

BÁO CÁO THỰC HÀNH

Môn học: Bảo mật web và ứng dụng

Lab 4: Pentesting Android Applications

Bài Tập Làm Ở Nhà

GVHD: Nghi Hoàng Khoa

1. **THÔNG TIN CHUNG:**

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT213.P11.ATCL.1

STT	Họ và tên	MSSV	Email
1	Tôn Thất Bình	21520639	21520639@gm.uit.edu.vn
2	Nguyễn Văn Hào	20521293	20521293@gm.uit.edu.vn

2. **NỘI DUNG THỰC HIỆN:**¹

STT	Công việc	Kết quả tự đánh giá
1	EVABS	100%
2	Droid	100%

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

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

BÁO CÁO CHI TIẾT

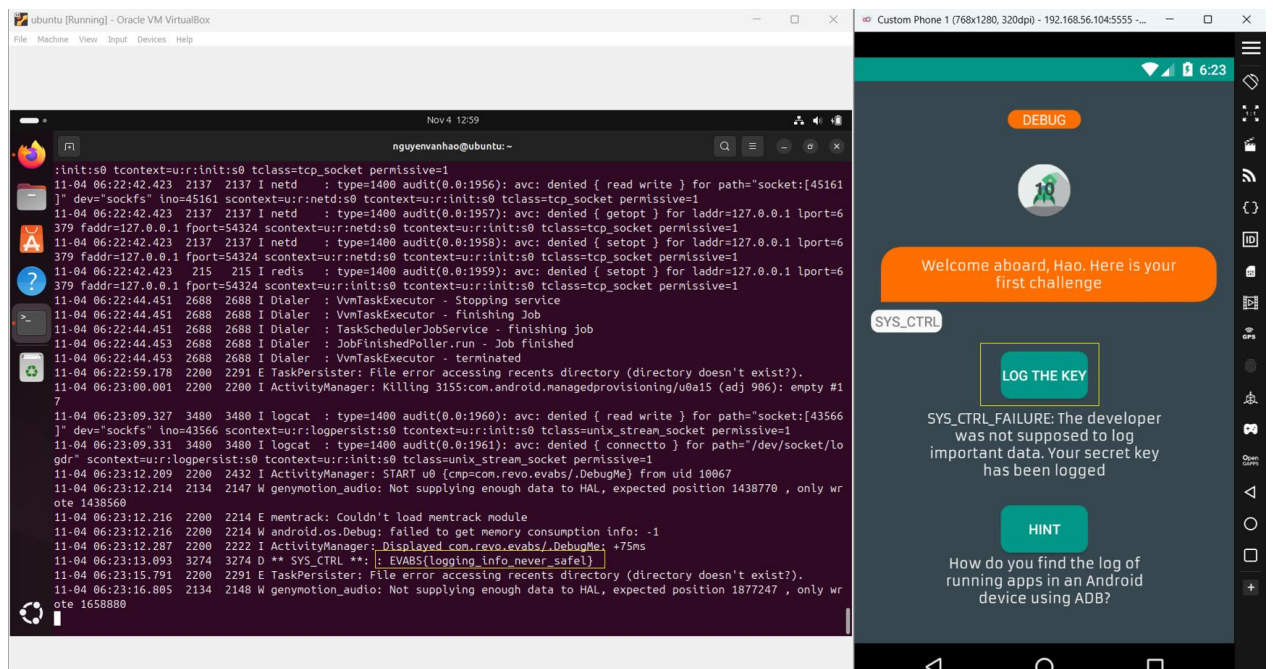
D.1 EVABS

Challenges 1: Debug Me

- Trước tiên dùng lệnh **adb logcat** để xem quá trình log :

```
nguyenvanhao@ubuntu:~$ adb logcat
```

- Ta thực hiện bấm vào LOG THE KEY và kiểm tra thấy tiến trình PID: 3274 thì ta có được plag cần tìm:



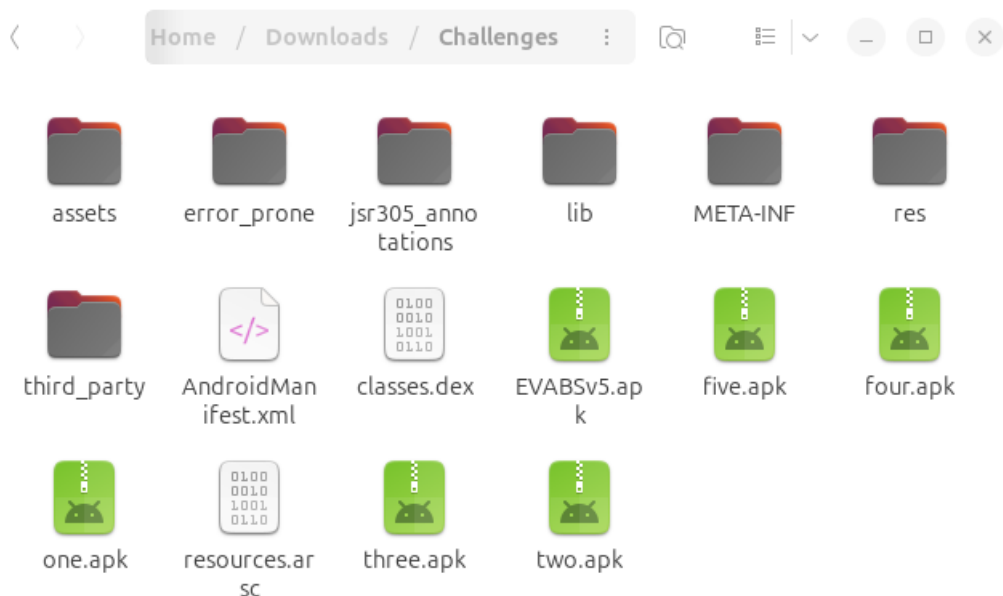
- Kết quả Flag: **EVABS{logging_info_never_safe!}**

Challenges 2: File Access

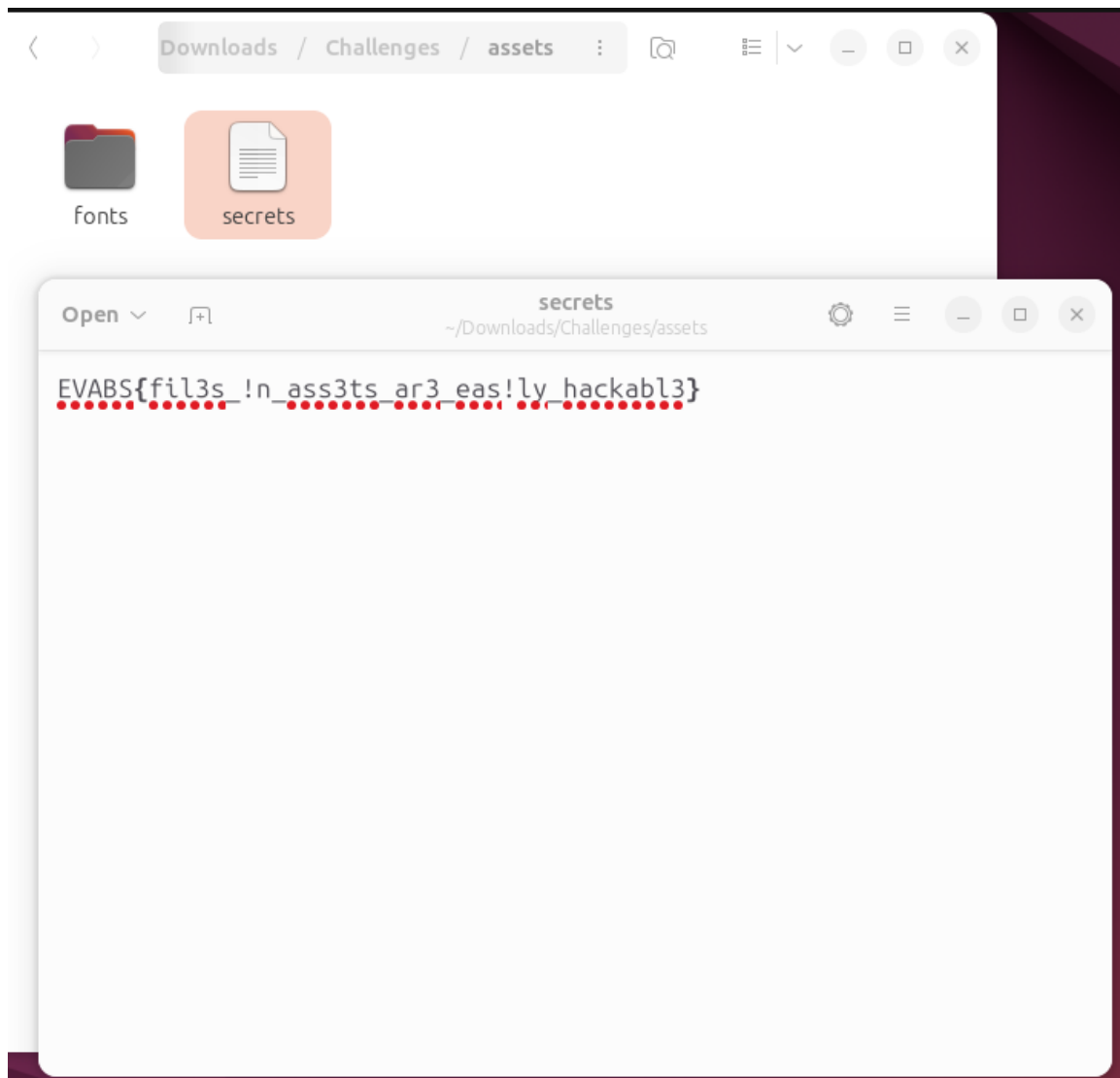
- Tiến hành unzip file bằng lệnh **unzip EVABSv5.apk**

```
nguyenvanhao@ubuntu: ~/Downloads/Challenges
nguyenvanhao@ubuntu:~/Downloads/Challenges$ unzip EVABSV5.apk
Archive:  EVABSV5.apk
  inflating: AndroidManifest.xml
  inflating: META-INF/CERT.RSA
  inflating: META-INF/CERT.SF
  inflating: META-INF/MANIFEST.MF
  inflating: assets/fonts/SR.otf
  inflating: assets/fonts/ssb.otf
  inflating: assets/fonts/trench100free.otf
  extracting: assets/secrets
  inflating: classes.dex
  inflating: error_prone/Annotations.gwt.xml
  inflating: jsr305_annotations/Jsr305_annotations.gwt.xml
  inflating: lib/arm64-v8a/libnative-lib.so
  inflating: lib/armeabi-v7a/libnative-lib.so
  inflating: lib/x86/libnative-lib.so
  inflating: lib/x86_64/libnative-lib.so
  inflating: res/anim-v21/design_bottom_sheet_slide_in.xml
  inflating: res/anim-v21/design_bottom_sheet_slide_out.xml
  inflating: res/anim/abc_fade_in.xml
  inflating: res/anim/abc_fade_out.xml
  inflating: res/anim/abc_grow_fade_in_from_bottom.xml
  inflating: res/anim/abc_popup_enter.xml
  inflating: res/anim/abc_popup_exit.xml
```

- Sau khi giải nén xong:



- Tiến hành kiểm tra file /assets/secrets:



- Kết quả Flag: **EVABS{fil3s_!n_ass3ts_ar3_eas!ly_hackabl3}**

Challenges 3: Strings

- Dùng apktool để thực hiện decompile file ta sử dụng lệnh **apktool d EVABsv5.apk**

```
nguyenvanhao@ubuntu: ~/Downloads/Challenges
nguyenvanhao@ubuntu:~/Downloads/Challenges$ apktool d EVABSV5.apk
I: Using Apktool 2.7.0-dirty on EVABSV5.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /home/nguyenvanhao/.local/share/apktool/framework/1.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
nguyenvanhao@ubuntu:~/Downloads/Challenges$
```

- Ta vào thư mục /res/values/strings.xml thì ta có được flag:

```
file:///home/nguyenvanhao/Downloads/Challenges/EVABSV5/res/values/strings.xml
<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_eye">
  M12,4.5C7.4,5 2.73,7.61 1,12c1.73,4.39 6,7.5 11,7.5s9.27,-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
  -5.5zM12,9c-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 L4.54,1.73 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$d</string>
<string name="status_bar_notification_info_overflow">999+</string>
<string name="the_evabs_api_key">EVABS{saf3ly_st0red_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>
```

- Kết quả Flag: **EVABS{saf3ly_st0red_in_Strings?}**

Challenges 4: Resources

- Ta thực hiện tại thư mục /EVABSV5/res của file decompile ta sử dụng lệnh **grep -r "EVABS{"** để xem flag theo gợi ý


```
nguyenvanhao@ubuntu: ~/Downloads/Challenges/EVABSV5/res$ grep -r "EVABS{"
layout-v17/activity_flagcheck.xml: <EditText android:textColor="@color/colorWhite" android:id="@id/editTextflag"
android:layout_width="wrap_content" android:layout_height="wrap_content" android:layout_marginLeft="8.0dip" an
droid:layout_marginTop="8.0dip" android:layout_marginRight="8.0dip" android:layout_marginBottom="8.0dip" android:
text="EVABS{" android:ems="10" android:inputType="textPersonName" android:textAlignment="center" android:layout_
marginStart="8.0dip" android:layout_marginEnd="8.0dip" app:layout_constraintBottom_toBottomOf="parent" app:layout_
_constraintEnd_toEndOf="parent" app:layout_constraintHorizontal_bias="0.503" app:layout_constraintStart_toStartOf
="parent" app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="0.401" />
layout-v17/activity_flagcheck.xml: <TextView android:textSize="10.0dip" android:textColor="@color/colorWhite"
android:id="@id/textViewnote" android:paddingLeft="10.0dip" android:paddingRight="10.0dip" android:layout_width="
310.0dip" android:layout_height="24.0dip" android:layout_marginLeft="15.0dip" android:layout_marginTop="8.0dip" a
ndroid:layout_marginRight="15.0dip" android:layout_marginBottom="36.0dip" android:text="NOTE: All flags are in the
format EVABS{some_text_here}." android:textAlignment="center" android:layout_marginStart="8.0dip" android:layou
t_marginEnd="8.0dip" app:layout_constraintBottom_toBottomOf="parent" app:layout_constraintEnd_toEndOf="parent" ap
p:layout_constraintStart_toStartOf="parent" app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertic
al_bias="0.962" />
values/strings.xml: <string name="the_evabs_api_key">EVABS{saf3ly_st0red_in_Strings?}</string>
raw/link.txt: EVABS{th!s_plac3_is_n0t_as_s3cur3_as_it_l00ks}
layout/activity_flagcheck.xml: <EditText android:textColor="@color/colorWhite" android:id="@id/editTextflag" a
ndroid:layout_width="wrap_content" android:layout_height="wrap_content" android:layout_marginLeft="8.0dip" androi
d:layout_marginTop="8.0dip" android:layout_marginRight="8.0dip" android:layout_marginBottom="8.0dip" android:text
="EVABS{" android:ems="10" android:inputType="textPersonName" app:layout_constraintBottom_toBottomOf="parent" ap
p:layout_constraintEnd_toEndOf="parent" app:layout_constraintHorizontal_bias="0.503" app:layout_constraintStart_t
oStartOf="parent" app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="0.401" />
layout/activity_flagcheck.xml: <TextView android:textSize="10.0dip" android:textColor="@color/colorWhite" andr
oid:id="@id/textViewnote" android:paddingLeft="10.0dip" android:paddingRight="10.0dip" android:layout_width="310.
0dip" android:layout_height="24.0dip" android:layout_marginLeft="15.0dip" android:layout_marginTop="8.0dip" andro
id:layout_marginRight="15.0dip" android:layout_marginBottom="36.0dip" android:text="NOTE: All flags are in the fo
rmat EVABS{some_text_here}." app:layout_constraintBottom_toBottomOf="parent" app:layout_constraintEnd_toEndOf="pa
rent" app:layout_constraintStart_toStartOf="parent" app:layout_constraintTop_toTopOf="parent" app:layout_constrai
ntVertical_bias="0.962" />
nguyenvanhao@ubuntu: ~/Downloads/Challenges/EVABSV5/res$
```

- Kết quả:
 - + Flag: **EVABS{th!s_plac3_is_n0t_as_s3cur3_as_it_l00ks}**
 - + Flag: **EVABS{s0me_t3xt_here}**

Challenges 5: Shares and Prefs

- Trước tiên ta gõ lệnh **adb shell** để vào command line của máy ảo Android.

```
nguyenvanhao@ubuntu: ~$ adb shell
genymotion:/ $
```

- Vào thư mục **/data/data/com.revo.evabs/shared_prefs** sau đó ta sử dụng lệnh **grep -r "EVABS{"** thì ta có được flag:

```
nguyenvanhao@ubuntu: ~$ 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 #
```

- Kết quả Flag: **EVABS{shar3d_pr3fs_c0uld_be_c0mpromiz3ds}**

Challenges 6: DB Leak

- Ta gõ lệnh **adb shell** để vào command line của máy ảo Android. Sau đó vào thư mục `/data/data/com.revo.evabs/databases` thì thấy `MAINFRAME_ACCESS` :

```
nguyenvanhao@ubuntu: ~  
nguyenvanhao@ubuntu:~$ adb shell  
genymotion:/ # cd /data/data/com.revo.evabs  
genymotion:/data/data/com.revo.evabs # ls  
cache code_cache databases lib shared_prefs  
genymotion:/data/data/com.revo.evabs # cd databases  
genymotion:/data/data/com.revo.evabs/databases # ls  
MAINFRAME_ACCESS MAINFRAME_ACCESS-journal  
genymotion:/data/data/com.revo.evabs/databases #
```

- Ta thực hiện pull database về:

```
nguyenvanhao@ubuntu: ~/Downloads/Challenges/EVABsv5  
nguyenvanhao@ubuntu:~/Downloads/Challenges/EVABsv5$ adb pull "/data/data/com.revo.evabs/databases/MAINFRAME_ACCESS"  
/data/data/com.revo.evabs/databases/MAINFRAME_ACCESS pulled, 0 skipped. 0.2 MB/s (16384 bytes in 0.063s)  
nguyenvanhao@ubuntu:~/Downloads/Challenges/EVABsv5$
```

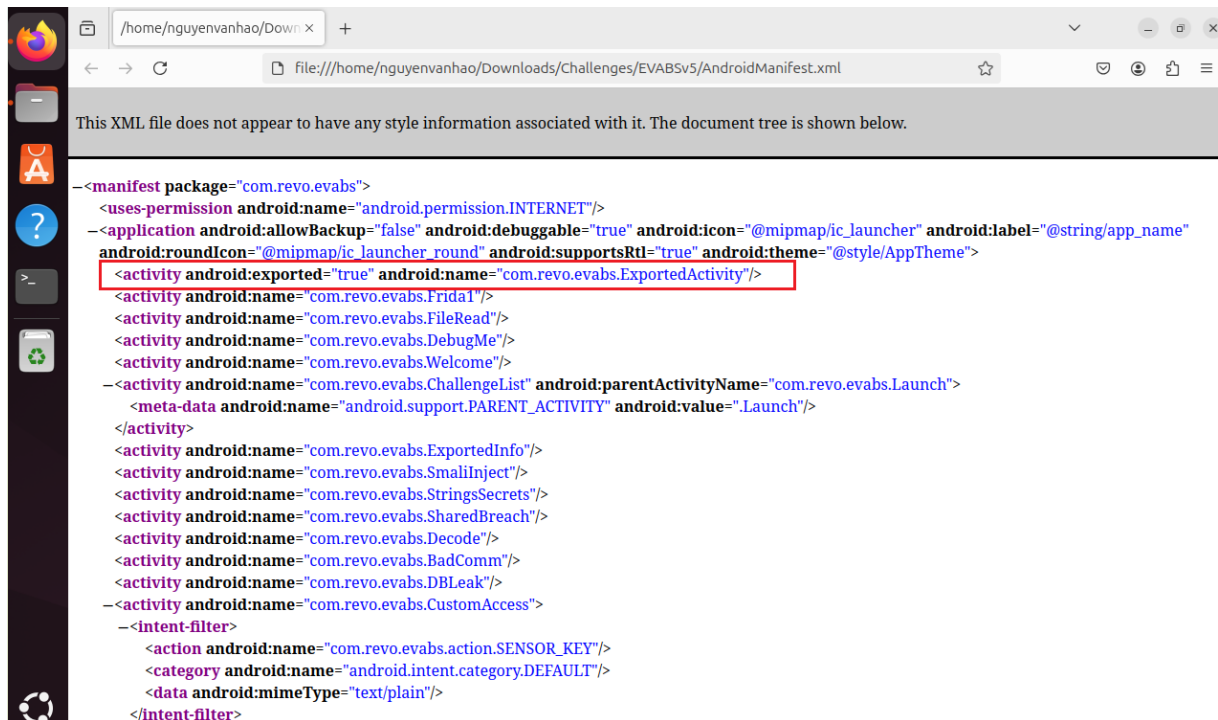
- Sử dụng `sqlite3` để xem database thì ta thấy được flag như bên dưới:

```
nguyenvanhao@ubuntu: ~/Downloads/Challenges/EVABsv5  
nguyenvanhao@ubuntu:~/Downloads/Challenges/EVABsv5$ sqlite3 MAINFRAME_ACCESS  
SQLite version 3.45.1 2024-01-30 16:01:20  
Enter ".help" for usage hints.  
sqlite> .databases  
main: /home/nguyenvanhao/Downloads/Challenges/EVABsv5/MAINFRAME_ACCESS r/w  
sqlite> .table  
CREDS          android_metadata  
sqlite> select * from CREDS  
...> ;  
Dr.l33t|EVABS{sqlite_is_not_safe}E|ADMIN  
Mr BufferOverflow|0xNotSecureSQLite_|STAFF  
Ms HeapSpray|SQLite_exploit|USER  
sqlite>
```

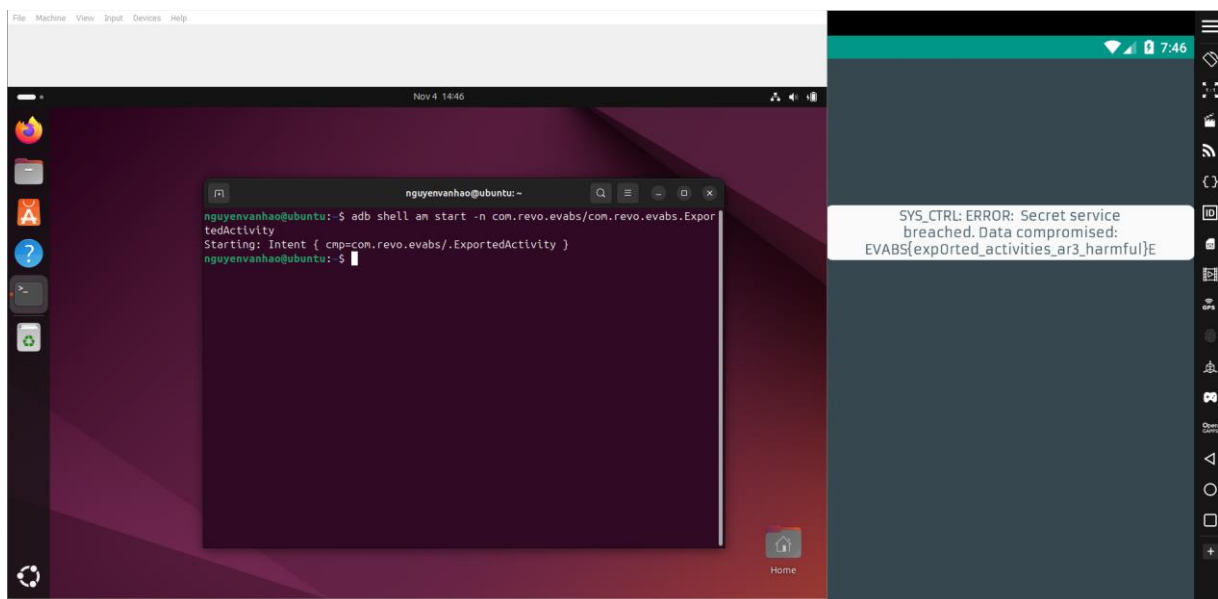
- Kết quả Flag: **EVABS{sqlite_is_not_safe}**

Challenges 7: Export

- Khi kiểm tra AndroidManifest.xml thì ta thấy có thông tin của một Activity bị exported:



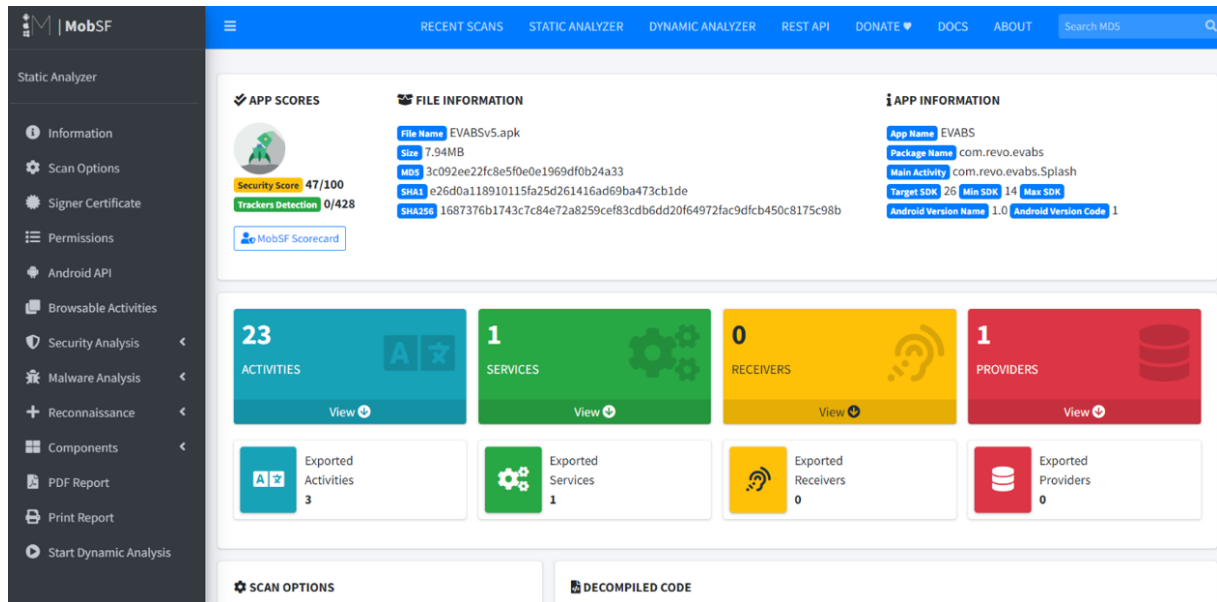
- Khi một activity bị exported. Ta sử dụng lệnh: **adb shell am start -n com.revo.evabs/com.revo.evabs.ExportedActivity** để trigger các exported activity và thu được kết quả bên dưới hình:



- Kết quả Flag: **EVABS{exp0rted_activities_ar3_harmful}**

Challenges 8: Decode

- Ta lên MobSF sau đó decompile code và tải code java về.



- Mở File **Decode.class** thì thấy 3 đoạn string ở dạng Base64:

```
1 package com.revo.evabs;
2
3 import android.os.Bundle;
4 import android.support.v7.app.AppCompatActivity;
5 import android.util.Base64;
6 import android.util.Log;
7 import android.widget.Button;
8 import android.widget.TextView;
9
10 public class Decode extends AppCompatActivity {
11     protected void onCreate(Bundle var1) {
12         super.onCreate(var1);
13         this setContentView(2131492896);
14         StringBuilder var2 = new StringBuilder();
15         var2.append("RVZBQIN7bmV2M3Jfc3QwcmU=");
16         var2.append("X3MzbnMhdG12M19kYXRh");
17         var2.append("XzFuXzdoM19zMHV2Y2VjMGRl");
18         var2.toString();
19         ((Button)this.findViewById(2131361842)).setOnClickListener(new 1(this, (TextView) this.findViewById(2131362094)));
20     }
21 }
22
```

- Ta tiến hành Decode Base64 online <https://www.base64decode.org/> với 3 đoạn string trên:

+ Đoạn string thứ nhất: **RVZBQIN7bmV2M3Jfc3QwcmU=**

Decode from Base64 format


Simply enter your data then push the decode button.



```
RVZBQIN7bmV2M3Jfc3QwcmU=
```

 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.

```
EVABS{nev3r_st0re
```


- Ta thu được: **EVABS{nev3r_st0re**

+ Đoạn string thứ hai: **X3MzbmMhdGL2M19kYXRh**

Decode from Base64 format


Simply enter your data then push the decode button.

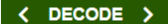

```
X3MzbmMhdGL2M19kYXRh
```

 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.

```
_s3ncltiv3_data
```


- Ta thu được: **_s3nc!tiv3_data**

+ Đoạn string thứ ba: **XzFuXzdoM19zMHVyY2VjMGRI**

Decode from Base64 format


Simply enter your data then push the decode button.



XzFuXzdoM19zMHVyY2VjMGRI

 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).

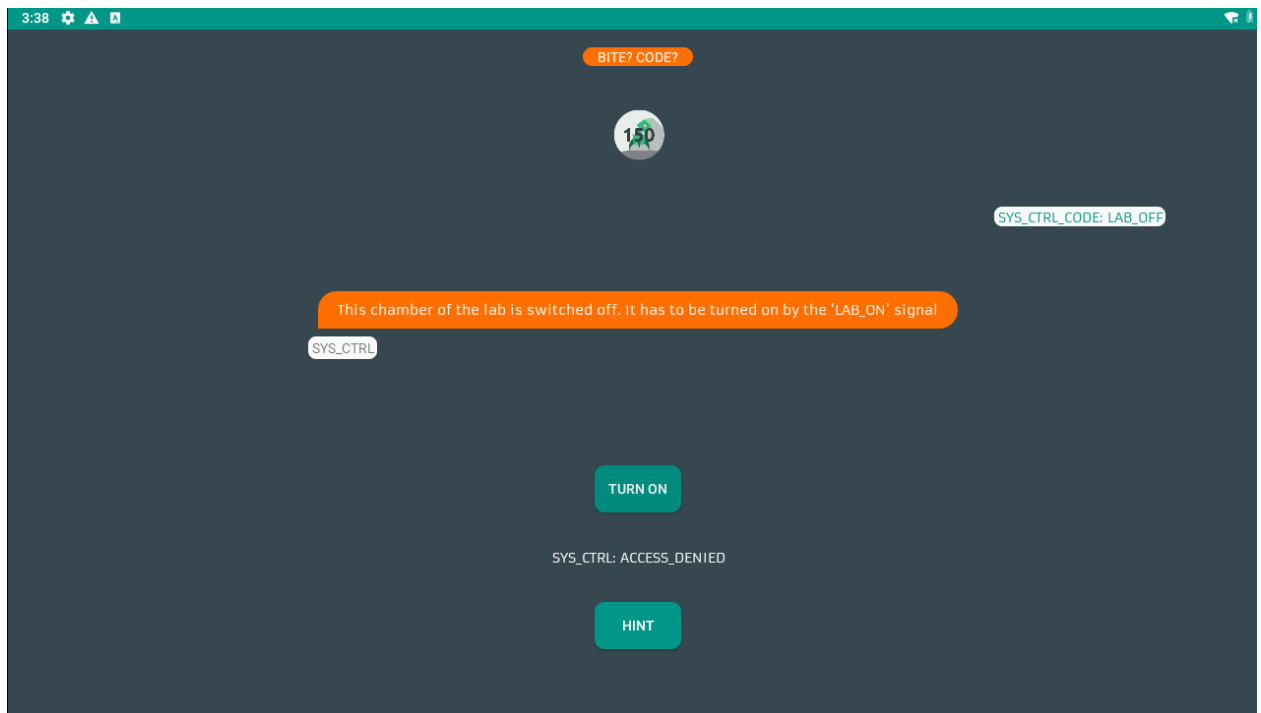
 **DECODE**  Decodes your data into the area below.

_1n_7h3_s0urcec0de

- Ta thu được: **_1n_7h3_s0urcec0de**

- Ghép 3 đoạn lại, kết quả Flag:
EVABS{nev3r_st0re _s3nc!tiv3_data _1n_7h3_s0urcec0de)

Challenges 9: Smali Injection



- Sử dụng jadx-gui để xem code java từ dịch ngược của thử thách.

```

SmaliInject$2 x  /home/ubuntu1/Desktop/lab4/Challenges/EVABSV5/smali/com/revo/evabs/SmaliInject$2.smali x
package com.revo.evabs;

import android.view.View;
import android.widget.TextView;

/* loaded from: /tmp/jadx-8061233653773349844.dex */
32 class SmaliInject$2 implements View.OnClickListener {
    final /* synthetic */ SmaliInject this$0;
    final /* synthetic */ TextView val$labstat;
    final /* synthetic */ TextView val$tvflag;
    final /* synthetic */ TextView val$tvlaboff;

33     SmaliInject$2(SmaliInject this$0, TextView textView, TextView textView2, TextView textView3) {
34         this.this$0 = this$0;
        this.val$tvlaboff = textView;
        this.val$labstat = textView2;
        this.val$tvflag = textView3;
    }

    @Override // android.view.View.OnClickListener
    37 public void onClick(View view) {
    38     String ctrl = this.this$0.stringFromSmali();
    40     if (this.this$0.SIGNAL.equals("LAB_ON")) {
    41         this.val$tvlaboff.setText("SYS_CTRL_CODE: LAB_ON");
    42         this.val$labstat.setText("SYS_CTRL: ACCESS_GRANTED. LAB UNLOCKED");
    43         this.val$tvflag.setText("EVABS(" + ctrl + ")");
    49         return;
    }
    46     this.val$tvlaboff.setText("SYS_CTRL_CODE: LAB_OFF");
    47     this.val$labstat.setText("SYS_CTRL: ACCESS_DENIED");
    }
}

```

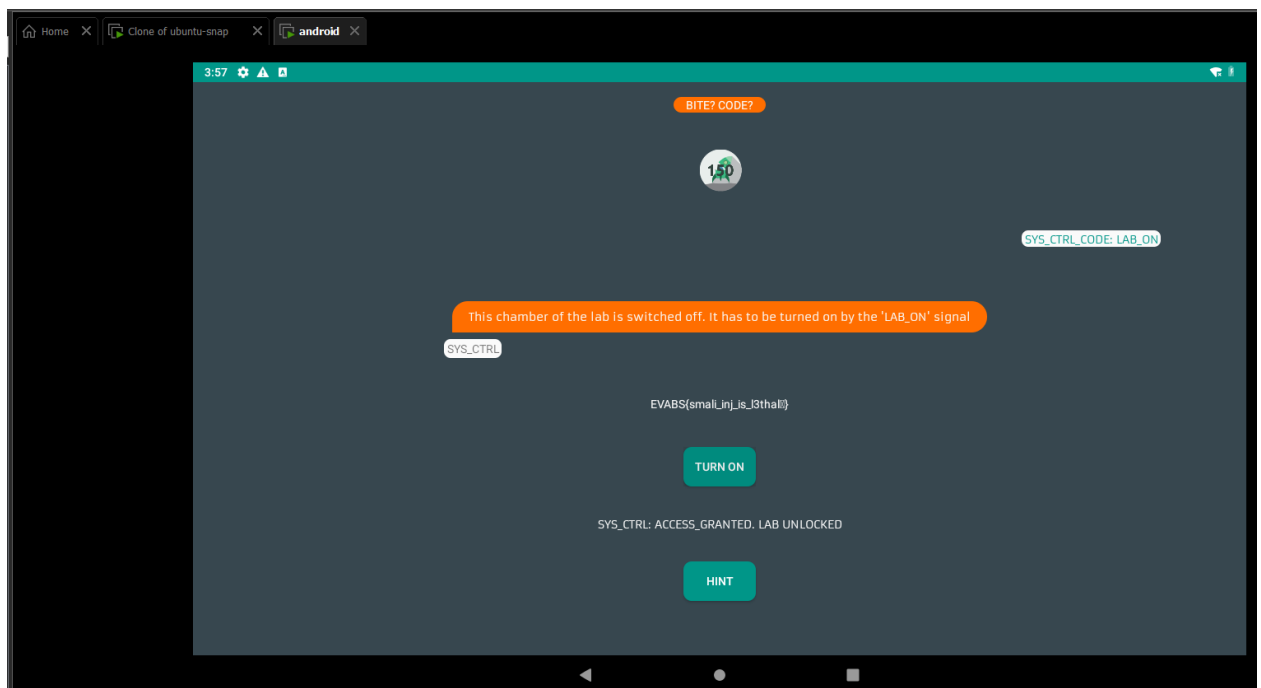
- Từ kết quả in ra của ứng dụng ta biết giá trị của biến SIGNAL khi nhấn nút TURN ON khác LAB_ON nên ứng dụng thực hiện in ra ACCESS_DENIED. Ta dự đoán giá trị của SIGNAL khi đó là LAB_OFF, để kiểm tra ta thực hiện sửa giá trị LAB_ON trong phương thức onClick thành LAB_OFF.

```

50 # virtual methods
51 .method public onClick(Landroid/view/View;)V
52   .locals 4
53   .param p1, "view"    # Landroid/view/View;
54
55   .line 38
56   iget-object v0, p0, Lcom/revo/evabs/SmaliInject$2; ->this$0:Lcom/revo/evabs/SmaliInject;
57
58   invoke-virtual {v0}, Lcom/revo/evabs/SmaliInject; ->stringFromSmali()Ljava/lang/String;
59
60   move-result-object v0
61
62   .line 40
63   .local v0, "ctrl":Ljava/lang/String;
64   iget-object v1, p0, Lcom/revo/evabs/SmaliInject$2; ->this$0:Lcom/revo/evabs/SmaliInject;
65
66   iget-object v1, v1, Lcom/revo/evabs/SmaliInject; ->SIGNAL:Ljava/lang/String;
67
68   const-string v2, "LAB_OFF"
69
70   invoke-virtual {v1, v2}, Ljava/lang/String; ->equals(Ljava/lang/Object;)Z
71
72   move-result v1
73
74   if-eqz v1, :cond_0

```

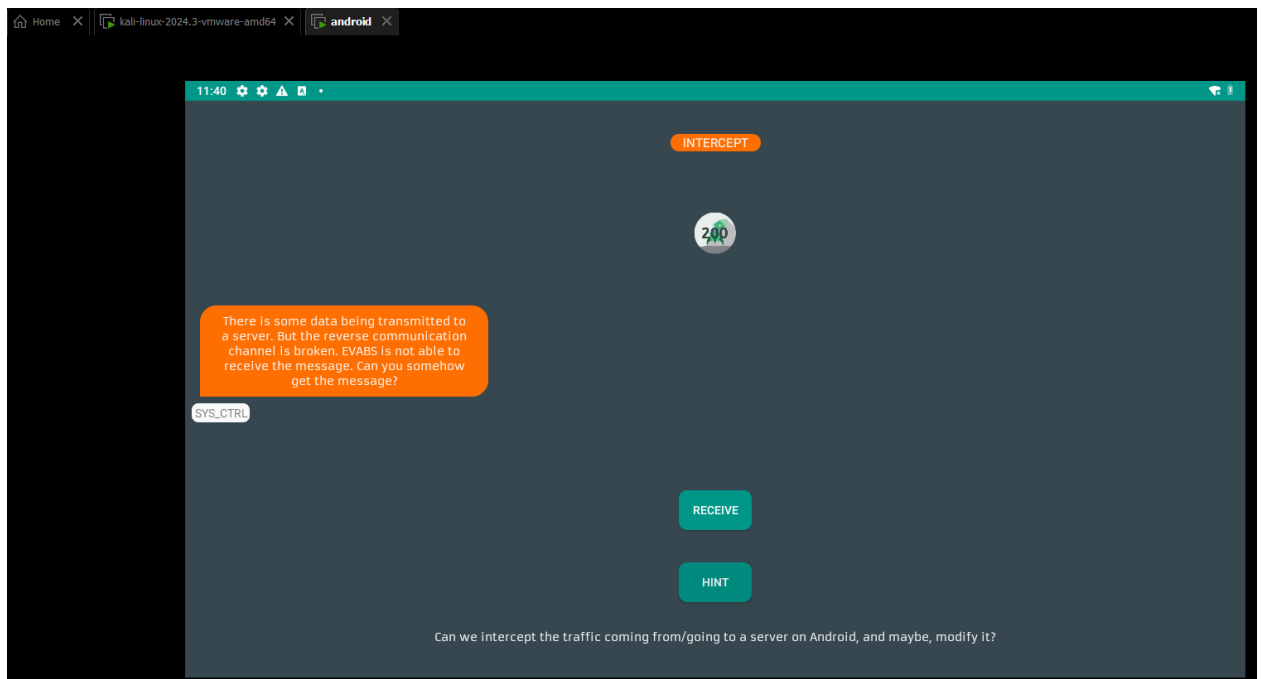
- Thực hiện patch, kí và cài đặt ứng dụng lại, ta có được flag:



- Kết quả Flag: **EVABS{smali_inj_is_l3thals}**

Challenges 10: Interception

- Ứng dụng gợi ý ta sử dụng proxy để chặn gói request gửi đi từ ứng dụng



- Thực hiện cài đặt burpsuite làm server proxy ta bắt được request sau của ứng dụng:

#	Host	Method	URL	Params	Edited	Status code
156	https://www.neonsec.com	POST	/evabs/reboot.php	✓		
157	https://firebaseinstallations.go...	POST	/v1/projects/android.com:api-project-...			
158	https://firebaseinstallations.go...	POST	/v1/projects/android.com:api-project-...			
159	https://android.apis.google.com	POST	/c2dm/register3	✓		200
160	https://firebaseinstallations.go...	POST	/v1/projects/android.com:api-project-...			
161	https://firebaseinstallations.go...	POST	/v1/projects/android.com:api-project-...			
162	http://www.google.com	GET	/gen_204			
163	http://connectivitycheck.gstatic....	GET	/generate_204			
164	https://www.google.com	GET	/generate_204			
165	https://www.neonsec.com	POST	/evabs/reboot.php	✓		

Request

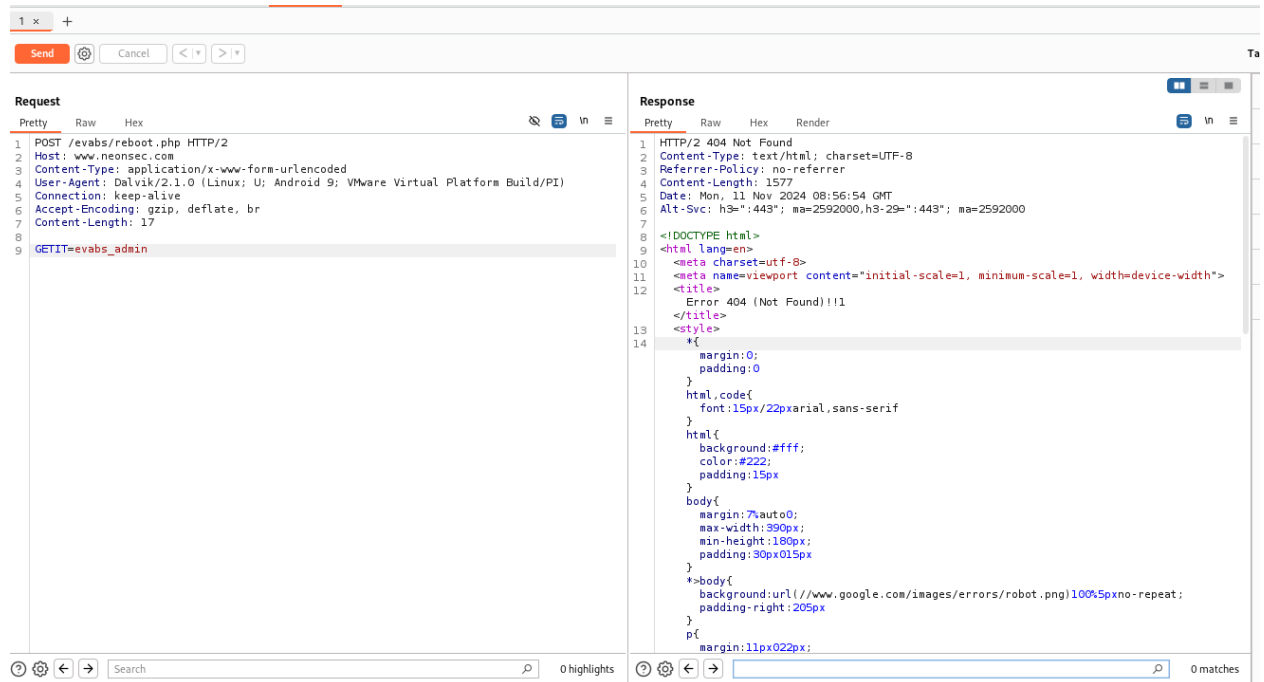
Pretty Raw Hex

```

1 POST /evabs/reboot.php HTTP/1.1
2 Content-Type: application/x-www-form-urlencoded
3 User-Agent: Dalvik/2.1.0 (Linux; U; Android 9; VMware Virtual Platform Build/PI)
4 Host: www.neonsec.com
5 Connection: keep-alive
6 Accept-Encoding: gzip, deflate, br
7 Content-Length: 17
8
9 GETIT=evabs_admin

```

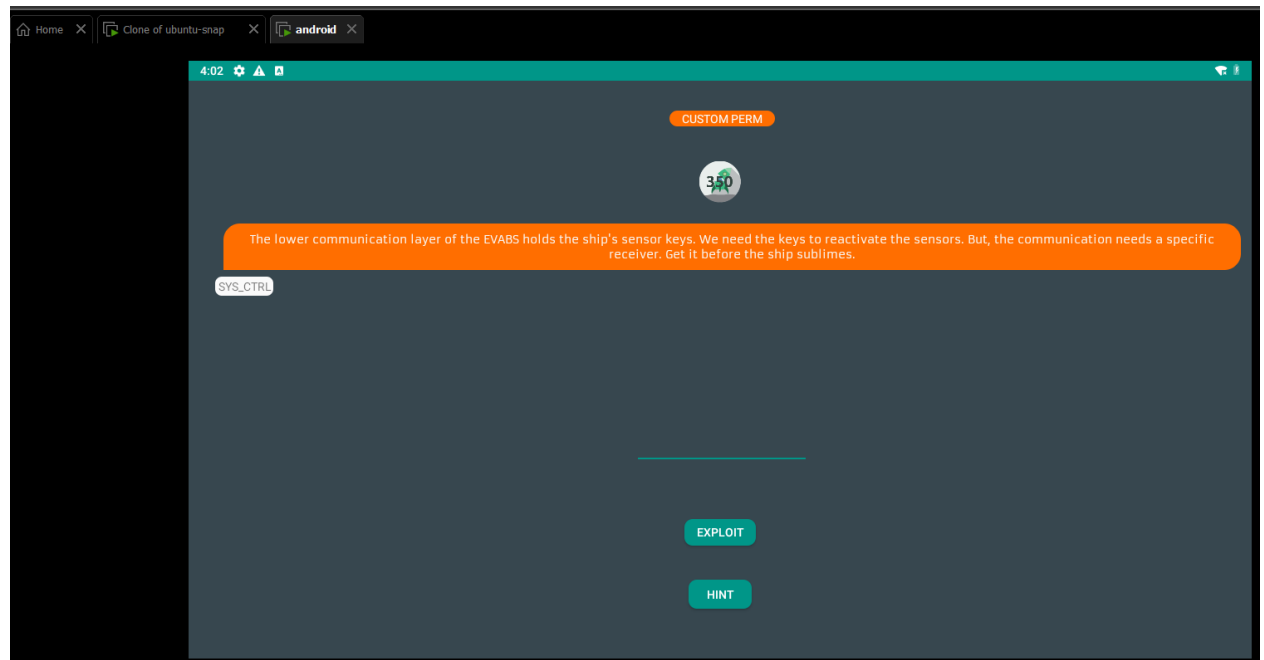
- Thực hiện chuyển gói qua repeater để gửi và nhận phản hồi của server:



- Không thể tìm thấy flag do lỗi.

Challenges 11: Custom Access

- Từ mô tả của bài lab có thể thấy sensor key là flag ta cần tìm



- Từ mã java dịch ngược ta có thể thấy input cần nhập vào là cust0m_p3rm. Nếu người dùng nhập đúng mật khẩu thì chuỗi flag "EVABS{" + stringFromJNI() + "}" sẽ được đưa vào intent bằng hàm putExtra.

```
CustomAccess x
import android.widget.Toast;

/* loaded from: /tmp/jadx-7333543432981404064.dex */
10 public class CustomAccess extends AppCompatActivity {
    public final String EVABS_SENSOR_KEY = "com.revo.evabs.action.SENSOR_KEY";

    public native String stringFromJNI();

    16 protected void onCreate(Bundle savedInstanceState) {
    17     super.onCreate(savedInstanceState);
    18     setContentView(2131492893);
    20     Button btncustomaccess = (Button) findViewById(2131361835);
    21     btncustomaccess.setOnClickListener(new 1(this));
    28     TextView tvhintcust = (TextView) findViewById(2131362091);
    29     Button hintcustom = (Button) findViewById(2131361841);
    30     hintcustom.setOnClickListener(new 2(this, tvhintcust));
    }

    /* JADX INFO: Access modifiers changed from: private */
    /* JADX WARN: Multi-variable type inference failed */
    38 public void GetSensorKey() {
    39     EditText et = (EditText) findViewById(2131361891);
    40     String tosplit = et.getText().toString();
    41     char[] split = {'c', 'u', 's', 't', 'o', 'm', '_', 'p', '3', 'r', 'm'};
    42     String fromsplit = new String(split);
    44     if (fromsplit.equals(tosplit)) {
    45         Toast.makeText((Context) this, (CharSequence) "SYS_CTRL: CREDENTIALS ACCEPTED. SENSOR_KEY SENT", 1).show();
    46         Intent sendSensorkey = new Intent("com.revo.evabs.action.SENSOR_KEY");
    48         sendSensorkey.putExtra("android.intent.extra.TEXT", "EVABS{" + stringFromJNI() + "}");
    49         sendSensorkey.setType("text/plain");
    50         startActivity(sendSensorkey);
    51         return;
    52     }
    53     Toast.makeText((Context) this, (CharSequence) "SYS_CTRL: WRONG_CREDENTIALS. SENSOR_KEY LOCKED", 1).show();
    }

    static {
    59     System.loadLibrary("native-lib");
    }
}
```

- Sử dụng frida hook để can thiệp và lấy flag từ intent sử dụng đoạn script sau:

```
hook.py
~/Desktop/lab4/Challenges/EVABSV5
Save

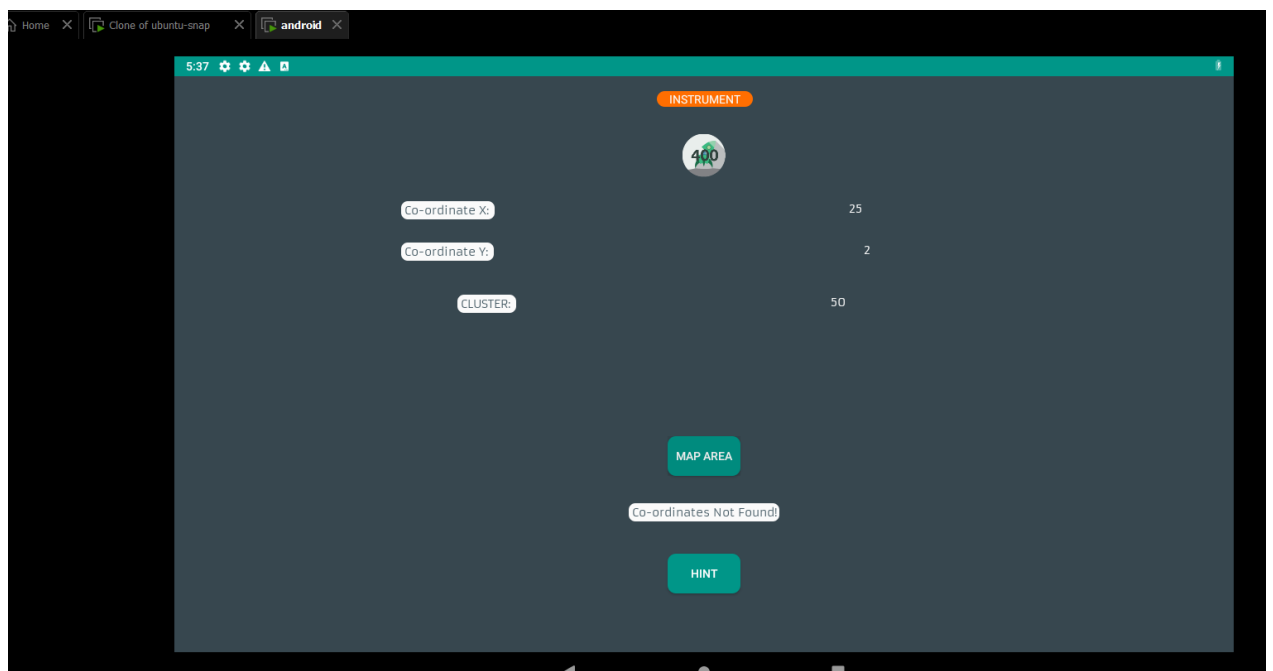
1 import frida
2 import sys
3
4 def onMessage(message, data):
5     print(message)
6
7 package = "EVABS" # chương trình cần hook
8
9 jscode = """
10 Java.perform(function () {
11     send("[*] Starting hooks android.content.Intent.putExtra");
12     var intent = Java.use("android.content.Intent");
13     intent.putExtra.overload("java.lang.String", "java.lang.String").implementation = function(var_1, var_2) {
14         send("[+] Flag: " + var_2);
15     };
16 });
17 """
18
19 try:
20     device = frida.get_device_manager().add_remote_device("192.168.64.128:27042") # địa chỉ frida server
21     process = device.attach(package)
22
23     script = process.create_script(jscode)
24     script.on("message", onMessage)
25     print("[*] Hooking", package)
26     script.load()
27     sys.stdin.read() # giữ script luôn chạy
28 except frida.TransportError as e:
29     print(f"Không thể kết nối với Frida server: {e}")
30 except frida.ProcessNotFoundError as e:
31     print(f"Không tìm thấy tiến trình {package}: {e}")
32 except Exception as e:
33     print(f"Lỗi không xác định: {e}")
34
```

- Chạy đoạn scrip, sau đó vào ứng dụng và nhập password cus0m_p3rm.
- Kết quả hook thành công và ta thu được flag:

```
ubuntu1@ubuntu1-virtual-machine:~/Desktop/lab4/Challenges/EVABSV5$ python3 hook.py
[*] Hooking EVABS
{'type': 'send', 'payload': '[-] Starting hooks android.content.Intent.putExtra'}
{'type': 'send', 'payload': '[+] Flag: EVABS{always_verify_packag3s}'}
```

- Kết quả Flag: **EVABS{always_verify_packag3sa}**

Challenges 12: Intrument



- Từ đoạn code java từ dịch ngược có thể thấy string x (flag) chỉ được trả về khi $a * b > r.nextint(70) + 150$. Hàm nextint sẽ trả về một số ngẫu nhiên từ 0 đến 70, đồng thời a và b đã được gán cố định nên biểu thức sẽ không bao giờ thỏa mãn trừ khi ta can thiệp.

```

Fridal x
/* loaded from: /tmp/jadx-13258230780716461141.dex */
16 public class Fridal extends AppCompatActivity implements View.OnClickListener {
    int a = 25;
    int b = 2;
    int x;

    public native String stringFromJNI();

17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(2131492901);
20         Button bt = (Button) findViewById(2131361902);
21         bt.setOnClickListener(this);
23         Button btnhint = (Button) findViewById(2131361844);
24         TextView tvhint = (TextView) findViewById(2131362093);
25         btnhint.setOnClickListener(new 1(this, tvhint));
    }

    @Override // android.view.View.OnClickListener
35     public void onClick(View view) {
36         TextView tv = (TextView) findViewById(2131361996);
37         TextView at = (TextView) findViewById(2131362132);
38         TextView bt = (TextView) findViewById(2131362134);
39         TextView xt = (TextView) findViewById(2131362142);
41         at.setText(String.valueOf(this.a));
42         bt.setText(String.valueOf(this.b));
44         this.x = this.a * this.b;
45         Random r = new Random();
46         int rand = r.nextInt(70) + 150;
48         xt.setText(String.valueOf(this.x));
50         if (this.x > rand) {
51             tv.setText("VIBRAN IS RESDY TO FLY! YOU ARE GOING HOME!");
52             String x = stringFromJNI();
53             Log.d("CONGRATZ!", x);
59             return;
        }
56         tv.setText("Co-ordinates Not Found!");
    }

    static {

```

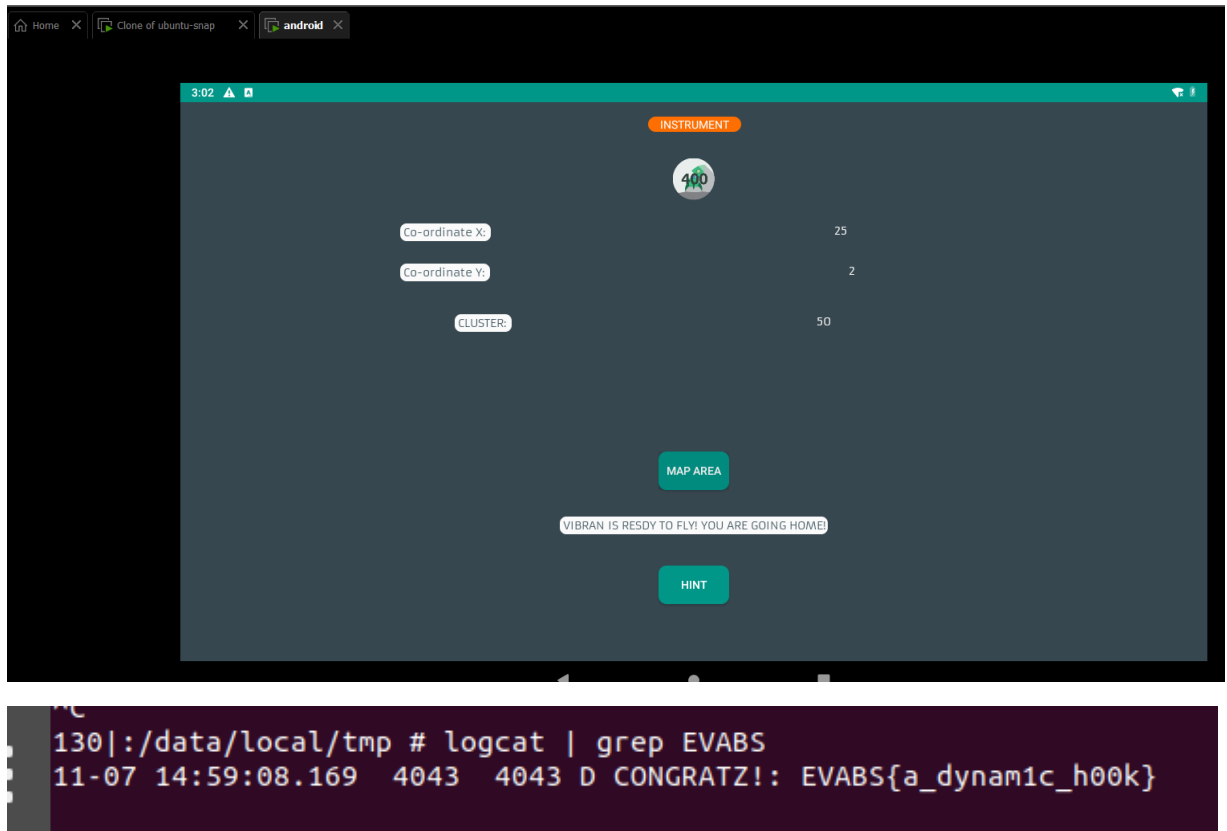
- Ta sử dụng script sau để hook vào hàm nexint và khiến hàm trả về -150 dẫn đến biểu thức $a*b > 0$ luôn đúng.

```

hook12.py
1 import frida
2 import sys
3
4 def onMessage(message, data):
5     print(message)
6
7 package = "EVABS"
8
9 jscode = """
10 Java.perform(function () {
11     send("[-] Starting hooks java.util.Random.nextInt");
12     var random = Java.use("java.util.Random");
13     random.nextInt.overload("int").implementation = function(var_1) {
14         return -150;
15     };
16
17 });
18 """
19
20 device = frida.get_device_manager().add_remote_device("192.168.64.128:27042")
21 process = device.attach(package)
22
23 script = process.create_script(jscode)
24 script.on("message", onMessage)
25 print("[*] Hooking", package)
26 script.load()
27 sys.stdin.read()
28

```


- Ta có được Flag:

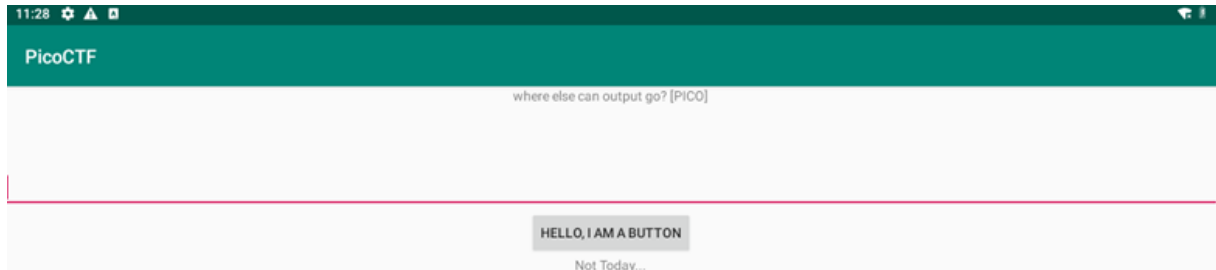


- Kết quả Flag: **EVABS{a_dynamic_h00k}**

D.2 Droid

Challenges 1: Nhật ký droid đã đi đâu. Bạn có thể tìm thấy tại: **one.apk**

- Thực hiện cài đặt one.apk và truy cập vào ứng dụng



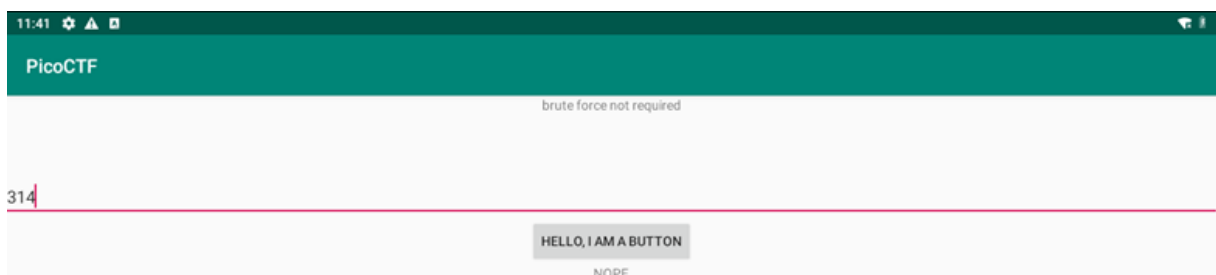
- Từ kết quả thấy được và mô tả của thử thách ta thực hiện kiểm tra log của thiết bị sử dụng **logcat | grep "pico"** và thu được flag:

```
11-03 23:08:44.006 4781 4793 I ellocmu.picoctf: Background concurrent copying GC freed 6111(3MB) AllocSpace objects, 0(0B) LOS objects, 0.226ms
11-03 23:08:44.952 2054 2077 I ActivityManager: Displayed com.helloctmu.picoctf/.MainActivity: +4s212ms
11-03 23:08:45.357 1923 1990 W SurfaceFlinger: Attempting to set client state on removed layer: Splash Screen com.helloctmu.picoctf#0
11-03 23:08:45.357 1923 1990 W SurfaceFlinger: Attempting to destroy on removed layer: Splash Screen com.helloctmu.picoctf#0
11-03 23:08:49.732 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
11-03 23:08:57.925 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
11-03 23:09:07.370 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
11-03 23:09:19.830 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
11-03 23:09:21.586 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
11-03 23:10:11.381 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
11-03 23:28:54.717 4781 4781 I PICO : picoCTF{a.moose.once.bit.my.sister}
```

- Kết quả Flag: **picoCTF{a.moose.once.bit.my.sister}**

Challenges 2: Tìm kiếm và lấy flag. Bạn có thể tìm thấy tại: **two.apk**.

- Thực hiện cài đặt và chạy two.apk



- Thực hiện decompile ứng dụng bằng **apk d two.apk**

```
ubuntu1@ubuntu1-virtual-machine: ~/Desktop/lab4/Challenges$ apktool d two.apk
I: Using Apktool 2.10.0 on two.apk with 2 thread(s).
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file: /home/ubuntu1/.local/share/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...
```

- Thực hiện tìm kiếm cụm từ **NOPE** hiển thị trên màn hình bằng **grep -r "NOPE"**

```
ubuntu1@ubuntu1-virtual-machine: ~/Desktop/lab4/Challenges/two$ grep -r "NOPE"
smali/com/helloctmu/picoctf/FlagstaffHill.smali:    const-string v1, "NOPE"
ubuntu1@ubuntu1-virtual-machine: ~/Desktop/lab4/Challenges/two$
```

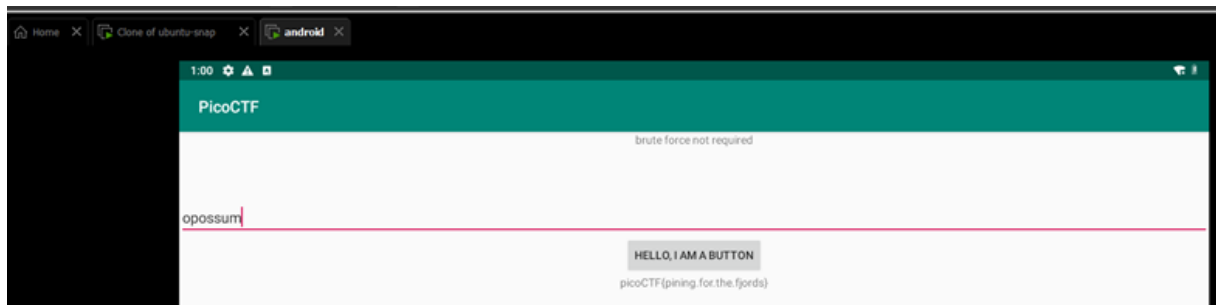
- Mở file **FlagstaffHill.smali** ta thấy ứng dụng thực hiện so sánh chuỗi **input** với biến **password**, nếu đúng sẽ thực hiện gọi hàm **fenugreek** ngược lại in ra **NOPE**. Hàm **fenugreek** nhận **input** làm tham số, có thể flag đã được mã hóa và dùng chuỗi input để giải mã do ta không thể thực hiện grep flag được.

```
31    .line 12
32    .local v0, "password":Ljava/lang/String;
33    invoke-virtual {p0, v0}, Ljava/lang/String; ->equals(Ljava/lang/Object;)Z
34
35    move-result v1
36
37    if-eqz v1, :cond_0
38
39    invoke-static {p0}, Lcom/helloctmu/picoctf/FlagstaffHill; ->fenugreek(Ljava/lang/String;)Ljava/lang/String;
40
41    move-result-object v1
42
43    return-object v1
44
45    .line 13
46    :cond_0
47    const-string v1, "NOPE"
48
49    return-object v1
50 .end method
```

- Thực hiện tìm kiếm password bằng lệnh **grep -ri "password"**. Ta tìm thấy giá trị password là **opossum**

```
ubuntu1@ubuntu1-virtual-machine: ~/Desktop/lab4/Challenges/two$ grep -ri "password"
res/values/strings.xml:    <string name="password">opossum</string>
res/values/public.xml:    <public type="string" name="password" id="0x7f0b002f" />
smali/androidx/customview/widget/ExploreByTouchHelper.smali:    invoke-virtual {v1}, Landroidx/core/view/accessibility/AccessibilityNodeInfoCompat; ->isPassword()Z
smali/androidx/core/widget/TextViewCompat.smali:    instance-of v0, v0, Landroid/text/method/PasswordTransformationMethod;
smali/androidx/core/widget/TextViewCompat.smali:    invoke-virtual {v0}, Landroid/text/method/PasswordTransformationMethod; ->isPassword()Z
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:    invoke-virtual {v0}, Landroid/view/accessibility/AccessibilityNodeInfo; ->isPassword()Z
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:    .method public setPassword(Z)V
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:        .param p1, "password" # Z
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:        invoke-virtual {v0, p1}, Landroid/view/accessibility/AccessibilityNodeInfo; ->setPassword(Z)V
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:        const-string v2, "; password: "
smali/androidx/core/view/accessibility/AccessibilityNodeInfoCompat.smali:        invoke-virtual {p0}, Landroidx/core/view/accessibility/AccessibilityNodeInfoCompat; ->isPassword()Z
smali/androidx/core/view/accessibility/AccessibilityRecordCompat.smali:    invoke-virtual {v0}, Landroid/view/accessibility/AccessibilityRecord; ->isPassword()Z
smali/androidx/core/view/accessibility/AccessibilityRecordCompat.smali:    .method public setPassword(Z)V
smali/androidx/core/view/accessibility/AccessibilityRecordCompat.smali:        .param p1, "password" # Z
smali/androidx/core/view/accessibility/AccessibilityRecordCompat.smali:        invoke-virtual {v0, p1}, Landroid/view/accessibility/AccessibilityRecord; ->setPassword(Z)V
smali/androidx/appcompat/widget/AppCompatTextHelper.smali:    instance-of v9, v9, Landroid/text/method/PasswordTransformationMethod;
smali/com/helloctmu/picoctf/FlagstaffHill.smali:    .local v0, "password":Ljava/lang/String;
smali/com/helloctmu/picoctf/FlagstaffHill.smali:    .field public static final password:I = 0x7f0b002f
ubuntu1@ubuntu1-virtual-machine: ~/Desktop/lab4/Challenges/two$
```

- Nhập chuỗi vào ứng dụng và thu được flag:



- Kết quả Flag: **picoCTF{pining.for.the.fjords}**

Challenges 3: Tìm kiếm và lấy flag. Bạn có thể tìm thấy tại: three.apk.

- Thực hiện tương tự như bt2, ta tìm thấy cụm từ NOPE trong FlagstaffHill.smali. Thực hiện mở file bằng công cụ jadx-gui, ta thấy được mã java thông qua dịch ngược.



- Copy và viết lại chương trình java để in kết quả ra màn hình:

```

ubuntu1@ubuntu1-virtual-machine:~/Desktop$ cat FlagstaffHill.java
public class FlagstaffHill {
    public static void main(String[] args) {
        String[] witches = {"weatherwax", "ogg", "garlick", "nitt", "aching", "dismiss"};

        int second = 3 - 3;
        int third = (3 / 3) + second;
        int fourth = (third + third) - second;
        int fifth = 3 + fourth;
        int sixth = (fifth + second) - third;

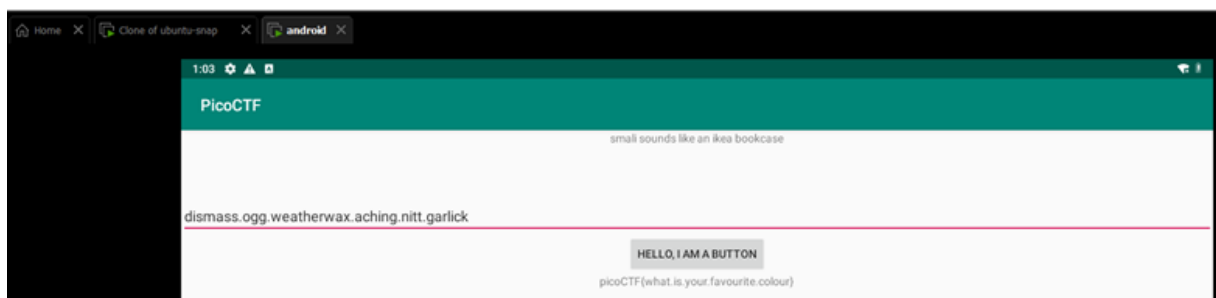
        String password = "".concat(witches[fifth])
                           .concat(".").concat(witches[third])
                           .concat(".").concat(witches[second])
                           .concat(".").concat(witches[sixth])
                           .concat(".").concat(witches[3])
                           .concat(".").concat(witches[fourth]);

        System.out.println("Generated password: " + password);
    }
}

ubuntu1@ubuntu1-virtual-machine:~/Desktop$ javac FlagstaffHill.java
ubuntu1@ubuntu1-virtual-machine:~/Desktop$ java FlagstaffHill
Generated password: dismiss.ogg.weatherwax.aching.nitt.garlick
ubuntu1@ubuntu1-virtual-machine:~/Desktop$

```

- Nhập mật khẩu tính được vào ứng dụng và ta thu được flag bên dưới:

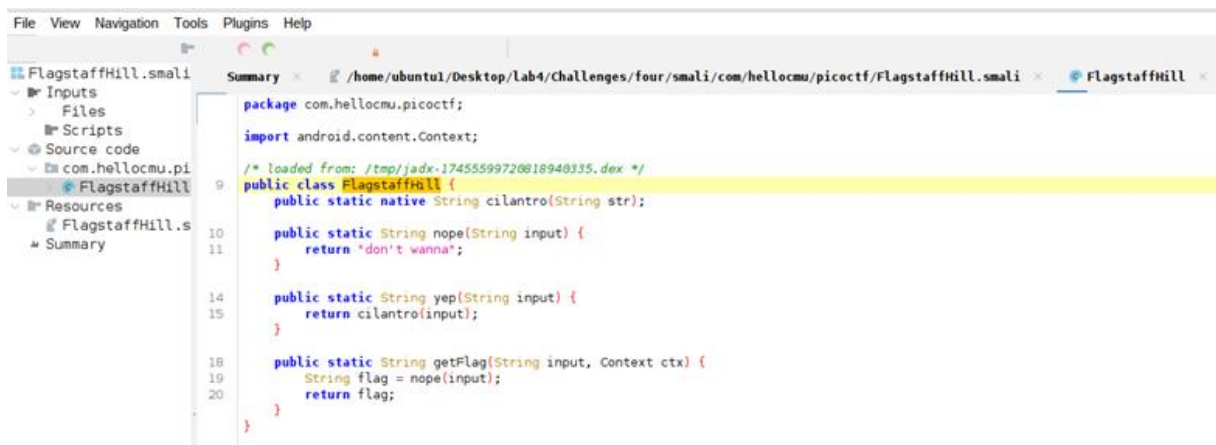


- Kết quả Flag: **picoCTF{what.is.your.favourite.colour}**

Challenges 4: Dịch ngược, vá lại tập tin và lấy cờ. Bạn có thể tìm thấy tại: **four.apk**.

- Thực hiện tương tự bài 3.

- Sử dụng jadx-gui để xem mã java từ dịch ngược. Có thể thấy phương thức **getflag** sẽ trả về cờ cho người dùng, phương thức hiện đang gọi hàm **nope**, ta sẽ thử đổi **nope** thành **yep** để in ra flag.



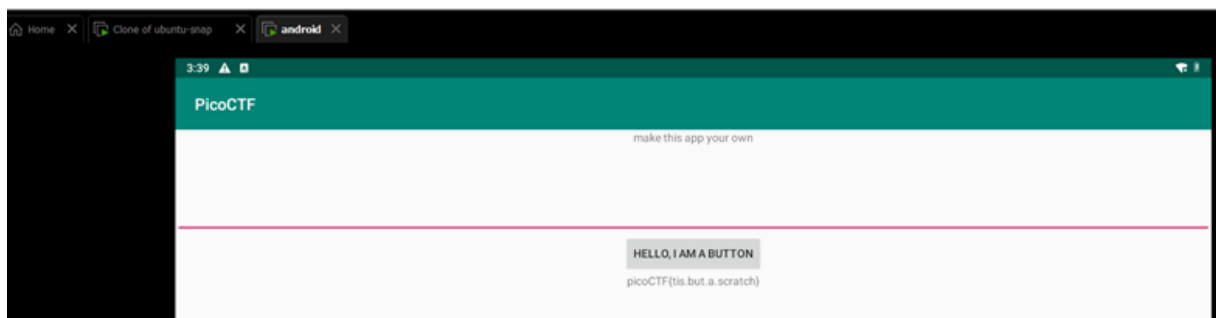
- Thực hiện thay đổi code smali trong file **flagstaffhill.smali** từ nope thành yep.

```

24 .line 19
25 invoke-static {p0}, Lcom/helloctmu/picocftf/FlagstaffHill;->yep(Ljava/lang/String;):Ljava/lang/String;
26
27 move-result-object v0

```

- Thực hiện patch ứng dụng và kí, gỡ bỏ ứng dụng cũ và cài ứng dụng đã patch, ta thu được flag:



- Kết quả Flag: **picoCTF{tis.but.a.scratch}**

Challenges 5: Dịch ngược, vá lại tập tin và lấy cờ. Bạn có thể tìm thấy tại: **five.apk**.

- Thực hiện tương tự bài tập 4. Từ đoạn mã java có được nhờ dịch ngược ta có thể thấy hàm **getflag** không trả về giá trị flag mà trả về **call it** hoặc **nope**. Ngoài ra lớp còn có hàm **cardamom** không được gọi đến và nhận một chuỗi input đầu vào. Tên giá trị trả về **call it** gợi ý cho ta gọi hàm **cardamom** tại vị trí đó sẽ trả về flag.

```

FlagstaffHill x Summary x /home/ubuntu1/Desktop/lab4/Challenges/five/smali/com/helloemu/picocftf/FlagstaffHill.smali x
package com.helloemu.picocftf;

import android.content.Context;

/* loaded from: /tmp/jadx-15728540262272452306.dex */
10 public class FlagstaffHill {
    public static native String cardamom(String str);

    public static String getFlag(String input, Context ctx) {
11         StringBuilder ace = new StringBuilder("aaa");
12         StringBuilder jack = new StringBuilder("aaa");
13         StringBuilder queen = new StringBuilder("aaa");
14         StringBuilder king = new StringBuilder("aaa");
15         ace.setCharAt(0, (char) (ace.charAt(0) + 4));
16         ace.setCharAt(1, (char) (ace.charAt(1) + 19));
17         ace.setCharAt(2, (char) (ace.charAt(2) + 18));
18         jack.setCharAt(0, (char) (jack.charAt(0) + 7));
19         jack.setCharAt(1, (char) (jack.charAt(1) + 0));
20         jack.setCharAt(2, (char) (jack.charAt(2) + 1));
21         queen.setCharAt(0, (char) (queen.charAt(0) + 0));
22         queen.setCharAt(1, (char) (queen.charAt(1) + 11));
23         queen.setCharAt(2, (char) (queen.charAt(2) + 15));
24         king.setCharAt(0, (char) (king.charAt(0) + 14));
25         king.setCharAt(1, (char) (king.charAt(1) + 20));
26         king.setCharAt(2, (char) (king.charAt(2) + 15));
27         String password = "".concat(queen.toString()).concat(jack.toString()).concat(ace.toString()).concat(king.toString());
28         return input.equals(password) ? "call it" : "NOPE";
29     }
30 }

```

- Ta cần tìm giá trị của password để hàm trả về **call it**. Copy và sửa lại đoạn mã java để in ra giá trị của password

```

Open [icon] FlagstaffHill.java ~/Desktop
1 public class FlagstaffHill {
2     public static void main(String[] args) {
3         // Calculate the password string
4         StringBuilder ace = new StringBuilder("aaa");
5         StringBuilder jack = new StringBuilder("aaa");
6         StringBuilder queen = new StringBuilder("aaa");
7         StringBuilder king = new StringBuilder("aaa");
8
9         ace.setCharAt(0, (char) (ace.charAt(0) + 4));
10        ace.setCharAt(1, (char) (ace.charAt(1) + 19));
11        ace.setCharAt(2, (char) (ace.charAt(2) + 18));
12
13        jack.setCharAt(0, (char) (jack.charAt(0) + 7));
14        jack.setCharAt(1, (char) (jack.charAt(1) + 0));
15        jack.setCharAt(2, (char) (jack.charAt(2) + 1));
16
17        queen.setCharAt(0, (char) (queen.charAt(0) + 0));
18        queen.setCharAt(1, (char) (queen.charAt(1) + 11));
19        queen.setCharAt(2, (char) (queen.charAt(2) + 15));
20
21        king.setCharAt(0, (char) (king.charAt(0) + 14));
22        king.setCharAt(1, (char) (king.charAt(1) + 20));
23        king.setCharAt(2, (char) (king.charAt(2) + 15));
24
25        // Concatenate all parts to form the password
26        String password = queen.toString() + jack.toString() + ace.toString() + king.toString();
27
28        // Print the password to the console
29        System.out.println("Password: " + password);
30    }
31 }

```

- Kết quả nhận được password:

```

ubuntu1@ubuntu1-virtual-machine:~/Desktop$ javac FlagstaffHill.java
ubuntu1@ubuntu1-virtual-machine:~/Desktop$ java FlagstaffHill
Password: alphabetsoup
ubuntu1@ubuntu1-virtual-machine:~/Desktop$

```

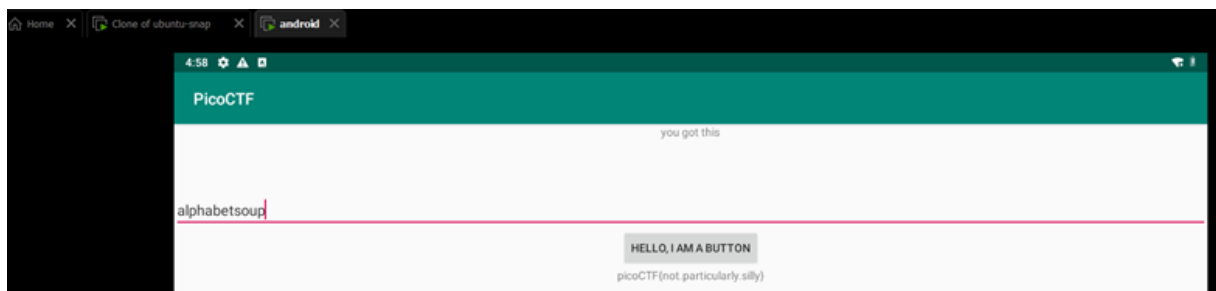
- Thực hiện sửa đoạn code smali trả về **call it** thành hàm **cardamon**.

```
232     if-eqz v5, :cond_0
233
234     invoke-static {p0}, Lcom/hellocmu/picocft/FlagstaffHill;->cardamon(Ljava/lang/String;)Ljava/lang/String;
235     move-result-object v0
236     return-object v0
237
238     .line 37
239     :cond_0
240     const-string v5, "NOPE"
241
242     return-object v5
243 .end method
```

- Thực hiện patch ứng và kí. Gỡ cài đặt ứng dụng cũ và cài ứng dụng mới.

```
Keystore password for signer #1: ubuntu@ubuntu-virtual-machine:~/Desktop/lab4/Challenges/Five/dis$ apksigner sign --ks my-release-key.keystore --out five_signed.apk five.apk
Keystore password for signer #1:
ubuntu@ubuntu-virtual-machine:~/Desktop/lab4/Challenges/Five/dis$ adb uninstall com.hellocmu.picocft
Success
ubuntu@ubuntu-virtual-machine:~/Desktop/lab4/Challenges/Five/dis$ adb install five_signed.apk
Performing Streamed Install
Success
ubuntu@ubuntu-virtual-machine:~/Desktop/lab4/Challenges/Five/dis$
```

- Kết quả khi chạy ứng dụng và dùng mật khẩu là **alphabetsoup**, ta thu được flag:



- Kết quả Flag: **picoCTF{not.particularly.silly}**