var4 = 3 / 3 + var2;
10        int var1 = var4 + var2 - var2;
11        int var5 = 3 + var3;
12
13        return var0.equals("") ? concat(var6[var5]).concat(".").concat(var6[var4]).concat(".").concat(var6[var2]).concat(".").concat(var6[var5 + var2 - var4]).concat(".").concat(var6[3]).concat(".").concat(var6[var3])) ? sesame(var0) : "NOPE";
14    }
15
16    public static native String sesame(String var0);
17
18 }

```

- Dòng return của FlagstaffHill.class.

```

return var0.equals("") ? concat(var6[var5]).concat(".").concat(var6[var4]).concat(".").concat(var6[var2]).concat(".").concat(var6[var5 + var2 - var4]).concat(".").concat(var6[3]).concat(".").concat(var6[var3])) ? sesame(var0) : "NOPE";

```

- Thông qua tính toán, ta có:

$$var5 = 3 - 3 = 0$$

$$var4 = 3 / 3 + var5 = 1 + 0 = 1$$

$$var3 = var4 + var4 - var5 = 1 + 1 - 0 = 2$$

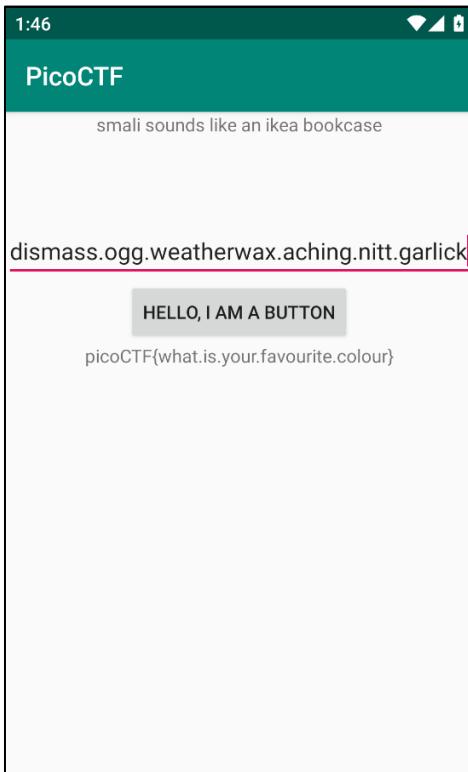
$$var2 = 3 + var3 = 3 + 2 = 5$$

$$var2 + var5 - var4 = 5 + 0 - 1 = 4$$

→ Thứ tự các giá trị trong var6 được gọi theo dòng return sẽ là: 5.1.0.4.3.2

→ Chuỗi tương ứng sẽ là: dismiss.ogg.weatherwax.aching.nitt.garlick

- Quay lại ứng dụng, nhập dismiss.ogg.weatherwax.aching.nitt.garlick và nhấn button, ta thu được flag.



Level 4

- Video thực hiện: <https://youtu.be/lnZIpcO4kRk>
- Flag: picoCTF{tis.but.a.scratch}
- Sử dụng Bytecode-Viewer để xem mã nguồn của chương trình, truy cập vào đường dẫn four.apk/com/hellocmu/FlagStaffHill.class

```

1 package com.hellocmu.picoctf;
2
3 import android.content.Context;
4
5 public class FlagstaffHill {
6     public static native String cilantro(String var0);
7
8     public static String getFlag(String var0, Context var1) {
9         return nope(var0);
10    }
11
12    public static String nope(String var0) {
13        return "don't wanna";
14    }
15
16    public static String yep(String var0) {
17        return cilantro(var0);
18    }
19
20
}

```

- Dựa trên mã nguồn của chương trình có thể thấy hàm getFlag đang trả đến sai hàm là hàm nope

=> ta phải đổi lại mã nguồn sao cho hàm getFlag sẽ trả về hàm yep trong chương trình

- Dùng apktool để decompile file apk, sau đó vào đường dẫn ~/four/com/hellocmu/FlagStaffHill.smali để chỉnh sửa như sau

```

new 1 FlagstaffHill.smali
7 .method public constructor <init>()V
8     .locals 0
9
10    .line 6
11    invoke-direct {p0}, Ljava/lang/Object;-><init>()V
12
13    return-void
14 .end method
15
16 .method public static native cilantro(Ljava/lang/String;)Ljava/lang/String;
17 .end method
18
19 .method public static getFlag(Ljava/lang/String;Landroid/content/Context;)Ljava/lang/String;
20     .locals 1
21     .param p0, "input"    # Ljava/lang/String;
22     .param p1, "ctx"      # Landroid/content/Context;
23
24     .line 19
25     invoke-static {p0}, Lcom/hellocmu/picoctf/FlagstaffHill;->yep(Ljava/lang/String;)Ljava/lang/String;
26
27     move-result-object v0
28
29     .line 20
30     .local v0, "flag":Ljava/lang/String;
31     return-object v0
32 .end method
33
34 .method public static nope(Ljava/lang/String;)Ljava/lang/String;
35     .locals 1
36     .param p0, "input"    # Ljava/lang/String;
37
38     .line 11
39     const-string v0, "don't wanna"
40
41     return-object v0
42 .end method
43
44 .method public static yep(Ljava/lang/String;)Ljava/lang/String;
45     .locals 1

```

- Thực hiện build lại file apk mới và kí lại nó bằng key đã tạo sẵn ở các labs trước

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.26120.1252]
(c) Microsoft Corporation. All rights reserved.

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\Challenges>apktool d four.apk
I: Using Apktool 2.10.0 on four.apk with 16 thread(s).
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: C:\Users\namphuong\AppData\Local\apktool\framework\1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...

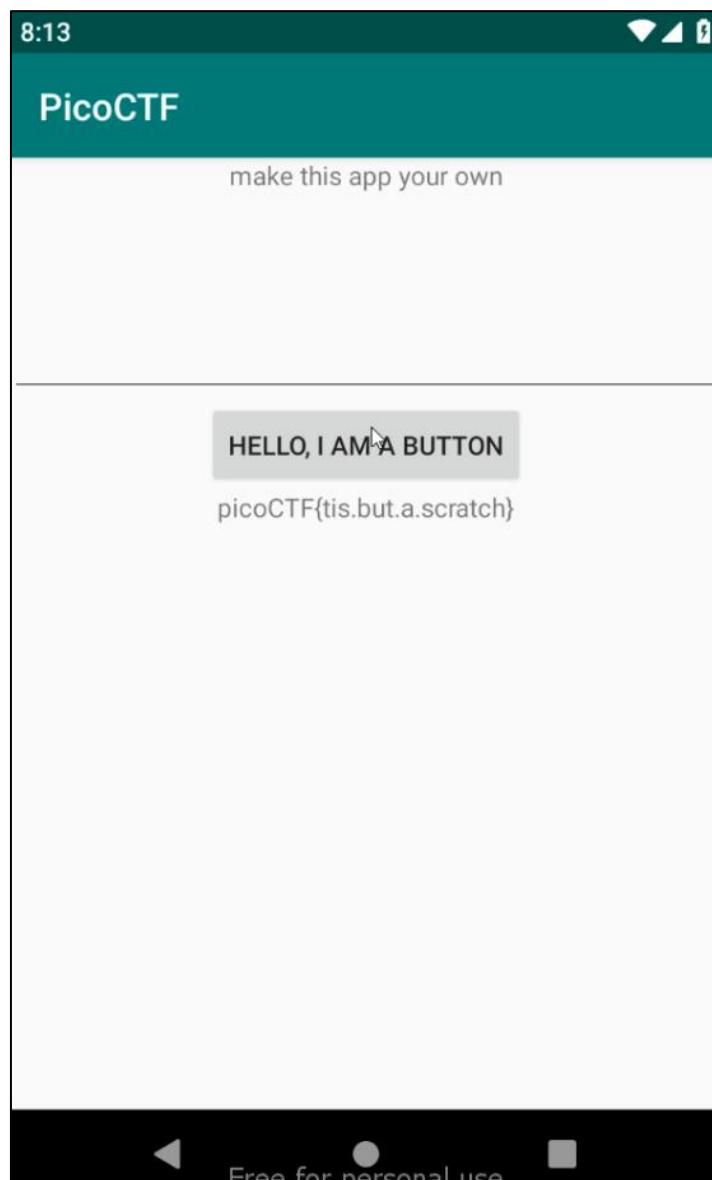
C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\Challenges>apktool b four -o newfour.apk
I: Using Apktool 2.10.0 with 16 thread(s).
I: Checking whether sources has changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources has changed...
I: Building resources...
I: Copying libs... (/lib)
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk into: newfour.apk

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\Challenges>jarsigner -keystore "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\Challenges\key.keystore" -storepass namphuong "C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\Challenges\newfour.apk" CTf9
Warning:
The signer's certificate is self-signed.

C:\Code\Bao mat web va ung dung - NT213.P11.ANTT\Lab\W4\Lab4\Lab4\Challenges\Challenges>adb install newfour.apk
Performing Streamed Install
Success

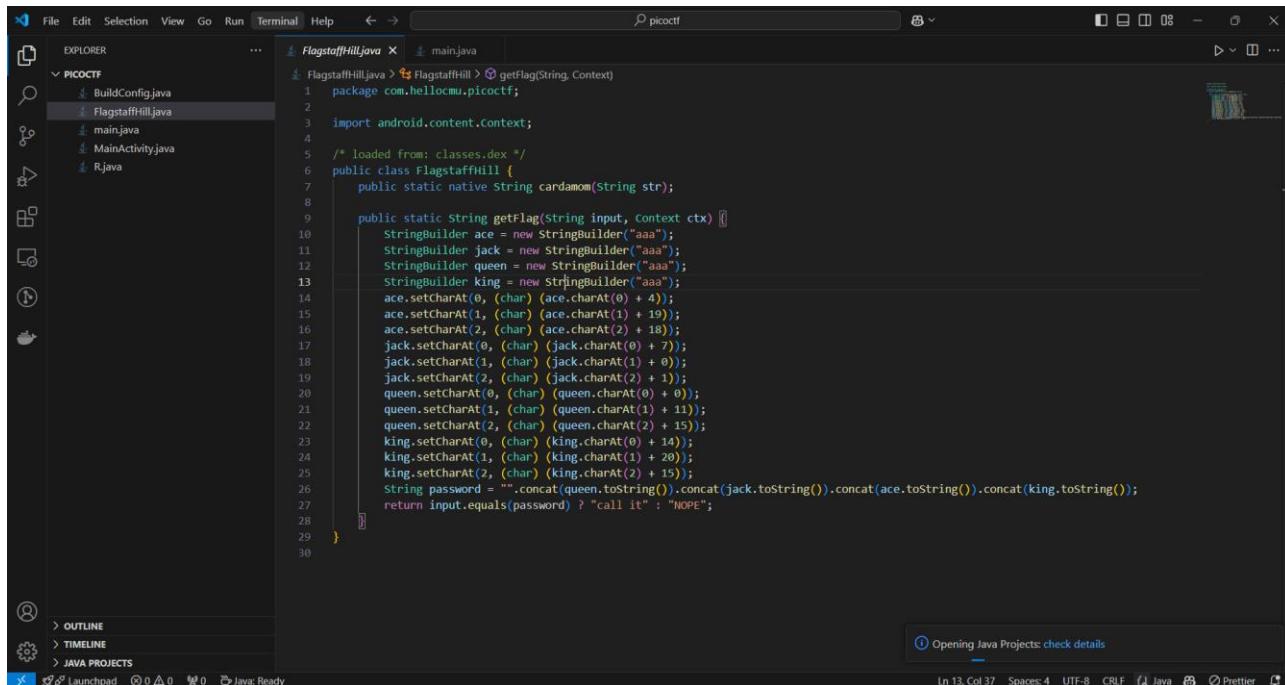
```

- Kiểm thử kết quả trong file apk mới khi ấn vào button



Level 5

- Video thực hiện: <https://youtu.be/95SyEukHh2Y>
- Đầu tiên ta sẽ decompile chương trình và xem thông tin trong FlagstaffHill:



```

File Edit Selection View Go Run Terminal Help ← → picoctf

EXPLORER FlagstaffHill.java main.java
BuildConfig.java FlagstaffHill.java
main.java MainActivity.java R.java

FlagstaffHill.java > FlagstaffHill > getFlag(String, Context)
1 package com.hellocmu.picoctf;
2
3 import android.content.Context;
4
5 /* loaded from: classes.dex */
6 public class FlagstaffHill {
7     public static native String cardamom(String str);
8
9     public static String getFlag(String input, Context ctx) {
10         StringBuilder ace = new StringBuilder("aaa");
11         StringBuilder jack = new StringBuilder("aaa");
12         StringBuilder queen = new StringBuilder("aaa");
13         StringBuilder king = new StringBuilder("aaa");
14         ace.setCharAt(0, (char) (ace.charAt(0) + 4));
15         ace.setCharAt(1, (char) (ace.charAt(1) + 19));
16         ace.setCharAt(2, (char) (ace.charAt(2) + 18));
17         jack.setCharAt(0, (char) (jack.charAt(0) + 7));
18         jack.setCharAt(1, (char) (jack.charAt(1) + 0));
19         jack.setCharAt(2, (char) (jack.charAt(2) + 1));
20         queen.setCharAt(0, (char) (queen.charAt(0) + 0));
21         queen.setCharAt(1, (char) (queen.charAt(1) + 11));
22         queen.setCharAt(2, (char) (queen.charAt(2) + 15));
23         king.setCharAt(0, (char) (king.charAt(0) + 14));
24         king.setCharAt(1, (char) (king.charAt(1) + 20));
25         king.setCharAt(2, (char) (king.charAt(2) + 15));
26         String password = ".concat(queen.toString()).concat(jack.toString()).concat(ace.toString()).concat(king.toString());
27         return input.equals(password) ? "call it" : "NOPE";
28     }
29 }

```

Opening Java Projects: check details

- Ở đây ta thấy thông tin chính là kiểm tra password, dựa vào đây ta sẽ thực hiện chỉnh sửa code để chương trình trả về password:



```

main.java
1 public class FlagstaffHill {
2     public static native String cardamom(String str);
3
4     public static String getFlag() {
5         StringBuilder ace = new StringBuilder(str:"aaa");
6         StringBuilder jack = new StringBuilder(str:"aaa");
7         StringBuilder queen = new StringBuilder(str:"aaa");
8         StringBuilder king = new StringBuilder(str:"aaa");
9         ace.setCharAt(index:0, (char) (ace.charAt(index:0) + 4));
10        ace.setCharAt(index:1, (char) (ace.charAt(index:1) + 19));
11        ace.setCharAt(index:2, (char) (ace.charAt(index:2) + 18));
12        jack.setCharAt(index:0, (char) (jack.charAt(index:0) + 7));
13        jack.setCharAt(index:1, (char) (jack.charAt(index:1) + 0));
14        jack.setCharAt(index:2, (char) (jack.charAt(index:2) + 1));
15        queen.setCharAt(index:0, (char) (queen.charAt(index:0) + 0));
16        queen.setCharAt(index:1, (char) (queen.charAt(index:1) + 11));
17        queen.setCharAt(index:2, (char) (queen.charAt(index:2) + 15));
18        king.setCharAt(index:0, (char) (king.charAt(index:0) + 14));
19        king.setCharAt(index:1, (char) (king.charAt(index:1) + 20));
20        king.setCharAt(index:2, (char) (king.charAt(index:2) + 15));
21        String password = ".concat(queen.toString()).concat(jack.toString()).concat(ace.toString()).concat(king.toString());
22        return password;
23    }
Run | Debug
24    public static void main(String[] args) {
25        System.out.println("Password: ");
26        // In kết quả
27        System.out.println(getFlag());
28    }
}

```

- Chạy chương trình và ta nhận được password là alphabetsoup



```

C:\Users\WanThinnn\Downloads\jadex-gui-1.5.1-win\sources\com\hellocmu\picoctf>java main.java
Password:
alphabetsoup

C:\Users\WanThinnn\Downloads\jadex-gui-1.5.1-win\sources\com\hellocmu\picoctf>

```

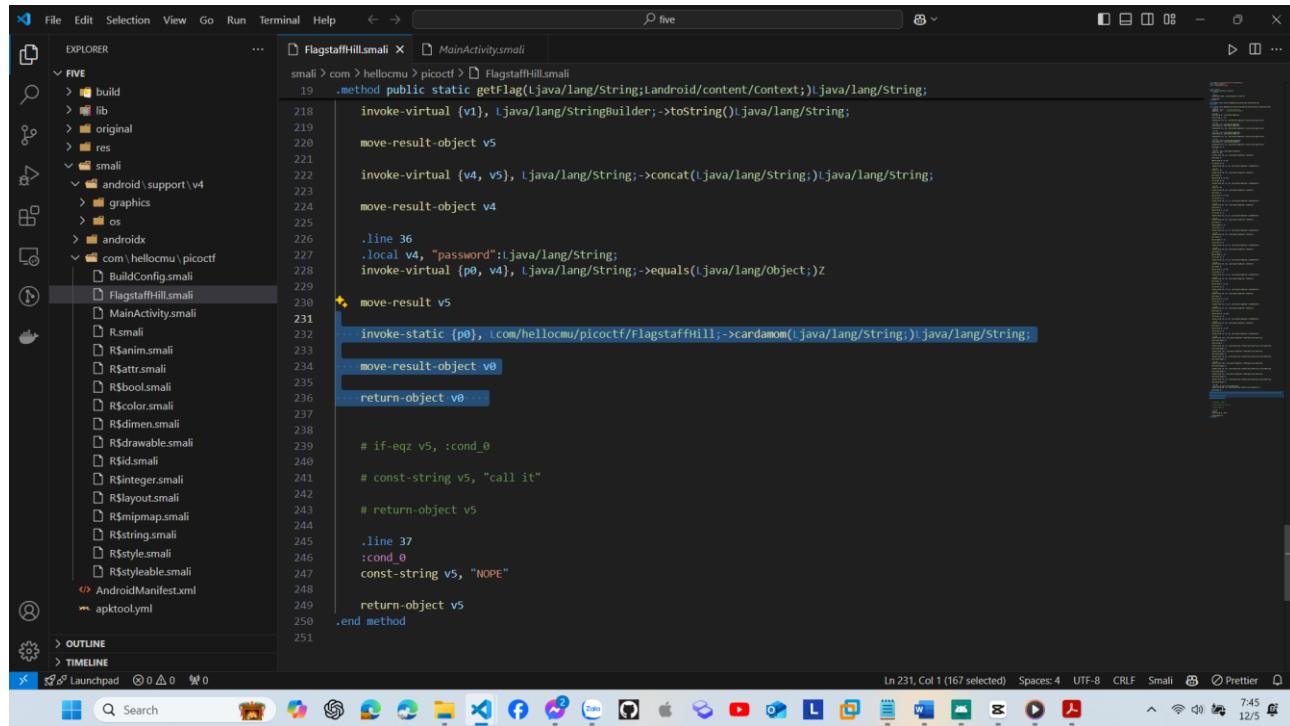
- Tiếp tục kiểm tra trong code smali thì ta có được thông tin về các dòng dòng 232 tới dòng 236 trả về không đúng flag:

```

19 .method public static getFlag(Ljava/lang/String;Landroid/content/Context;)Ljava/lang/String;
20     invoke-virtual {v0}, Ljava/lang/StringBuilder;->toString()Ljava/lang/String;
21
22     move-result-object v5
23
24     invoke-virtual {v4, v5}, Ljava/lang/String;->concat(Ljava/lang/String;)Ljava/lang/String;
25
26     move-result-object v4
27
28     invoke-virtual {v1}, Ljava/lang/StringBuilder;->toString()Ljava/lang/String;
29
30     move-result-object v5
31
32     invoke-virtual {v4, v5}, Ljava/lang/String;->concat(Ljava/lang/String;)Ljava/lang/String;
33
34     move-result-object v4
35
36     .line 36
37     .local v4, "password":Ljava/lang/String;
38     invoke-virtual {p0, v4}, Ljava/lang/String;->equals(Ljava/lang/Object;)Z
39
40     move-result v5
41
42     if-eqz v5, :cond_0
43
44     const-string v5, "call it"
45
46     return-object v5
47
48     .line 37
49     :cond_0
50     const-string v5, "NOPE"
51
52     return-object v5
53 .end method

```

- Nên ta sẽ thực hiện chỉnh sửa code lại bằng cách invoke hàm cardamom, sau đó gán kết quả vào v0 và trả về v0:



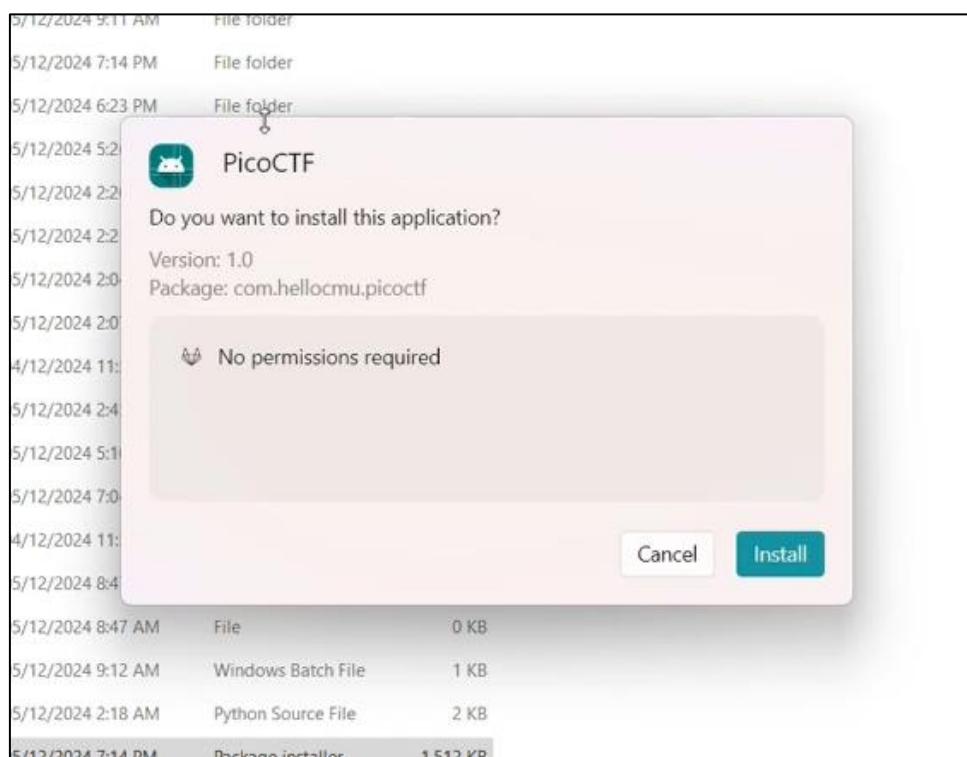
- Thực hiện build lại file, tạo key và ký vào file apk:

```
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>apktool b five -o five_v2.apk
I: Using Apktool 2.10.0 with 12 thread(s).
I: Checking whether sources has changed...
I: Checking whether resources has changed...
I: Building resources...
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk into: five_v2.apk
```

```
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>rebuild_app.bat five_v2.apk
C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>keytool -genkeypair -v -keystore key.keystore -alias publishingdoc -keyalg RSA -keysize 2048 -validity 10000
Enter keystore password:
Re-enter new password:
Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press ENTER to use the default value in braces.
What is your first and last name?
What is the name of your organizational unit?
What is the name of your organization?
What is the name of your City or Locality?
What is the name of your State or Province?
What is the two-letter country code for this unit?
Is CN=Thien Lai, OU=UIT, O=UIT, L=HCM, ST=HCM, C=HCM correct?
[no]: yes
Generating 2,048 bit RSA key pair and self-signed certificate (SHA384withRSA) with a validity of 10,000 days
for: CN=Thien Lai, OU=UIT, O=UIT, L=HCM, ST=HCM, C=HCM
[Storing key.keystore]

C:\Users\WanThinnn\Documents\UIT\Nam_3\HK1\NT213-BMWUD\Thuc_hanh\Lab_4>jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore ./key.keystore five_v2.apk publishingdoc
Enter Passphrase for keystore: |
```

- Install app vào Windows Subsystem For Android (WSA) thông qua WSA-pacman (cách này nhanh hơn xíu, do máy em có sẵn WSA, nên em install thẳng vào PC luôn, không cần thông qua adb và genymotion)



- Ta lấy được cờ là picoCTF{not.particularly.silly}, cạnh bên là phiên bản cũ của app, nó sẽ không in ra cờ.

