

BÁO CÁO BÀI TẬP

Môn học: Bảo mật Web và Ứng dụng

Kỳ báo cáo: Buổi 04 (Session 04)

Tên chủ đề: Android Pentesting

GV: Nghi Hoàng Khoa

Ngày báo cáo: 16/05/2023

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT213.N21.ANTN

STT	Họ và tên	MSSV	Email
1	Bùi Tấn Hải Đăng	20520173	20520173@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:¹

STT	Công việc	Kết quả tự đánh giá	Người đóng góp
1	Kịch bản 1: 1 → 12: EVABS	100%	Đăng
2	Kịch bản 2: 1 → 5: DROID	100%	Đăng

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

1. Kịch bản 01/Câu hỏi 01

Level 1:

- Dùng lệnh adb logcat để xem log của tất cả thiết bị đang chạy trên android.
- Tìm được flag:

```
C:\Windows\System32\cmd.exe
05-12 14:28:50.484 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4341835, only wrote 4341600
05-12 14:28:50.612 659 2029 E eglCodeCommon: goldfish_dma_create_region: could not obtain fd to device fd -1 errno=2
05-12 14:28:50.822 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4357852, only wrote 4353120
05-12 14:28:50.822 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4353125, only wrote 4353120
05-12 14:28:50.927 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4358118, only wrote 4353840
05-12 14:28:52.232 508 520 I ActivityManager: START u0 {cmp=com.revo.evabs/.DebugMe} from uid 10066
05-12 14:28:52.435 508 530 I ActivityManager: Displayed com.revo.evabs/.DebugMe: +183ms
05-12 14:28:54.289 1819 1819 D ** SYS_CTRL **: EVABS{logging_info_never_safel}
05-12 14:28:55.752 508 658 E TaskPersister: File error accessing recents directory (directory doesn't exist?).
05-12 14:28:57.467 217 252 W genymotion_audio: Not supplying enough data to HAL, expected position 4980410, only wrote 4667760
05-12 14:29:01.716 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4667828, only wrote 4667760
05-12 14:29:04.942 217 252 W genymotion_audio: Not supplying enough data to HAL, expected position 4975398, only wrote 4822560
05-12 14:30:07.108 508 1275 D WifiCondControl: Scan result ready event
05-12 14:30:31.952 235 235 I ->host-7: type=1400 audit(0.0:1246): avc: denied { write } for path="/socket:[9468]" dev="sockfs" ino=9468 scont
05-12 14:30:31.952 235 235 I ->host-7: type=1400 audit(0.0:1247): avc: denied { read } for path="/socket:[9468]" dev="sockfs" ino=9468 scont
05-12 14:30:35.664 1819 1819 D ** SYS_CTRL **: EVABS{logging_info_never_safel}
05-12 14:30:35.671 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4822774, only wrote 4822560
05-12 14:30:36.022 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4839400, only wrote 4834080
05-12 14:30:36.022 217 251 W genymotion_audio: Not supplying enough data to HAL, expected position 4834085, only wrote 4834080
05-12 14:30:37.068 508 508 I Binder:508_9: type=1400 audit(0.0:1248): avc: denied { sendto } for path="/dev/socket/logd" scontext=u:r:syst
e=1
```

- Ban đầu em có tìm bằng command: adb logcat -s com.revo.evabs nhưng không có kết quả flag.

```
D:\ADB\platform-tools>adb logcat -s com.revo.evabs
----- beginning of main -----
----- beginning of system -----
05-12 13:49:46.110 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1059): avc: denied { sendto } for path="/dev/socket/logd" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_dgram_socket permissive=1
05-12 13:49:46.346 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1071): avc: denied { write } for comm=45474C20496E6974 name="property_service" dev="tmpfs" ino=9274 scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:property_service:s0 tclass=sock_file permissive=1
05-12 13:49:46.422 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1072): avc: denied { write } for name="com.revo.evabs" dev="sdb3" ino=81891 scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0 tclass=dir permissive=1
05-12 13:49:46.422 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1073): avc: denied { add_name } for name="shared_prefs" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0 tclass=dir permissive=1
05-12 13:49:46.422 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1074): avc: denied { create } for name="shared_prefs" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0 tclass=dir permissive=1
05-12 13:49:46.422 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1075): avc: denied { setattr } for name="shared_prefs" dev="sdb3" ino=81844 scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=dir permissive=1
05-12 13:49:46.458 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1076): avc: denied { connectto } for comm=45474C20496E6974 path="/dev/socket/logd" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_dgram_socket permissive=1
05-12 13:49:46.458 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1086): avc: denied { write } for name="shared_prefs" dev="sdb3" ino=81844 scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=dir permissive=1
05-12 13:49:49.446 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1087): avc: denied { add_name } for name="REFERENCE.xml" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=dir permissive=1
05-12 13:49:49.446 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1088): avc: denied { create } for name="REFERENCE.xml" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=dir permissive=1
05-12 13:49:49.454 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1089): avc: denied { write open } for path="/data/data/com.revo.evabs/shared_prefs/PREFERENCE.xml" dev="sdb3" ino=81896 scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=file permissive=1
05-12 13:49:49.454 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1090): avc: denied { setattr } for name="REFERENCE.xml" dev="sdb3" ino=81896 scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=file permissive=1
05-12 14:18:24.253 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1156): avc: denied { sendto } for path="/dev/socket/logd" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_dgram_socket permissive=1
05-12 14:25:19.150 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1229): avc: denied { connectto } for comm=4173796E6354617368202331 path="/dev/socket/dnsproxyd" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_stream_socket permissive=1
05-12 14:26:09.747 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1243): avc: denied { connectto } for comm=4173796E6354617368202331 path="/dev/socket/dnsproxyd" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_stream_socket permissive=1
05-12 14:37:15.719 1819 1819 I com.revo.evabs: type=1400 audit(0.0:1278): avc: denied { sendto } for path="/dev/socket/logd" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_dgram_socket permissive=1
```

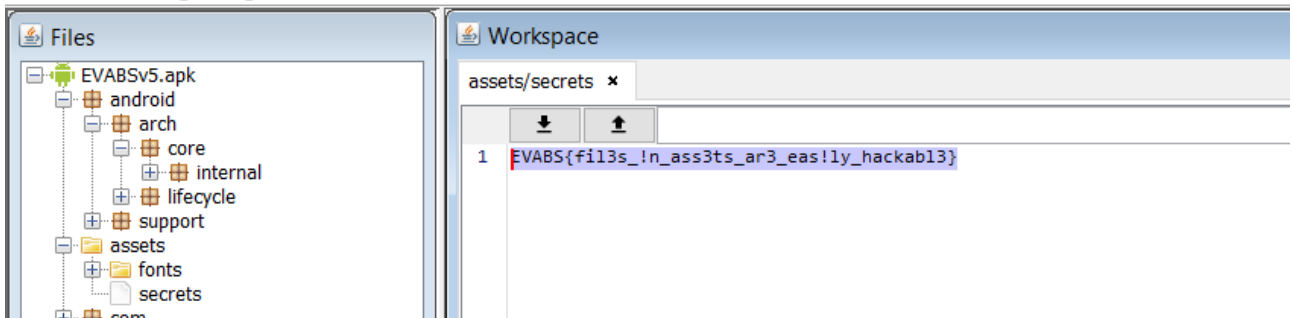
EVABS{logging_info_never_safel}

Level 2:

- Bỏ file apk vào Bytecode Viewer để xem, tại thư mục assets em tìm thấy file secret chứa flag.

Bytecode Viewer 2.11.2 - <https://bytecodeviewer.com> | <https://the.bytecode.club> - @Konloch

File View Settings Plugins



EVABS{f1l3s_!n_ass3ts_ar3_eas!ly_hackab13}

Level 3:

- Tiến hành decompile lại EVABSV5.apk bằng tool apktool với lệnh:

apktool d EVABSV5.apk

```
D:\ADB\platform-tools>apktool d EVABSV5.apk
I: Using Apktool 2.7.0 on EVABSV5.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: C:\Users\Dan\AppData\Local\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...
```

- Sau khi decompile xong thì mã nguồn được tạo ra bên trong thư mục EVABSV5, tại file res/values/strings.xml, nơi chứa các chuỗi trong ứng dụng em đọc được flag.

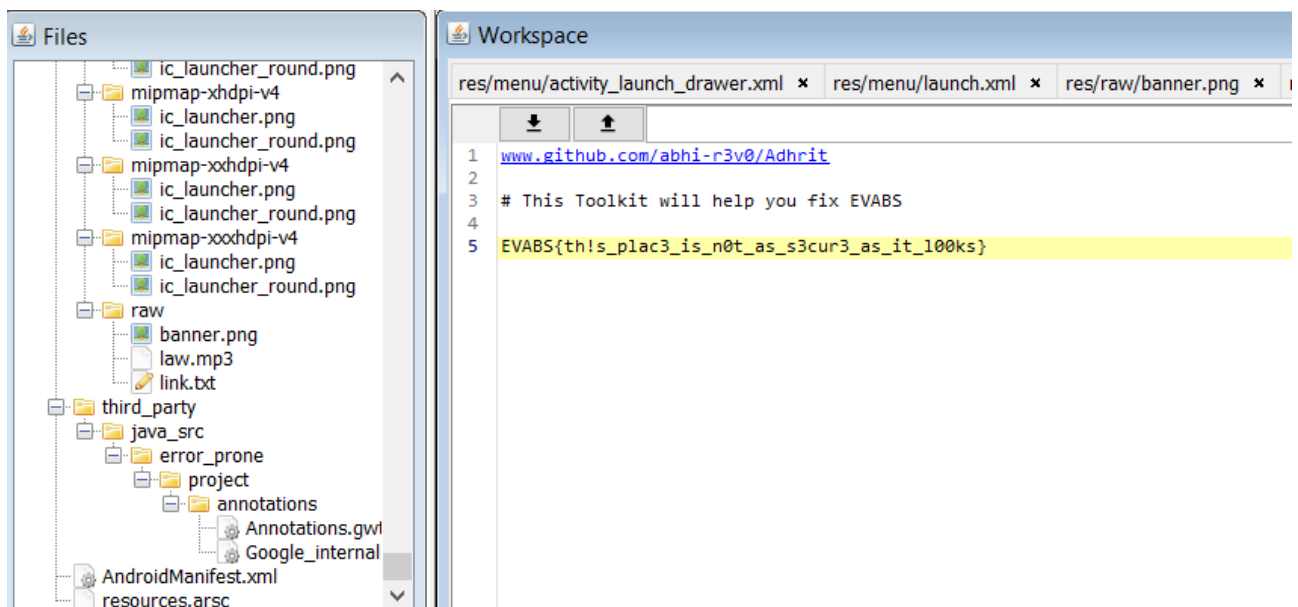
```

79 <string name="ob_desc2">If the person you're calling also has Wonep the call will be
80 <string name="ob_desc3">We have a very small charge for non-Wonep calls to mobiles o
81 <string name="ob_get_started">GET STARTED</string>
82 <string name="ob_header1">SIMPLE ABROAD CALLS</string>
83 <string name="ob_header2">FREE WONEP TO WONEP</string>
84 <string name="ob_header3">NO HIDDEN CHARGES OR FEES</string>
85 <string name="password_toggle_content_description">Toggle password visibility</string>
86 <string name="path_password_eye">M12,4.5C7,4.5 2.73,7.61 1,12c1.73,4.39 6,7.5 11,7.5
-5,-2.24 -5,-5s2.24,-5 5,-5 5,2.24 5,5 -2.24,5 -5,5zM12,9c-1.66,0 -3,1.34 -3,3s1.34,
87 <string name="path_password_eye_mask_strike_through">M2,4.27 L19.73,22 L22.27,19.46
88 <string name="path_password_eye_mask_visible">M2,4.27 L2,4.27 L4.54,1.73 L4.54,1.73
89 <string name="path_password_strike_through">M3.27,4.27 L19.74,20.74</string>
90 <string name="permission_rationale">"Contacts permissions are needed for providing e
91 completions."</string>
92 <string name="project_id">evabs-c0e8b</string>
93 <string name="prompt_email">Email</string>
94 <string name="prompt_password">Password (optional)</string>
95 <string name="search_menu_title">Search</string>
96 <string name="section_format">Hello World from section: %1$d</string>
97 <string name="status_bar_notification_info_overflow">999+</string>
98 <string name="the_evabs_api_key">EVABS{saf3ly_st0red_in_Strings?}</string>
99 <string name="title_activity_home">Home</string>
100 <string name="title_activity_launch">Launch</string>
101 <string name="title_activity_login">Sign in</string>
102 <string name="title_activity_splash">Splash</string>
103 <string name="title_activity_test">Test</string>
104 </resources>

```

Level 4:

- Load file apk vào bytecodeviewer, dựa vào hint của đề bài, em tìm đến res/raw/link.txt thì thấy có chứa flag.



EVABS{th!s_plac3_is_n0t_as_s3cur3_as_it_l00ks}

Level 5:

- Dựa vào hint của đề bài gợi ý về thư mục shared_prefs, em vào thư mục này.
- Tại /data/data/com.revo.evabs/shared_prefs/, tồn tại 2 file xml.

```
genymotion:/ # cd /data/data/com.revo.evabs/shared_prefs/
genymotion:/data/data/com.revo.evabs/shared_prefs # ls
DETAILS.xml  PREFERENCE.xml
```

- Vào file PREFERENCE.xml nhưng không có thông tin flag.

```
DETAILS.xml  PREFERENCE.xml
genymotion:/data/data/com.revo.evabs/shared_prefs # cat PREFERENCE.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
  <boolean name="isFirstRun" value="false" />
</map>
```

- Vào file DETAILS.xml thì có flag.

```
genymotion:/data/data/com.revo.evabs/shared_prefs # cat DETAILS.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
  <string name="password">EVABS{shar3d_pr3fs_c0uld_be_c0mpromiz3ds}</string>
  <string name="username">hidang</string>
</map>
```

EVABS{shar3d_pr3fs_c0uld_be_c0mpromiz3ds}

Level 6:

- Đề bài yêu cầu truy cập vào database của ứng dụng, dùng command adb shell để vào shell của app.
- Sau đó vào database của chương trình tại thư mục data/data/com.revo.evabs/databases/
- Tại đây có database MAINFRAME_ACCESS. Dùng lệnh .tables để xem các table có trong database, ở đây có tables CREDs chứa các thông tin tài khoản của người dùng.
- Truy xuất tất cả thông tin của tables này ta được flag.


```

D:\ADB\platform-tools>adb shell
genymotion:/ # cd data/data/com.revo.evabs/databases/
genymotion:/data/data/com.revo.evabs/databases # ls
MAINFRAME_ACCESS  MAINFRAME_ACCESS-journal
genymotion:/data/data/com.revo.evabs/databases # sqlite M
MAINFRAME_ACCESS  MAINFRAME_ACCESS-journal
genymotion:/data/data/com.revo.evabs/databases # sqlite3
MAINFRAME_ACCESS  MAINFRAME_ACCESS-journal
genymotion:/data/data/com.revo.evabs/databases # sqlite3 MAINFRAME_ACCESS
SQLite version 3.18.2 2017-07-21 07:56:09
Enter ".help" for usage hints.
sqlite> tables.
...> ;
Error: near "tables": syntax error
sqlite> .tables;
Error: unknown command or invalid arguments: "tables;". Enter ".help" for help
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>

```

EVABS{sqlite_is_not_safe}

Level 7:

- Dựa vào hint của đề bài là tìm các activity được gán nhãn exported, em vào file AndroidManifest.xml – nơi thông tin chi tiết của các Activity.
- Tại đây, quan sát thấy có 1 package có nhãn exported = true.



The screenshot shows the AndroidManifest.xml file in a code editor. A red box highlights the following XML element:

```

<activity android:exported="true" android:name="com.revo.evabs.ExportedActivity"/>

```

- Khi một Activity có thuộc tính exported bằng true thì activity này có thể được kích hoạt bởi các ứng dụng khác.
- Sử dụng lệnh adb shell như bên dưới để kích hoạt activity này.

```

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

```

- Sau khi kích hoạt thành công activity này, tại app evabs sẽ nhận được flag.

```
SYS_CTRL: ERROR: Secret service
breached. Data compromised:
EVABS{exp0rted_activities_ar3_harmful}E
```

Level 8:

- Dựa vào yêu cầu của đề bài, flag sẽ nằm trong source code của ứng dụng, vào class Decode để phân tích, tại đây có 1 chuỗi ciphertext.

```
JADX Decompiler
package com.revo.evabs;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.widget.Button;
import android.widget.TextView;
/* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-1863670937834284495\classes.dex */
public class Decode extends AppCompatActivity {
    protected void onCreate(Bundle bundle) {
        Decode.super.onCreate(bundle);
        setContentView(2131492896);
        String str = "RVZBQ1N7bmV2M3Jfc3QwcmU=X3MzbnMhdGl2M19kYXRhXzFuXzdoM19zMHVyY2VjMGRl";
        ((Button) findViewById(2131361842)).setOnClickListener(new 1(this, (TextView) findViewById(2131362094)));
    }
}
```

- Dùng python để giải mã chuỗi cipher này thì được flag.

```
>>> str = "RVZBQ1N7bmV2M3Jfc3QwcmU=X3MzbnMhdGl2M19kYXRhXzFuXzdoM19zMHVyY2VjMGRl"
>>> base64.b64decode(str)
b'EVABS{nev3r_st0re}'
>>> str = "RVZBQ1N7bmV2M3Jfc3QwcmU="
>>> base64.b64decode(str)
b'EVABS{nev3r_st0re}'
>>> str = "X3MzbnMhdGl2M19kYXRhXzFuXzdoM19zMHVyY2VjMGRl"
>>> base64.b64decode(str)
b'_s3ns!tiv3_data_1n_7h3_s0urcec0de'
>>>
```

Level 9:

- Yêu cầu của ứng dụng là thay đổi luồng thực thi của ứng dụng bằng cách chỉnh sửa chuỗi LAB_OFF thành LAB_ON.
- Ban đầu chuỗi v0 được gán bằng giá trị "LAB_OFF", chỉnh sửa nó thành "LAB_ON" như yêu cầu đề bài và tiến hành repackaging.

```
.method public constructor <init>()V
    .locals 1

    .line 11
    invoke-direct {p0}, Landroid/support/v7/app/CompatActivity;-><init>()V

    .line 13
    const-string v0, "LAB_OFF"

    iput-object v0, p0, Lcom/revo/evabs/SmaliInject;->SIGNAL:Ljava/lang/String;

    return-void
.end method
```

- Sau khi chỉnh sửa thành LAB_ON.



```
SmaliInject.smali X
SmaliInject.smali
19      .line 57
20      return-void
21  .end method
22
23  .method public constructor <init>()V
24      .locals 1
25
26      .line 11
27      invoke-direct {p0}, Landroid/support/v7/app/CompatActivity;-><init>()V
28
29      .line 13
30      const-string v0, "LAB_ON"
31
32      iput-object v0, p0, Lcom/revo/evabs/SmaliInject;->SIGNAL:Ljava/lang/String;
33
34      return-void
35  .end method
36
```

- Dùng apktool b để repackaging, sau khi thành công, file apk mới được lưu tại EVABSv5\dist\EVABSv5.apk.

```
D:\ADB\platform-tools>apktool b EVABSv5 EVABSv6.apk
I: Using Apktool 2.7.0
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: EVABSv5\dist\EVABSv5.apk
```


- Tiến hành uninstall package evabs trong ứng dụng android vì android không cho phép cài đặt 2 ứng dụng có cùng tên.

```
D:\ADB\platform-tools>adb uninstall com.revo.evabs
Success
```

- Tuy nhiên sẽ không thể install ngay được mà phải kí lên gói apk mới.

```
D:\ADB\platform-tools>adb install EVABsv5.apk
Performing Streamed Install
adb: failed to install EVABsv5.apk: Failure [INSTALL_PARSE_FAILED_NO_CERTIFICATES: Failed to collect certificates from /
data/app/vmdl1151656924.tmp/base.apk: Attempt to get length of null array]
```

- Tiến hành sinh key bằng tool keygen và kí bằng tool apksigner.

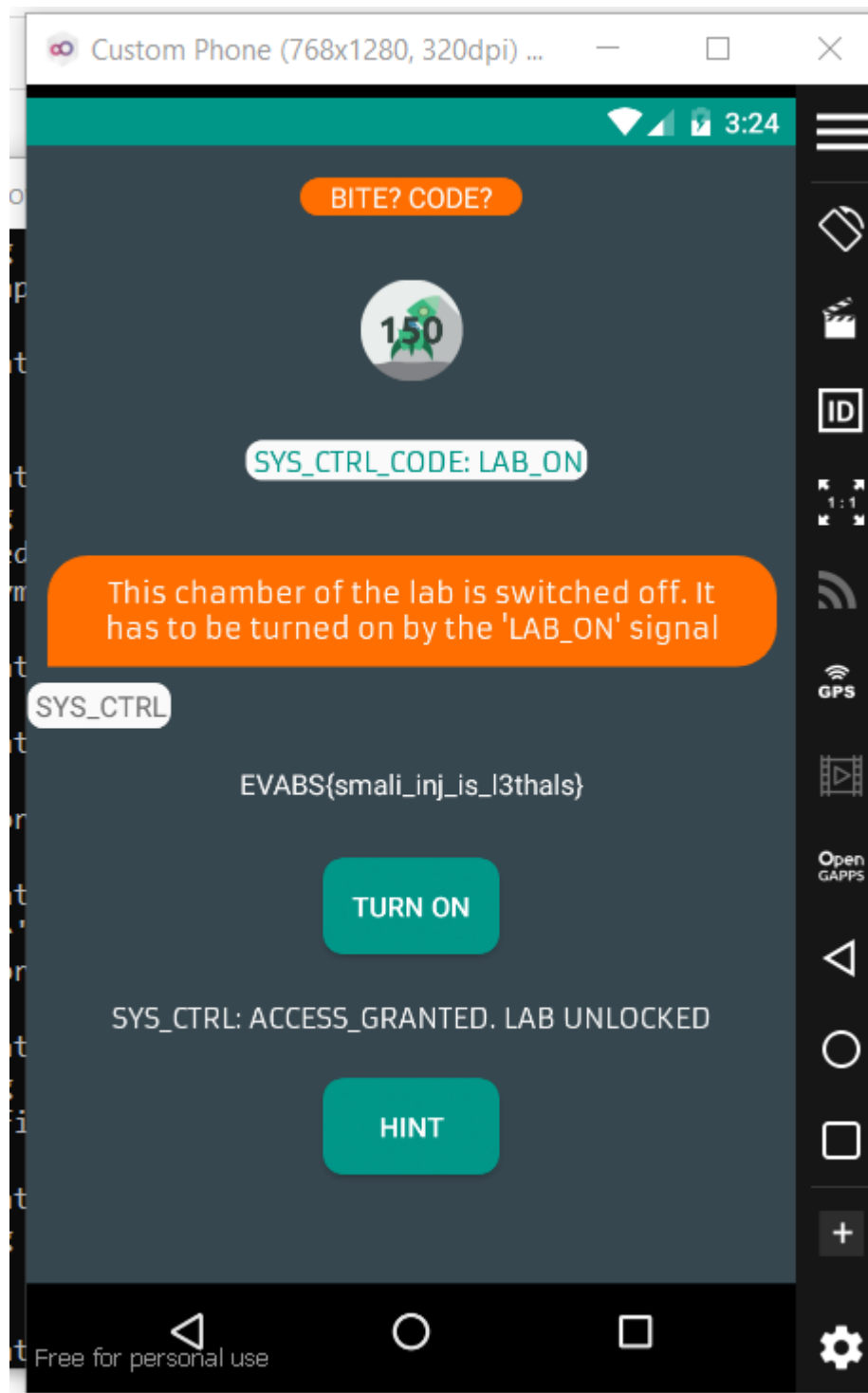
```
hidang@LAPTOP-0FG5MOP3:/tmp$ keytool -genkey -v -keystore EVABsv5.keystore -alias EVABsv5 -keyalg RSA -keysize 2048 -val
idity 10000
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]:
What is the name of your organizational unit?
[Unknown]:
What is the name of your organization?
[Unknown]:
What is the name of your City or Locality?
[Unknown]:
What is the name of your State or Province?
[Unknown]:
What is the two-letter country code for this unit?
[Unknown]:
Is CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?
[no]: yes
Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
for: CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
[Storing EVABsv5.keystore]
```

```
hidang@LAPTOP-0FG5MOP3:/tmp$ apksigner sign --ks EVABsv5.keystore EVABsv5.apk
Keystore password for signer #1:
```

- Sau khi kí thành công thì lúc này đã có thể dùng file apk này để install vào ứng dụng Android.

```
D:\ADB\platform-tools>adb install EVABsv5.apk
Performing Streamed Install
Success
```

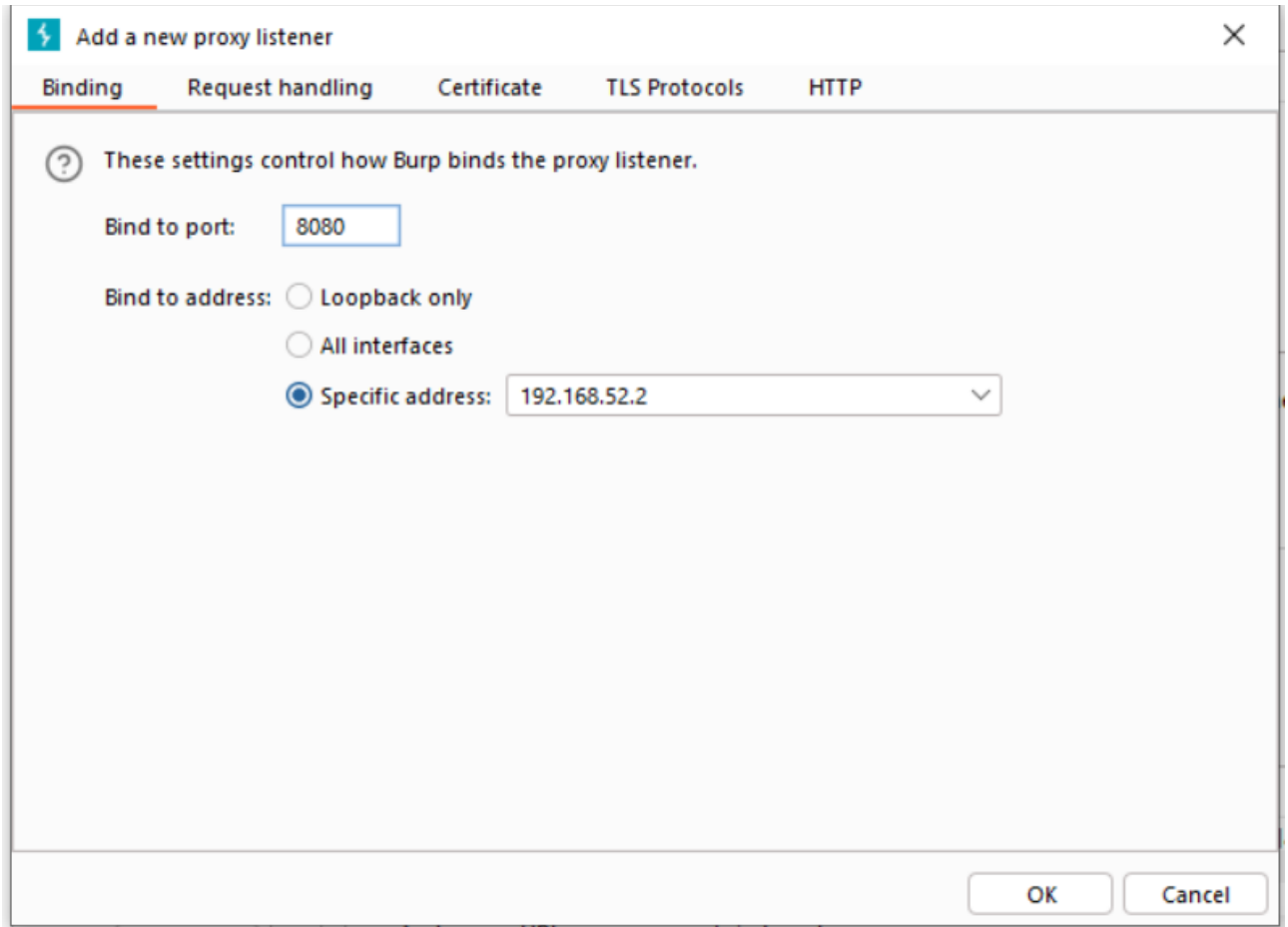
- Cài đặt ứng dụng thành công thì ta click vào button Turn On, lúc này flag sẽ hiện ra.

**Level 10:**

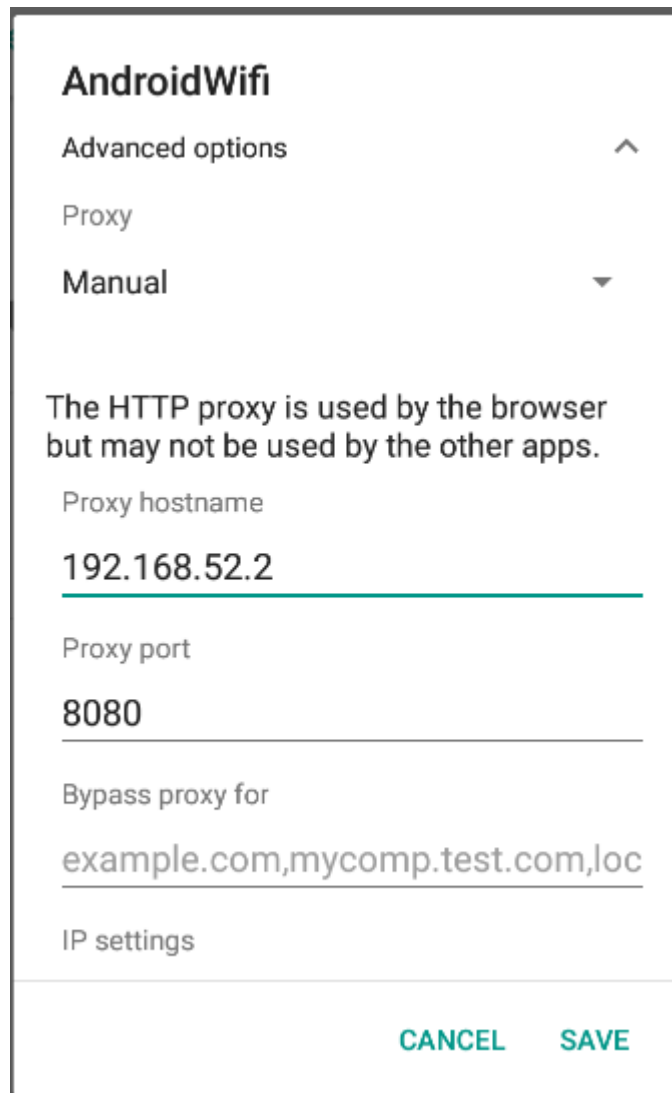
- Dựa theo hint của bài này là intercept các traffic qua android server. Em cấu hình Burp Suite làm proxy để bắt các gói tin.
- Xem địa chỉ IP của devices.

```
D:\ADB\platform-tools>adb devices
List of devices attached
192.168.52.101:5555    device
```

- Chọn địa chỉ IP cho proxy



- Modify wifi của thiết bị android



AndroidWifi

Advanced options ^

Proxy

Manual ▼

The HTTP proxy is used by the browser but may not be used by the other apps.

Proxy hostname

192.168.52.2

Proxy port

8080

Bypass proxy for

example.com,mycomp.test.com,loc

IP settings

CANCEL SAVE

- Sau đó nhấn vào button Receive để Proxy Burp Suite bắt các gói tin. Tuy nhiên có vẻ như server Android đã chết, nên nó không trả về response chứa flag cho chúng ta 😞.

Level 11:

- Dựa vào hint đề bài, phân tích source code của class Custom Access.
- Quan sát hàm GetSensorKey(), có thể thấy khi user nhập input bằng chuỗi cust0m_p3rm thì sẽ tạo ra một Intent mới. Tại intent này sẽ sử dụng phương thức putExtra() để in ra flag.

```

1 package com.revo.evabs;
2
3 import android.content.Context;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.support.v7.app.AppCompatActivity;
7 import android.widget.Button;
8 import android.widget.EditText;
9 import android.widget.TextView;
10 import android.widget.Toast;
11
12 /* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-7089746045694398505\classes.dex */
13 public class CustomAccess extends AppCompatActivity {
14     public final String EVABS_SENSOR_KEY = "com.revo.evabs.action.SENSOR_KEY";
15
16     static {
17         System.loadLibrary("native-lib");
18     }
19
20 /* JADX INFO: Access modifiers changed from: private */
21 /* JADX WARN: Multi-variable type inference failed */
22 public void GetSensorKey() {
23     if ("cust0m_p3rm".equals(((EditText) findViewById(2131361891)).getText().toString())) {
24         Toast.makeText((Context) this, (CharSequence) "SYS_CTRL: CRED5 ACCEPTED. SENSOR_KEY SENT", 1).show();
25         Intent intent = new Intent("com.revo.evabs.action.SENSOR_KEY");
26         intent.putExtra("android.intent.extra.TEXT", "EVABS{" + stringFromJNI() + "}");
27         intent.setType("text/plain");
28         startActivity(intent);
29         return;
30     }
31     Toast.makeText((Context) this, (CharSequence) "SYS_CTRL: WRONG_CRED5. SENSOR_KEY LOCKED", 1).show();
32 }
33
34 protected void onCreate(Bundle bundle) {
35     CustomAccess.super.onCreate(bundle);
36     setContentView(2131492893);
37     ((Button) findViewById(2131361835)).setOnClickListener(new 1(this));
38     ((Button) findViewById(2131361841)).setOnClickListener(new 2(this, (TextView) findViewById(2131362091)));
39 }
40
41 public native String stringFromJNI();
42 }
43

```

- Ý tưởng của em là sử dụng frida để hooking phương thức putExtra() trong class Custom Access. Em sẽ dùng Java.use() để ghi đè lên phương thức putExtra(), nhằm in ra flag.
- Ở bài này, vì hàm putExtra() có nhiều bản tùy theo kiểu tham số, nên em sử dụng overload() để chỉ ra đúng hàm putExtra() nhận tham số là 2 string. Tham số đầu tiên là android.intent.extra.Text và tham số thứ 2 là flag.
- Vậy sau khi hooking được hàm putExtra(), em sẽ in ra flag với tham số thứ 2.
- Đoạn code python sử dụng frida để hooking.

```
import frida
import sys

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

package = "com.revo.evabs"

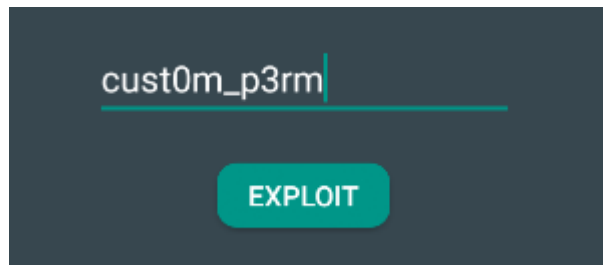
jscode = """
Java.perform(function () {
    send("[-] Starting hooks android.content.Intent.putExtra");
    var intent = Java.use("android.content.Intent");
    intent.putExtra.overload("java.lang.String", "java.lang.String").implementation = function(param_1, param_2) {
        send("-->The Flag is: " + param_2);
    };
});
"""

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

- Tiến hành chạy chương trình python aaa.py chứa đoạn script trên.

```
D:\ADB\platform-tools>py aaa.py
[+] Hooking successfully com.revo.evabs
{'type': 'send', 'payload': '[-] Starting hooks android.content.Intent.putExtra'}
```

- Tại máy ảo android, bật app Evabs và nhập đúng chuỗi cust0m_p3rm vào và ấn exploit



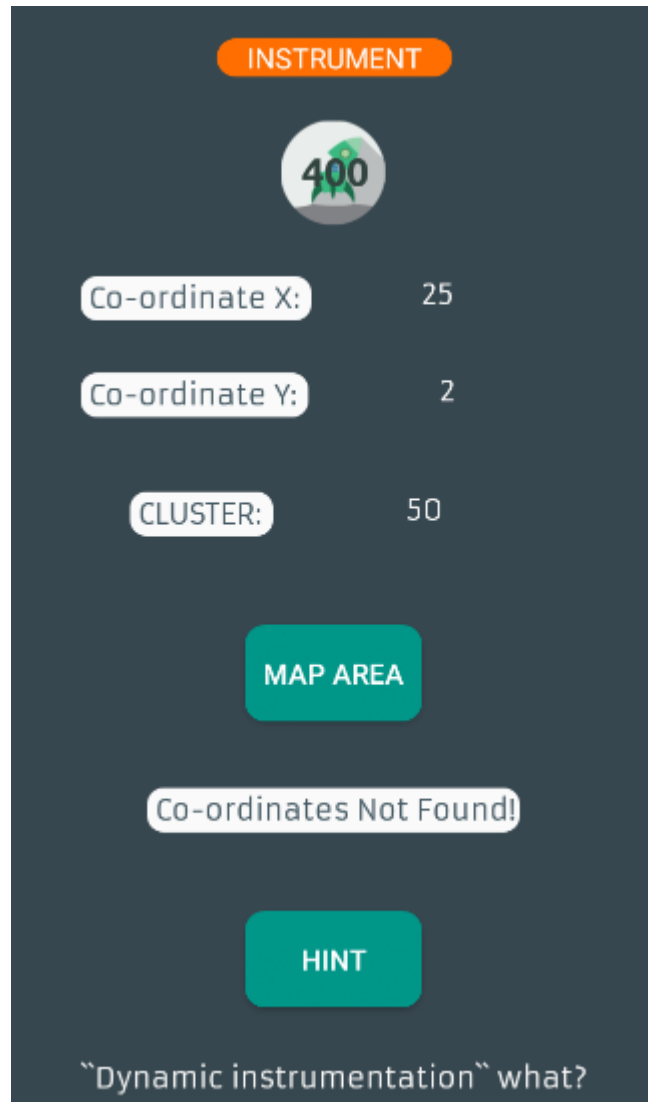
- Lúc này tại terminal python sẽ trả về flag.

```
D:\ADB\platform-tools>py aaa.py
[+] Hooking successfully com.revo.evabs
{'type': 'send', 'payload': '[-] Starting hooks android.content.Intent.putExtra'}
{'type': 'send', 'payload': '-->The Flag is: EVABS{always_ver1fy_packag3sa}'}
{'type': 'error', 'description': 'Error: Implementation for putExtra expected return value compatible with android.content.Intent', 'stack': 'Error: Implementation for putExtra expected return value compatible with android.content.Intent\n    at ne (frida/node_modules/frida-java-bridge/lib/class-factory.js:614)\n    at <anonymous> (frida/node_modules/frida-java-bridge/lib/class-factory.js:592)', 'fileName': 'frida/node_modules/frida-java-bridge/lib/class-factory.js', 'lineNumber': 614, 'columnNumber': 1}
```

- Flag: EVABS{always_ver1fy_packag3sa}

Level 12:

- Bài này sẽ cho một biến $X = 25$, biến $Y = 2$ và tính tích của nó ra. Ban đầu khi bấm vào button Map Area thì sẽ không cho ra flag.



- Tiếp tục đọc source code, phân tích class Frida1
- Tại hàm onClick(), chương trình sẽ gán giá trị tích của a, b là x – nghĩa là $25 \times 2 = 50$. Và giá trị x này là cố định.
- Sau đó chương trình sẽ kiểm tra giá trị x này có lớn hơn biến newInt + 150 hay không, nếu lớn hơn thì chương trình sẽ trả về flag và ngược lại thì không in ra gì. Biến newInt(70) sẽ random giá trị maximum là 70.
- Với logic trên, thì ta chỉ cần làm sao cho giá trị biến newInt + 150 < biến X (50).
- Vậy ta sẽ tiến hành sử dụng Frida để hooking hàm newInt(), làm cho nó luôn trả về một giá trị nhỏ bằng cách cho giá trị trả về của nó luôn trừ đi 200.

```

/* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-5756941945562235786\classes.dex */
public class Frida1 extends AppCompatActivity implements View.OnClickListener {
    int a = 25;
    int b = 2;
    int x;

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

    @Override // android.view.View.OnClickListener
    public void onClick(View view) {
        TextView textView = (TextView) findViewById(2131361996);
        ((TextView) findViewById(2131362132)).setText(String.valueOf(this.a));
        ((TextView) findViewById(2131362134)).setText(String.valueOf(this.b));
        this.x = this.a * this.b;
        int nextInt = new Random().nextInt(70);
        ((TextView) findViewById(2131362142)).setText(String.valueOf(this.x));
        if (this.x > nextInt + 150) {
            textView.setText("VIBRAN IS RESDY TO FLY! YOU ARE GOING HOME!");
            Log.d("CONGRATZ!", stringFromJNI());
            return;
        }
        textView.setText("Co-ordinates Not Found!");
    }

    protected void onCreate(Bundle bundle) {
        Frida1.super.onCreate(bundle);
        setContentView(2131492901);
        ((Button) findViewById(2131361902)).setOnClickListener(this);
        ((Button) findViewById(2131361844)).setOnClickListener(new 1(this, (TextView) findViewById(2131362
    }

    public native String stringFromJNI();
}

```

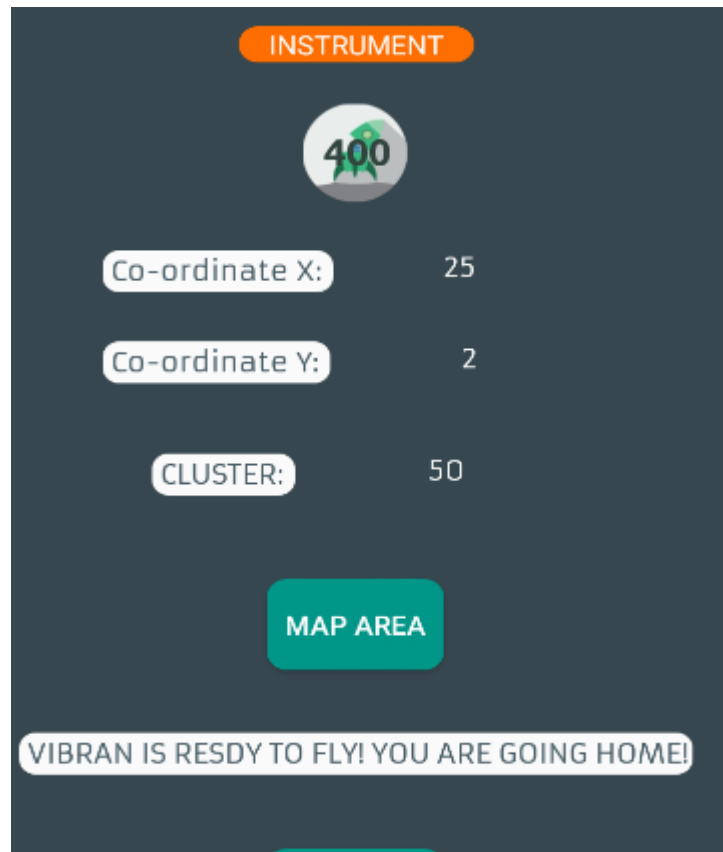
- Đoạn script python sử dụng frida hooking hàm newInt().

```
1 import frida
2 import sys
3
4 def onMessage(message, data):
5     print(message)
6
7 package = "com.revo.evabs"
8
9 jscode = """
10 Java.perform(function () {
11     send("[-] Starting hooks");
12     var random = Java.use("java.util.Random");
13     random.nextInt.overload("int").implementation = function(param_1) {
14         return -200;
15     };
16
17 });
18 """
19
20 process = frida.get_usb_device().attach(package)
21 script = process.create_script(jscode)
22 script.on("message", onMessage)
23 print("[+] Hooking successfully ", package)
24 script.load()
25 sys.stdin.read()
26
```

- Sau đó chạy chương trình python.

```
D:\ADB\platform-tools>py level12.py
[+] Hooking successfully com.revo.evabs
{'type': 'send', 'payload': '[-] Starting hooks'}
```

- Tại ứng dụng evabs(), click vào botton MAP AREA sẽ nhận được thông báo thành công.



- Tuy nhiên flag không được in ra ngay trong màn hình terminal đang chạy python. Vậy em đoán là nó sẽ được ghi vào trong logcat của ứng dụng này.
- Sử dụng lệnh adb shell ps để xem thông tin các process đang chạy trên ứng dụng Android.

```

u0_a69 1815 215 1259548 122000 ep_poll f2c34bb9 S com.revo.evabs
root 1849 2 0 0 worker_thread 0 S [kworker/1:1]
root 2104 234 6992 2488 0 f37f2bb9 R ps

```

- Có được pid của tiến trình evabs là 1815, sử dụng lệnh adb logcat để xem thì lúc này ta đã nhận được flag.

```

05-16 15:14:16.087 1815 1824 I zygote : Do partial code cache collection, code=110KB, data=69KB
05-16 15:14:16.089 1815 1824 I zygote : After code cache collection, code=110KB, data=69KB
05-16 15:14:16.089 1815 1824 I zygote : Increasing code cache capacity to 512KB
05-16 15:14:16.092 1815 1824 I zygote : JIT allocated 56KB for compiled code of void android.view.View.<init>(android
.content.Context, android.util.AttributeSet, int, int)
05-16 15:14:21.097 1815 1824 I zygote : Do full code cache collection, code=251KB, data=142KB
05-16 15:14:21.100 1815 1824 I zygote : After code cache collection, code=251KB, data=108KB
05-16 15:14:21.174 1815 1824 I zygote : Do partial code cache collection, code=251KB, data=111KB
05-16 15:14:21.175 1815 1824 I zygote : After code cache collection, code=251KB, data=111KB
05-16 15:14:21.176 1815 1824 I zygote : Increasing code cache capacity to 1024KB
05-16 15:16:14.737 1815 1815 I com.revo.evabs: type=1400 audit(0.0:4546): avc: denied { sendto } for path="/dev/socket
t/logdwn" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_dgram_socket permissive=1
05-16 15:20:25.415 1815 1824 I zygote : Do full code cache collection, code=499KB, data=297KB
05-16 15:20:25.416 1815 1824 I zygote : After code cache collection, code=487KB, data=260KB
05-16 15:33:25.882 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E
05-16 15:33:32.282 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E
05-16 15:33:33.570 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E
05-16 15:35:47.262 1815 1824 I zygote : Do partial code cache collection, code=497KB, data=275KB
05-16 15:35:47.262 1815 1824 I zygote : After code cache collection, code=497KB, data=275KB
05-16 15:35:47.262 1815 1824 I zygote : Increasing code cache capacity to 2MB
05-16 15:35:57.210 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E
05-16 15:36:02.933 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E
05-16 15:40:38.422 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E
05-16 15:40:38.418 1815 1815 I com.revo.evabs: type=1400 audit(0.0:20905): avc: denied { sendto } for path="/dev/socket
t/logdwn" scontext=u:r:untrusted_app:s0:c512,c768 tcontext=u:r:init:s0 tclass=unix_dgram_socket permissive=1
05-16 15:40:38.422 1815 1815 D CONGRATZ!: EVABS{a_dynam1c_h00k}E

```

D.2 Droid

Định dạng flag: picoCTF{th1s_1s_fl2g}

Challenges 2 Hoàn thành 5 challenges

One.apk.

- Yêu cầu để bài bảo tìm log của ứng dụng Droid, dùng command adb logcat >> log.txt để chuyển toàn bộ log trên ứng dụng vào file log.txt.

```

D:\ADB\platform-tools>adb logcat >> log.txt
^C

```

- Ctrl F để tìm flag của bài này thông qua keyword picoCTF.

log.txt - Notepad

```

File Edit Format View Help
05-14 03:57:06.132 216 248 W genymotion_audio: Not supplying enough data to HAL, expected position 28791368, only wrote 28791360
05-14 03:57:06.408 3550 3550 I PICO : picoCTF{a.moose.once.bit.my.sister}
05-14 03:57:09.339 660 660 I LatinIME: Starting input. Cursor position = 0,0
05-14 03:57:09.586 479 504 E SurfaceFlinger: Failed to find layer (PopupWindow:2cd122#0) in layer parent (no-parent).
05-14 03:57:09.598 216 249 W genymotion_audio: Not supplying enough data to HAL, expected position 29122822, only wrote 28957680
05-14 03:57:09.788 660 785 I LatinIME:LogUtils: Dictionary info: dictionary = spellcheck_userunigram.en_US; version = 1684036629; date = ?
05-14 03:57:09.876 1440 1440 I Binder:1440_3: type=1400 audit(0.0:1648): avc: denied { lock } for path="/data/data/com.android.providers.userdictio
nary/tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=file permissive=1
05-14 03:57:09.904 660 660 I ExecutorUtils: type=1400 audit(0.0:1649): avc: denied { lock } for path="/data/user_de/0/com.android.inputmethod.latin/
tcontext=u:object_r:system_data_file:s0:c512,c768 tclass=file permissive=1
05-14 03:57:09.925 660 785 I LatinIME:LogUtils: Dictionary info: dictionary = main:en; version = 54; date = 1414726273
05-14 03:57:12.246 3550 3550 I PICO : picoCTF{a.moose.once.bit.my.sister}
05-14 03:57:12.256 216 248 W genymotion_audio: Not supplying enough data to HAL, expected position 28957922, only wrote 28957680
05-14 03:57:15.482 216 249 W genymotion_audio: Not supplying enough data to HAL, expected position 29265226, only wrote 29112480
05-14 03:57:19.239 230 230 I rild : type=1400 audit(0.0:1650): avc: denied { sendto } for path="/dev/socket/logdwn" scontext=u:r:rild:s0 tcontext=u:r:
05-14 03:57:20.394 3550 3550 I PICO : picoCTF{a.moose.once.bit.my.sister}
05-14 03:57:20.409 216 248 W genymotion_audio: Not supplying enough data to HAL, expected position 29112749, only wrote 29112480
05-14 03:57:20.713 216 248 W genymotion_audio: Not supplying enough data to HAL, expected position 29127084, only wrote 29124000
05-14 03:57:20.714 216 248 W genymotion_audio: Not supplying enough data to HAL, expected position 29124007, only wrote 29124000
05-14 03:57:22.083 660 1947 E eglCodecCommon: goldfish dma create region: could not obtain fd to device! fd -1 errno=2

```

picoCTF{a.moose.once.bit.my.sister}

Two.apk.

- Dùng bytecode viewer để phân tích file two.apk, tại class MainActivity có hàm buttonClick và trong hàm này sẽ gọi đến class FlagstaffHill sử dụng phương thức getFlag(), tham số của phương thức getFlag() là giá trị biến text_input từ user nhập vào.

```

/* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-1128307895681653040\classes.dex */
public class MainActivity extends AppCompatActivity {
    Button button;
    Context ctx;
    TextView text_bottom;
    EditText text_input;
    TextView text_top;

    public void buttonClick(View view) {
        this.text_bottom.setText(FlagstaffHill.getFlag(this.text_input.getText().toString(), this.ctx));
    }

    protected void onCreate(Bundle bundle) {
        MainActivity.super.onCreate(bundle);
        setContentView(2131296284);
        this.text_top = (TextView) findViewById(2131165322);
        this.text_bottom = (TextView) findViewById(2131165320);
        this.text_input = (EditText) findViewById(2131165321);
        this.ctx = getApplicationContext();
        System.loadLibrary("hellojni");
        this.text_top.setText(2131427372);
    }
}

```

- Bên trong class FlagstaffHill, hàm getFlag() sẽ so sánh chuỗi input với getString(...). Nếu như 2 chuỗi bằng nhau thì trả về flag là chuỗi trả về của hàm fenugreek(str) ngược lại thì trả về NOPE.

```

package com.hellocmu.picocft;

import android.content.Context;

/* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-5528240965104224136\classes.dex */
public class FlagstaffHill {
    public static native String fenugreek(String str);

    public static String getFlag(String str, Context context) {
        return str.equals(context.getString(2131427375)) ? fenugreek(str) : "NOPE";
    }
}

```


- Vậy tham số bên trong hàm getString là gì, nó chính là giá trị id của các trường thông tin nằm trong file public.xml.

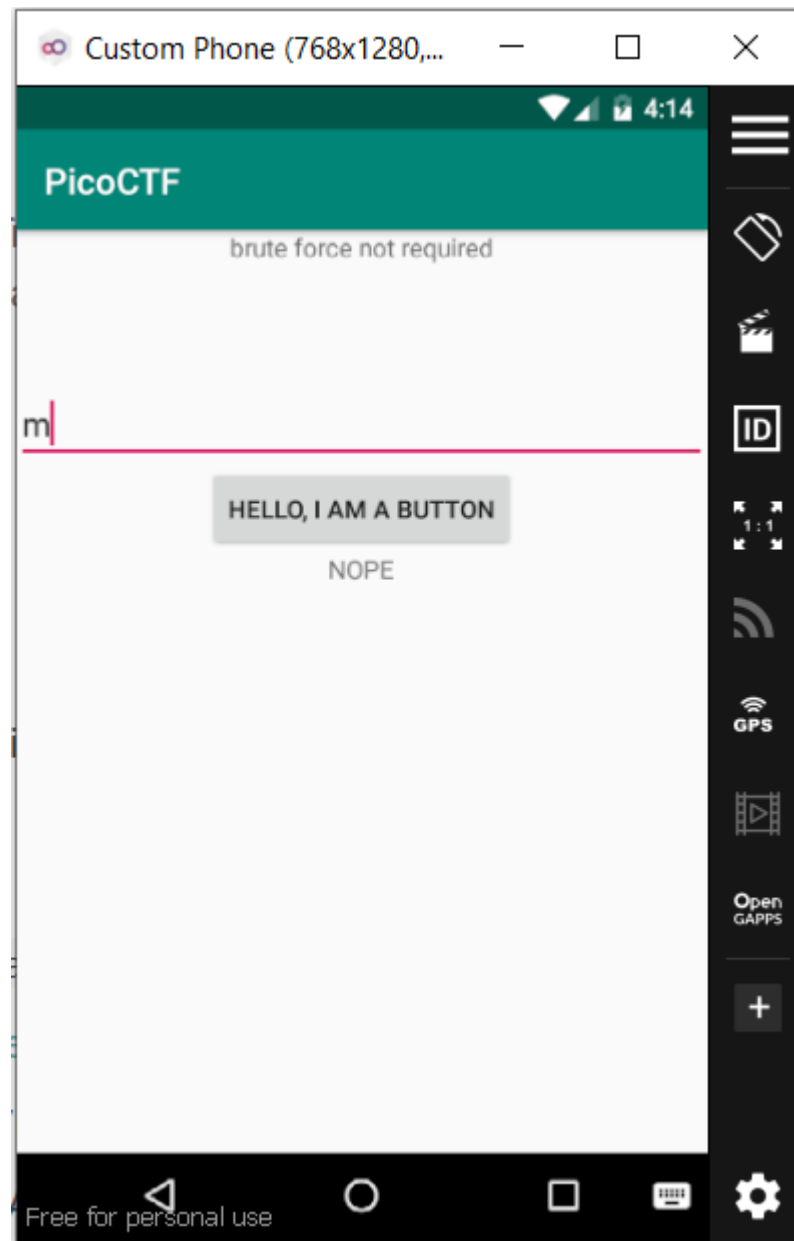
- Sau khi convert từ DEC sang HEX, ta được 1 giá trị id, dùng giá trị id đó để tìm trong file public.xml ta được 1 trường public tên là password với type là string.

```
<public type="string" name="password" id="0x7f0b002f" />
<public type="string" name="porcupine" id="0x7f0b0030" />
```

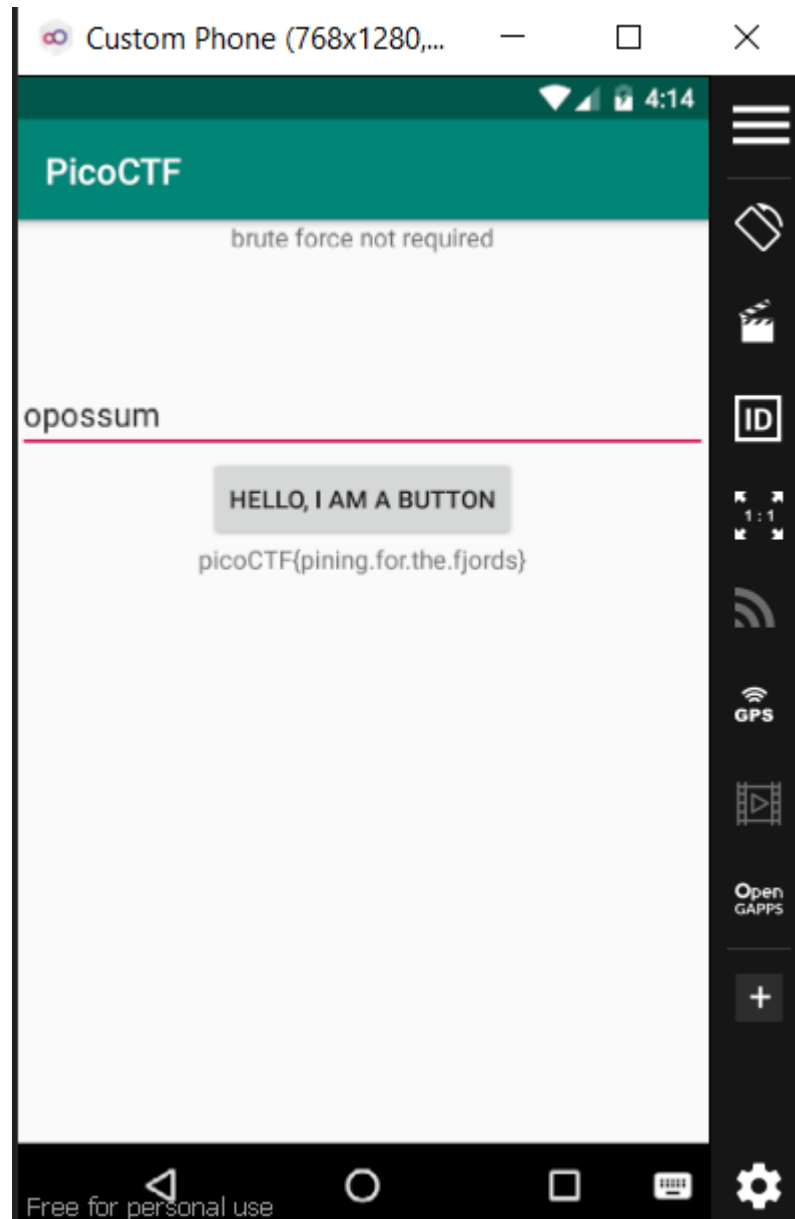
- Vào file string.xml để tìm string có tên là password, ta được giá trị của trường này là opossum.

```
<string name="myotis">jackrabbit</string>
<string name="password">opossum</string>
```

- Nếu như nhập giá trị bất kì thì nó sẽ trả về NOPE.



- Khi nhập chuỗi opossum thì ứng dụng sẽ trả về flag.



Three.apk

- Cũng như bài trên, em tiến hành vào phân tích lớp FlagstaffHill để xem cách hoạt động của hàm getflag().

```
public void buttonClick(View view) {
    this.text_bottom.setText(FlagstaffHill.getFlag(this.text_input.getText().toString(), this.ctx));
}
```

```
1 package com.hellocmu.picoctf;
2
3 import android.content.Context;
4
5 /* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-4474374215337280712\classes.dex */
6 public class FlagstaffHill {
7     public static String getFlag(String str, Context context) {
8         String[] strArr = {"weatherwax", "ogga", "garlick", "nitt", "aching", "dismiss"};
9         return str.equals("".concat(strArr[5]).concat(".").concat(strArr[1]).concat(".").concat(strArr[0]).concat(".").concat(strArr[4]).concat(".").concat(strArr[3]).concat(".").concat(strArr[2])) ? sesame(str) : "NOPE";
10     }
11
12     public static native String sesame(String str);
13 }
14
```

- Dùng chatGPT để cho nó giải hàm này. Kết quả thu được là khi nhập vào 1 chuỗi mà chatGPT trả về thì chương trình sẽ gọi hàm `sesame()` để in ra flag.

H cho tôi 1 chuỗi thỏa yêu cầu equal của hàm trên



Để có một chuỗi thỏa mãn yêu cầu



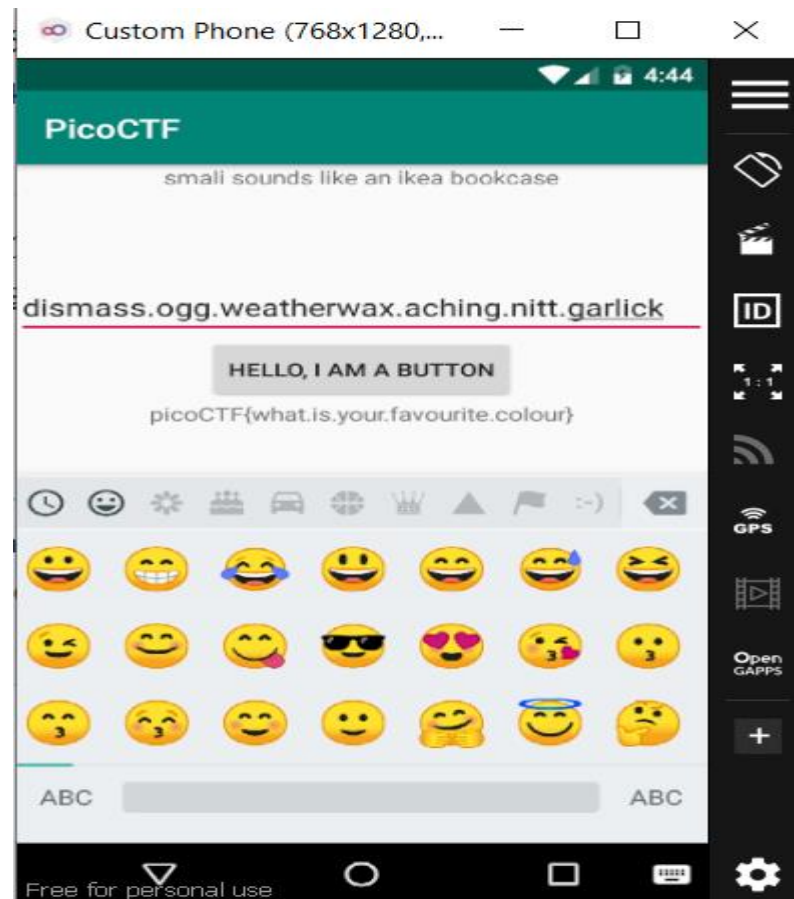
``str.equals("").concat(strArr[5]).concat(".").concat(strArr[1]).concat(".").concat(strArr[0]).concat(".").concat(strArr[4]).concat(".").concat(strArr[3]).concat(".").concat(strArr[2])``, ta có thể sử dụng các giá trị trong mảng ``strArr`` đã được định nghĩa trong phương thức ``getFlag``. Vì vị trí các phân tử trong mảng quan trọng, chúng ta cần xác định đúng thứ tự của chúng. Dựa vào đoạn mã, một chuỗi thỏa mãn có thể là:

Copy code

`dismiss.ogg.weatherwax.aching.nitt.garlick`

Chuỗi trên có đúng thứ tự như trong mảng ``strArr`` và đáp ứng yêu cầu của phương thức ``getFlag``.

Regenerate response



Four.apk

- Cũng như các yêu cầu trên, tiến hành phân tích class FlagstaffHill. Tuy nhiên thì ở bài này hàm getFlag không hề gọi đến hàm yep() – hàm để tạo ra flag về cho user mà chỉ gọi đến hàm nope(). Điều này làm ta không thể in ra flag nếu như vẫn giữ đúng luồng thực thi của chương trình.

```

1 package com.hellocmu.picoctf;
2
3 import android.content.Context;
4
5 /* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-2272670446827267364\classes.dex */
6 public class FlagstaffHill {
7     public static native String cilantro(String str);
8
9     public static String getFlag(String str, Context context) {
10         return nope(str);
11     }
12
13     public static String nope(String str) {
14         return "don't wanna";
15     }
16
17     public static String yep(String str) {
18         return cilantro(str);
19     }
20 }
21

```

- Do đó em sẽ thay đổi đoạn code trong smali của nó. Bằng việc sử dụng công cụ Bytecode Viewer, tại đây ô khoanh đỏ thấy sau khi gọi hàm getFlag() thì nó sẽ gọi đến hàm nope().

```

.method public static getFlag(Ljava/lang/String;Landroid/content/Context;)Ljava/lang/String;
    .registers 3
    invoke-static { p0 }, Lcom/hellocmu/picoctf/FlagstaffHill; ->nope(Ljava/lang/String;)Ljava/lang/String;
    move-result-object v0
    return-object v0
.end method

.method public static nope(Ljava/lang/String;)Ljava/lang/String;
    .registers 2
    const-string v0, "don't wanna"
    return-object v0
.end method

.method public static yep(Ljava/lang/String;)Ljava/lang/String;
    .registers 2
    invoke-static { p0 }, Lcom/hellocmu/picoctf/FlagstaffHill; ->cilantro(Ljava/lang/String;)Ljava/lang/Str
    move-result-object v0
    return-object v0
.end method

```

- Ý tưởng là sẽ sửa tên hàm này thành hàm yep() để cho chương trình gọi hàm cilantro() để in flag.
- Tiến hành dùng apktool để decompile file four.apk và edit đoạn code trong smali

```
D:\ADB\platform-tools>apktool d four.apk
I: Using Apktool 2.7.0 on four.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: C:\Users\Dan\AppData\Local\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...
```

- Sau khi decompile, sửa tên hàm nope thành yep như bên dưới để thay đổi luồng hoạt động của chương trình.

```
.method public static getFlag(Ljava/lang/String;Landroid/content/Context;)Ljava/lang/String;
    .locals 1
    .param p0, "input"    # Ljava/lang/String;
    .param p1, "ctx"      # Landroid/content/Context;

    .line 19
    invoke-static {p0}, Lcom/helloctmu/picoctf/FlagstaffHill; >yep(Ljava/lang/String;)Ljava/lang/String;

    move-result-object v0

    .line 20
    .local v0, "flag":Ljava/lang/String;
    return-object v0
.end method
```

- Chỉnh sửa xong thì em tiến hành recompile lại với command: *apktool b four*

```
D:\ADB\platform-tools>apktool b four fourv2.apk
I: Using Apktool 2.7.0
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: four\dist\four.apk
```

- Lúc này bỏ file four.apk vào bytecode viewer thì thấy luồng hoạt động đã thay đổi.

```
.method public static getFlag(Ljava/lang/String;Landroid/content/Context;)Ljava/lang/String;
    .registers 3
    invoke-static { p0 }, Lcom/helloctmu/picoctf/FlagstaffHill; >yep(Ljava/lang/String;)Ljava/lang/String;
    move-result-object v0
    return-object v0
.end method
```



```

/* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-6457149204521795639\classes.dex */
public class FlagstaffHill {
    public static native String cilantro(String str);

    public static String getFlag(String str, Context context) {
        return yep(str);
    }

    public static String nope(String str) {
        return "don't wanna";
    }

    public static String yep(String str) {
        return cilantro(str);
    }
}

```

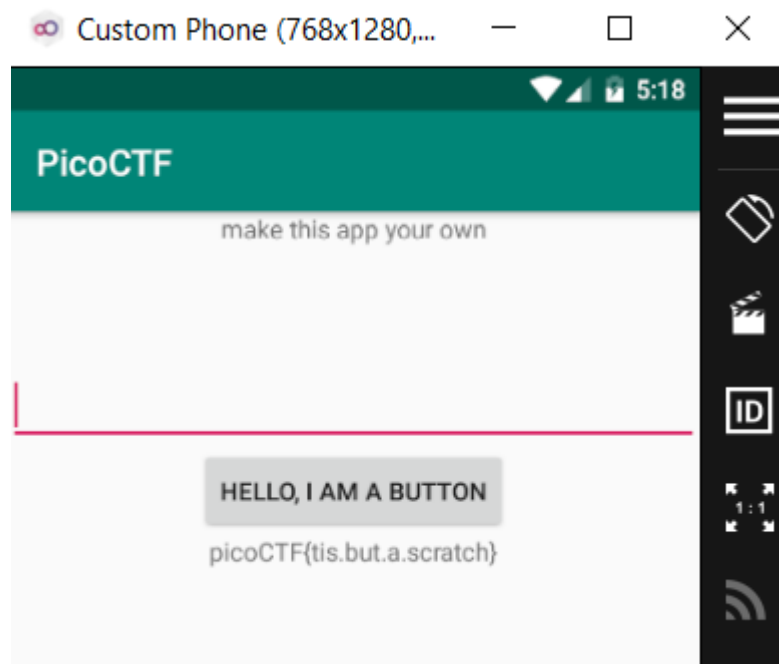
- Tiếp tục tạo keystore và kí vào file four.apk và reinstall lại.

```

hidang@LAPTOP-0FG5MOP3:/tmp$ apksigner sign --ks four.keystore four.apk
Keystore password for signer #1:

```

- Khi bật ứng dụng picoCTF và click vào button thì sẽ tự động hiện ra flag.



Five.apk.

- Cũng như những chall của Droids trước, em sẽ vào class FlagstaffHill để phân tích đoạn mã getFlag(). Tại đây có 1 đoạn code và em sẽ dùng tool online để giải mã đoạn code này.

```

public static String getFlag(String str, Context context) {
    StringBuilder sb = new StringBuilder("aaa");
    StringBuilder sb2 = new StringBuilder("aaa");
    StringBuilder sb3 = new StringBuilder("aaa");
    StringBuilder sb4 = new StringBuilder("aaa");
    sb.setCharAt(0, (char) (sb.charAt(0) + 4));
    sb.setCharAt(1, (char) (sb.charAt(1) + 19));
    sb.setCharAt(2, (char) (sb.charAt(2) + 18));
    sb2.setCharAt(0, (char) (sb2.charAt(0) + 7));
    sb2.setCharAt(1, (char) (sb2.charAt(1) + 0));
    sb2.setCharAt(2, (char) (sb2.charAt(2) + 1));
    sb3.setCharAt(0, (char) (sb3.charAt(0) + 0));
    sb3.setCharAt(1, (char) (sb3.charAt(1) + 11));
    sb3.setCharAt(2, (char) (sb3.charAt(2) + 15));
    sb4.setCharAt(0, (char) (sb4.charAt(0) + 14));
    sb4.setCharAt(1, (char) (sb4.charAt(1) + 20));
    sb4.setCharAt(2, (char) (sb4.charAt(2) + 15));
    return str.equals("").concat(sb3.toString()).concat(sb2.toString()).concat(sb.toString()).concat(sb4.toString()) ? "call it" : "NOPE";
}

```

- Kết quả sau khi giải mã thì được 1 chuỗi là alphabetsoup, nghĩa là khi user nhập input là chuỗi này vào thì ứng dụng sẽ hiển thị là call it và ngược lại thì in ra NOPE.

Main.java	Run	Output
<pre> 1 // Online Java Compiler 2 // Use this editor to write, compile and run your Java code online 3 4 class HelloWorld { 5 public static void main(String[] args) { 6 StringBuilder sb = new StringBuilder("aaa"); 7 StringBuilder sb2 = new StringBuilder("aaa"); 8 StringBuilder sb3 = new StringBuilder("aaa"); 9 StringBuilder sb4 = new StringBuilder("aaa"); 10 sb.setCharAt(0, (char) (sb.charAt(0) + 4)); 11 sb.setCharAt(1, (char) (sb.charAt(1) + 19)); 12 sb.setCharAt(2, (char) (sb.charAt(2) + 18)); 13 sb2.setCharAt(0, (char) (sb2.charAt(0) + 7)); 14 sb2.setCharAt(1, (char) (sb2.charAt(1) + 0)); 15 sb2.setCharAt(2, (char) (sb2.charAt(2) + 1)); 16 sb3.setCharAt(0, (char) (sb3.charAt(0) + 0)); 17 sb3.setCharAt(1, (char) (sb3.charAt(1) + 11)); 18 sb3.setCharAt(2, (char) (sb3.charAt(2) + 15)); 19 sb4.setCharAt(0, (char) (sb4.charAt(0) + 14)); 20 sb4.setCharAt(1, (char) (sb4.charAt(1) + 20)); 21 sb4.setCharAt(2, (char) (sb4.charAt(2) + 15)); 22 System.out.println("").concat(sb3.toString()).concat(sb2.toString()).concat (sb.toString()).concat(sb4.toString()); 23 24 25 } </pre>	Run	<pre> java -cp /tmp/2QaBkfMgbv HelloWorld alphabetsoup </pre>

➔ Str = alphabetsoup

- Quan sát thêm thì ngoài ra còn hàm cardamom() chưa được gọi đến và có thể hàm này sẽ là hàm để in ra flag cho chúng ta.

```

/* Loaded from: C:\Users\Dan\AppData\Local\Temp\jadx-84129983192402
public class FlagstaffHill {
    public static native String cardamom(String str);
}

```

- Ý tưởng cũng như challenge four.apk, vì em sẽ thay đổi luồng thực thi của chương trình, thay vì in ra "call it" thì em sẽ cho nó gọi hàm cardamom(str).
- Quan sát đoạn code smali tương ứng với hàm getFlag() trên, có thể thấy kết quả trả về được gán cho v0 và nó sẽ so sánh v0 với 0, nếu như bằng 0 thì sẽ nhảy đến nhãn L1 (NOPE) còn không thì tiếp tục thực thi nhãn L0 để in "call it"

```

move-result-object v0
invoke-virtual { p0, v0 }, Ljava/lang/String;->equals(Ljava/lang/Object;)Z
move-result v0
if-eqz v0, :L1
const-string v0, "call it"
:L0
return-object v0
:L1
const-string v0, "NOPE"
goto :L0
.end method

```

- Tiến hành decompile file five.apk với command `apktool d five.apk`

```

D:\ADB\platform-tools>apktool d five.apk
I: Using Apktool 2.7.0 on five.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: C:\Users\Dan\AppData\Local\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...

```

- Thay đổi luồng thực thi chương trình, em đổi dòng **const-string v0, "call it"** như khoảng đỏ bên dưới để gọi đến hàm `cardamom` nhằm in ra flag.

```

move-result v5

if-eqz v5, :cond_0

invoke-static { p0 }, Lcom/helloctmu/picoctf/FlagstaffHill;->cardamom(Ljava/lang/String;)Ljava/lang/String;
move-result-object v5

return-object v5

.line 37
:cond_0
const-string v5, "NOPE"

return-object v5
.end method

```

- Sau khi chỉnh sửa file smali xong thì em recompile lại.

```

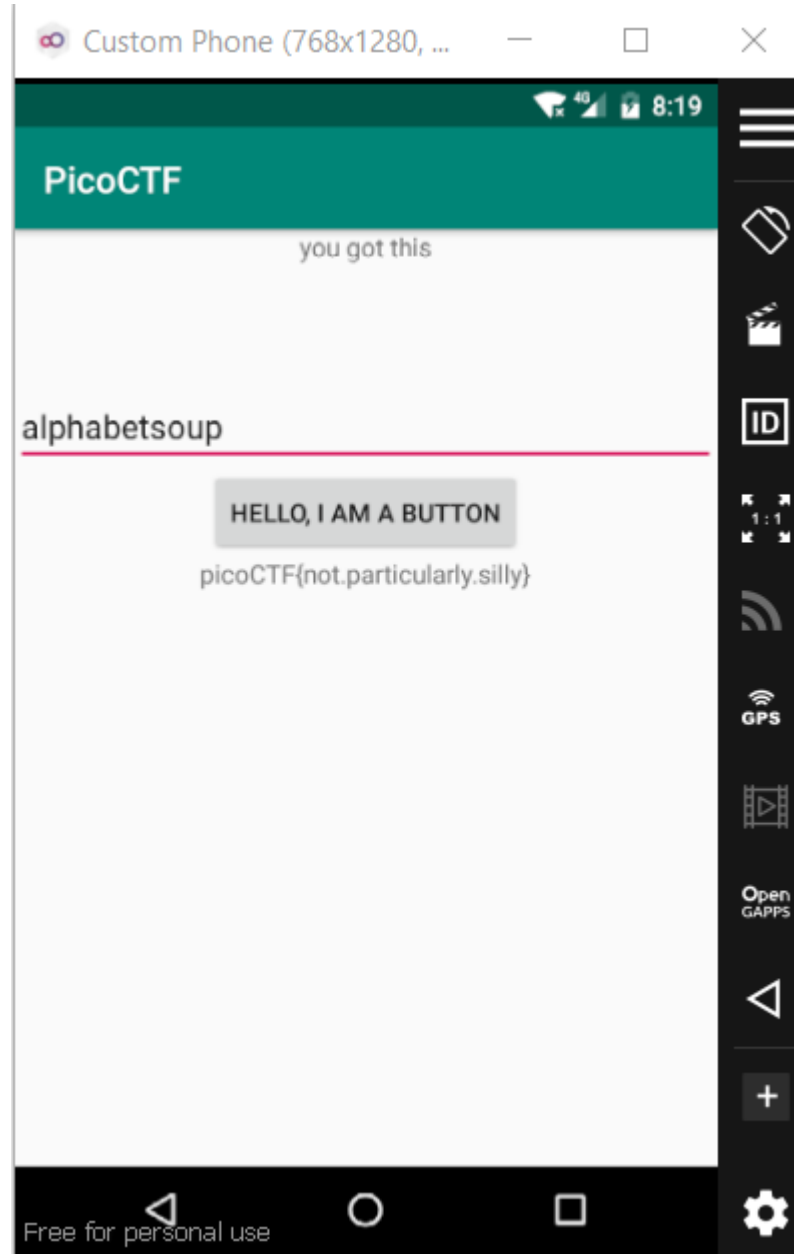
D:\ADB\platform-tools>apktool b five
I: Using Apktool 2.7.0
I: Checking whether sources has changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources has changed...
I: Building resources...
I: Copying libs... (/lib)
I: Building apk file...
I: Copying unknown files/dir...
I: Built apk into: five\dist\five.apk

```

- Tiến hành kí cho five.apk và reinstall app.

```
hidang@LAPTOP-0FG5M0P3:/tmp$ apksigner sign --ks five.keystore five.apk
Keystore password for signer #1:
```

- Nhập vào string tìm được ở bước giải mã trên để nhập vào và nhận được flag.



HẾT