BÁO CÁO THỰC HÀNH

**Bài thực hành số 04: Challenges CTF**

**Môn học:** Bảo mật Web và Ứng dụng

**Lớp:** NT213.P12.ANTT.1

**THÀNH VIÊN THỰC HIỆN (Nhóm xx):**

|  |  |  |
| --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** |
| 1 | Phan Nguyễn Nhật Trâm | 22521501 |

|  |
| --- |
| **Điểm tự đánh giá** |
| **10** |

**ĐÁNH GIÁ KHÁC:**

|  |  |
| --- | --- |
| Tổng thời gian thực hiện | 2 ngày |
| Phân chia công việc |  |
| Ý kiến *(nếu có)*  + Khó khăn  + Đề xuất, kiến nghị |  |

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

MỤC LỤC

[**A.** BÁO CÁO CHI TIẾT 2](#_Toc182693188)

[1. EVABS 2](#_Toc182693189)

[a. Level 1: 2](#_Toc182693190)

[b. Level 2: 3](#_Toc182693191)

[c. Level 3: 4](#_Toc182693192)

[d. Level 4: 4](#_Toc182693193)

[e. Level 5: 5](#_Toc182693194)

[f. Level 6: 7](#_Toc182693195)

[g. Level 7: 8](#_Toc182693196)

[h. Level 8: 10](#_Toc182693197)

[i. Level 9 12](#_Toc182693198)

[j. Level 10 13](#_Toc182693199)

[k. Level 11 14](#_Toc182693200)

[l. Level 12 17](#_Toc182693201)

[2. Droid 19](#_Toc182693202)

[a. Nhật ký droid đã đi đâu. Bạn có thể tìm thấy tại: one.apk. 19](#_Toc182693203)

[b. Tìm kiếm và lấy flag. Bạn có thể tìm thấy tại: two.apk. 20](#_Toc182693204)

[c. Tìm kiếm và lấy flag. Bạn có thể tìm thấy tại: three.apk. 21](#_Toc182693205)

[d. Dịch ngược, vá lại tập tin và lấy cờ. Bạn có thể tìm thấy tại: four.apk. 22](#_Toc182693206)

[e. Dịch ngược, vá lại tập tin và lấy cờ. Bạn có thể tìm thấy tại: five.apk 24](#_Toc182693207)

# BÁO CÁO CHI TIẾT

## EVABS

### Level 1:

Ý tưởng:

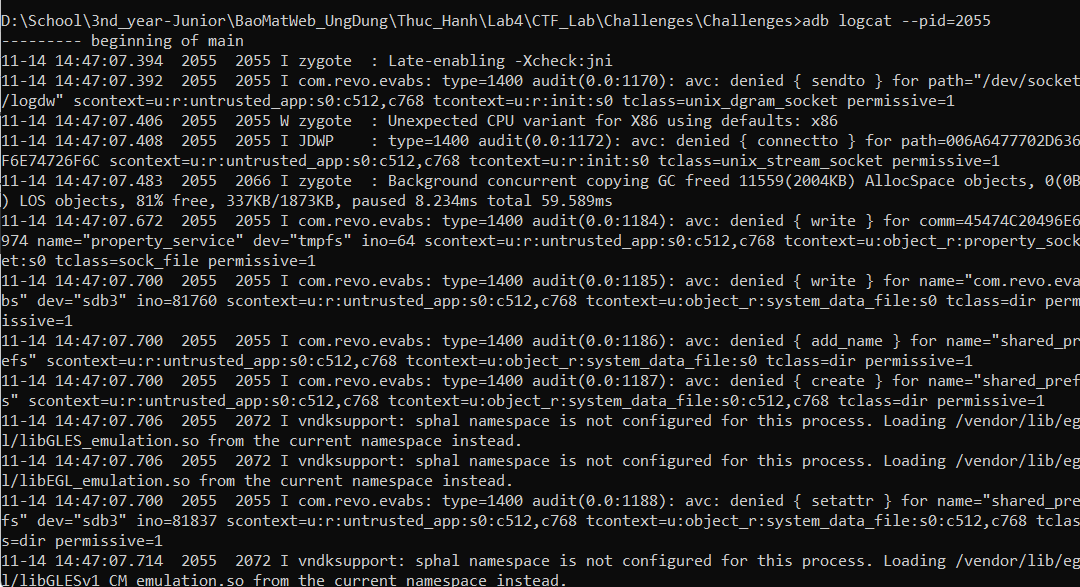
Dựa theo gợi ý ta sẽ xem log ứng dụng trên android bằng ADB

Thực hiện:

1. Tìm PID của ứng dụng



1. PID ứng dụng là 2055
2. Đọc log của ứng dụng bằng adb logcat



1. Logcat ứng dụng có PID 2055
2. Click vào nút “Log the key” và quay lại xem log



1. Flag level 1

### Level 2:

Ý tưởng

* Dựa vào gợi ý ta sẽ truy cập thư mục lưu trữ tài nguyên của ứng dụng

Thực hiện

1. Giải nén và dịch ngược tệp APK của ứng dụng

A computer screen with text on it

Description automatically generated

1. Hoàn tất giải nén và dịch ngược EVABSv5.apk
2. Truy cập thư mục asset

A screenshot of a computer

Description automatically generated

1. Kết quả flag level 2

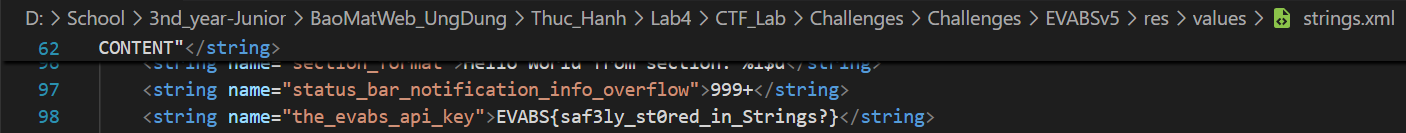
### Level 3:

Ý tưởng

Tìm kiếm file strings.xml

Thực hiện

1. Truy cập vào res/values của thư mục EVABSv5 vừa được tạo ở level 2 để xem file string.xml



1. Kết quả flag level 3

### Level 4:

Ý tưởng

Một tool đã được giấu trong resouce của ứng dụng. Dự đoán sẽ giấu dưới dạng **link** giúp tải công cụ

Thực hiện

1. Tìm kiếm lần lượt trong thư mục res

A screenshot of a computer

Description automatically generated

1. Kết quả flag level 4 (path raw/link.txt)

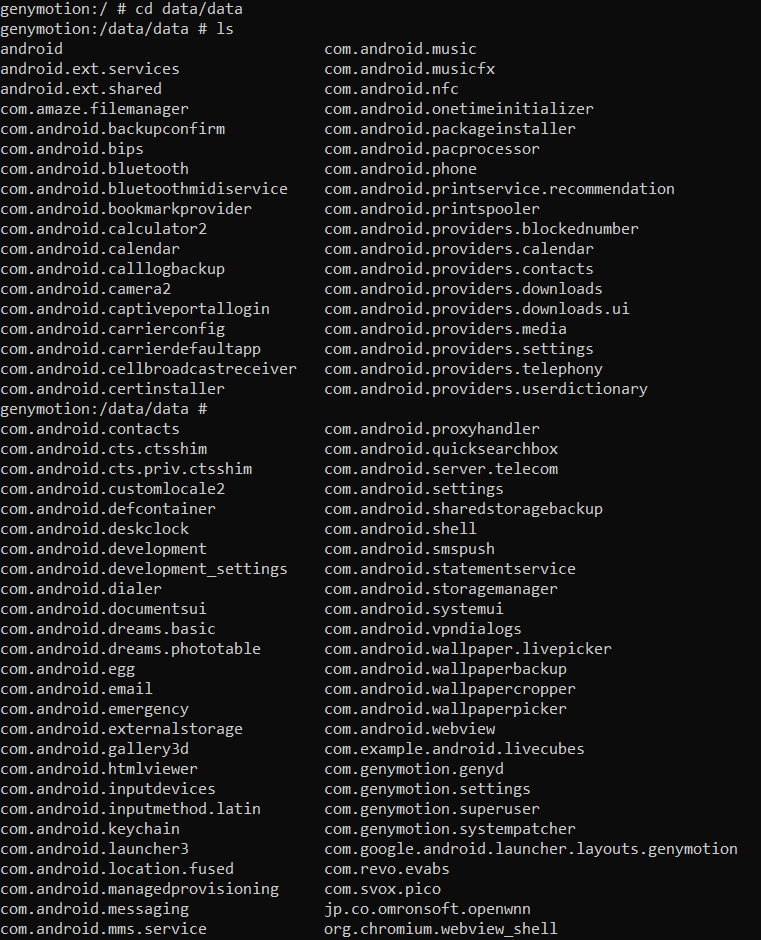
### Level 5:

Ý tưởng:

Xem file được lưu trữ trong /data/data/… của ứng dụng

Thực hiện

1. Mở ADB Shell
2. Đi đến đường dẫn data/data



1. Phát hiện tên gói ứng dụng com.revo.evabs
2. Đi đến com.revo.evabs và liệt kê các tập tin của các thư mục con

A computer screen with white text

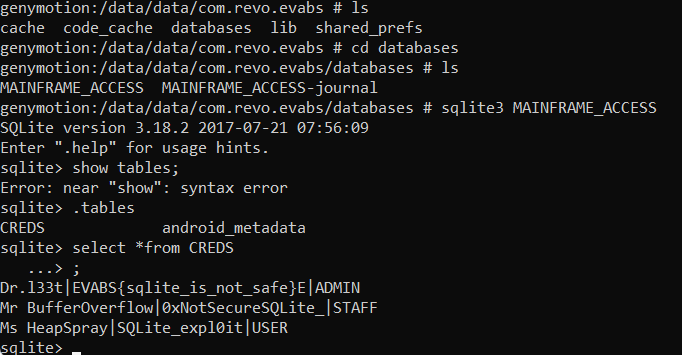
Description automatically generated

1. Kết quả flag level 5

### Level 6:

Thực hiện

1. Truy cập /data/data/com.revo.evabs để tìm database
2. Mở cơ sở dữ liệu SQLite với database MAINFRAME\_ACCESS
3. Truy vấn tìm thông tin



1. Kết quả lưu trữ trong database

A screenshot of a chat

Description automatically generated

1. Thông tin đăng nhập

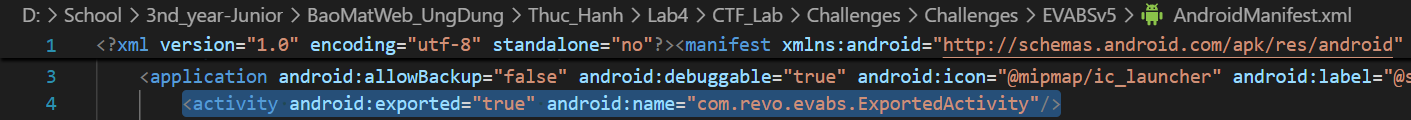
### Level 7:

Ý tưởng

* Tìm Activity được lưu giữ trong file AndroidManifest.xml
* Tìm thông tin về lỗ hổng, cách khai thác thuộc tính của Activity tìm được

Thực hiện

1. Tìm thông tin trong file AndroidManifest.xml



1. Ativity Export với thuộc tính android:exported=”true”
2. Tìm hiểu về android:exported=”true”

A close-up of a white background

Description automatically generated

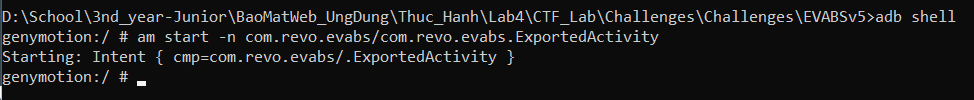
1. Thông tin về thuộc tính android:exported

* Nếu thuộc tính này có giá trị true thì activity đó có thể bị kích hoạt bởi các ứng dụng khác => chúng ta có thể khởi động activity này bằng adb

A screenshot of a computer

Description automatically generated

1. Cách khởi động Activity bằng adb
2. Khởi động Activity



1. Khởi động Activity bằng adb

A screenshot of a computer

Description automatically generated

1. Kết quả flag level 7

### Level 8:

Ý tưởng

Đọc code tìm kiếm ham Decode để giải mã

Thực hiện

1. Dùng Bytecode-Viewer-2.9.22.jar để phân tích và hiển thị code. Tìm thấy Decode.class
2. Giải mã base64

A screenshot of a computer

Description automatically generated

1. Decode



1. Kết quả flag level 8

### Level 9

Ý tưởng

Sửa code smali và recompile

Thực hiện

1. Dùng Bytecode-Viewer phân tích code, ta tìm được SmaliInject.class

A screenshot of a computer

Description automatically generated

1. Tín hiệu Lab\_OFF của code
2. Sửa Lab\_OFF thành Lab\_ON và recompile bằng apk tool (Lưu ý: tạo keystore và ký ứng dụng)

A computer screen with white text

Description automatically generated

1. Sửa code smali tại EVABSv5/smali/com/revo/evabs/ SmaliInject.smali

A black background with white text

Description automatically generated

1. Ký ứng dụng và cài đặt

Screens screenshot of a phone

Description automatically generated

**Sau**

**Trước**

1. Kết quả flag Level 9

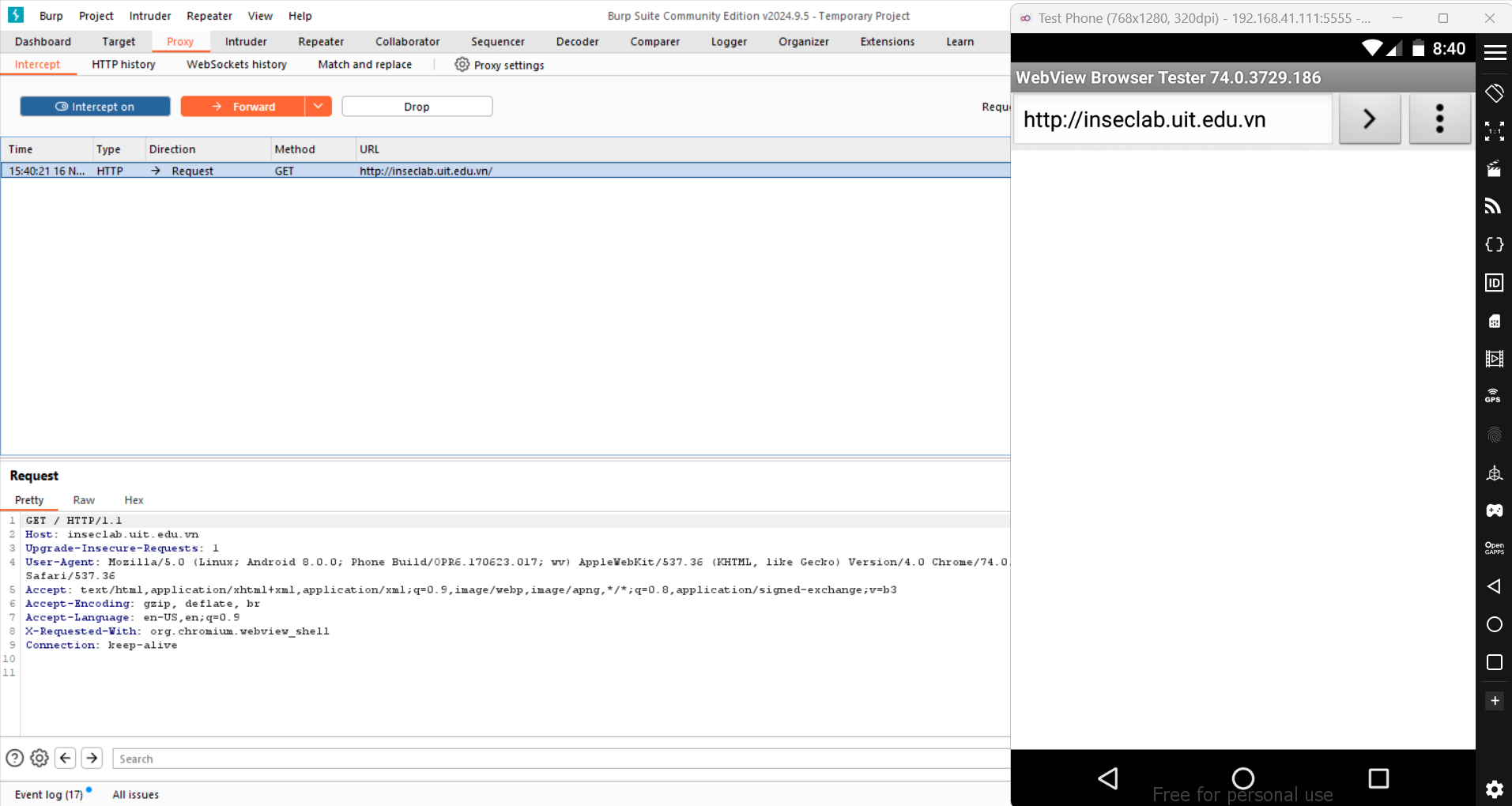
### Level 10

Ý tưởng

Dùng Burpsuite để bắt gói tin, nhận flag

Thực hiện

1. Cấu hình cho điện thoại ảo “proxy” thông qua Burp Suite



1. Thiết lập thành công

A screenshot of a computer

Description automatically generated

1. Kết quả nhận flag level 10

Nhận xét:

* Khi click vào Receive, BurpSuite không bắt được yêu cầu nào dù đã thiết lập thành công.
* Có vẻ đã có lỗi xảy ra bên phía gửi

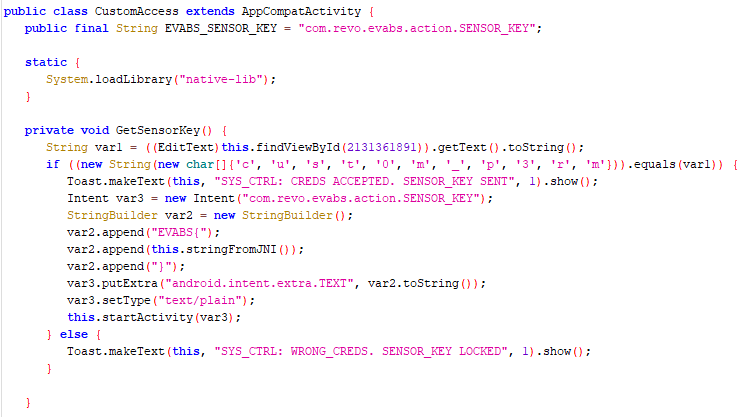
### Level 11

Ý tưởng

Sử dụng Frida để in flag

Thực hiện

1. Kiểm tra code để biết chuỗi nhập vào



1. Chuỗi input là “cust0m\_p3rm”
2. Nếu nhập đúng input (“cust0m\_p3rm”) thì flag sẽ được tạo ra vào truyền vào intent com.revo.evabs.action.SENSOR\_KEY bằng hàm putExtra() => hook hàm putExtra()

import frida

import sys

import time

device = frida.get\_usb\_device()

pid = device.spawn("com.revo.evabs")

device.resume(pid)

time.sleep(1)

session = device.attach(pid)

hook\_script = """

Java.perform(function () {

console.log("Hooking android.content.Intent.putExtra...");

var Intent = Java.use("android.content.Intent");

Intent.putExtra.overload('java.lang.String', 'java.lang.String').implementation = function (key, value) {

console.log("[\*] putExtra called with key: " + key + ", value: " + value);

if (key === "android.intent.extra.TEXT") {

console.log("[+] Flag detected: " + value);

}

return this.putExtra(key, value);

};

});

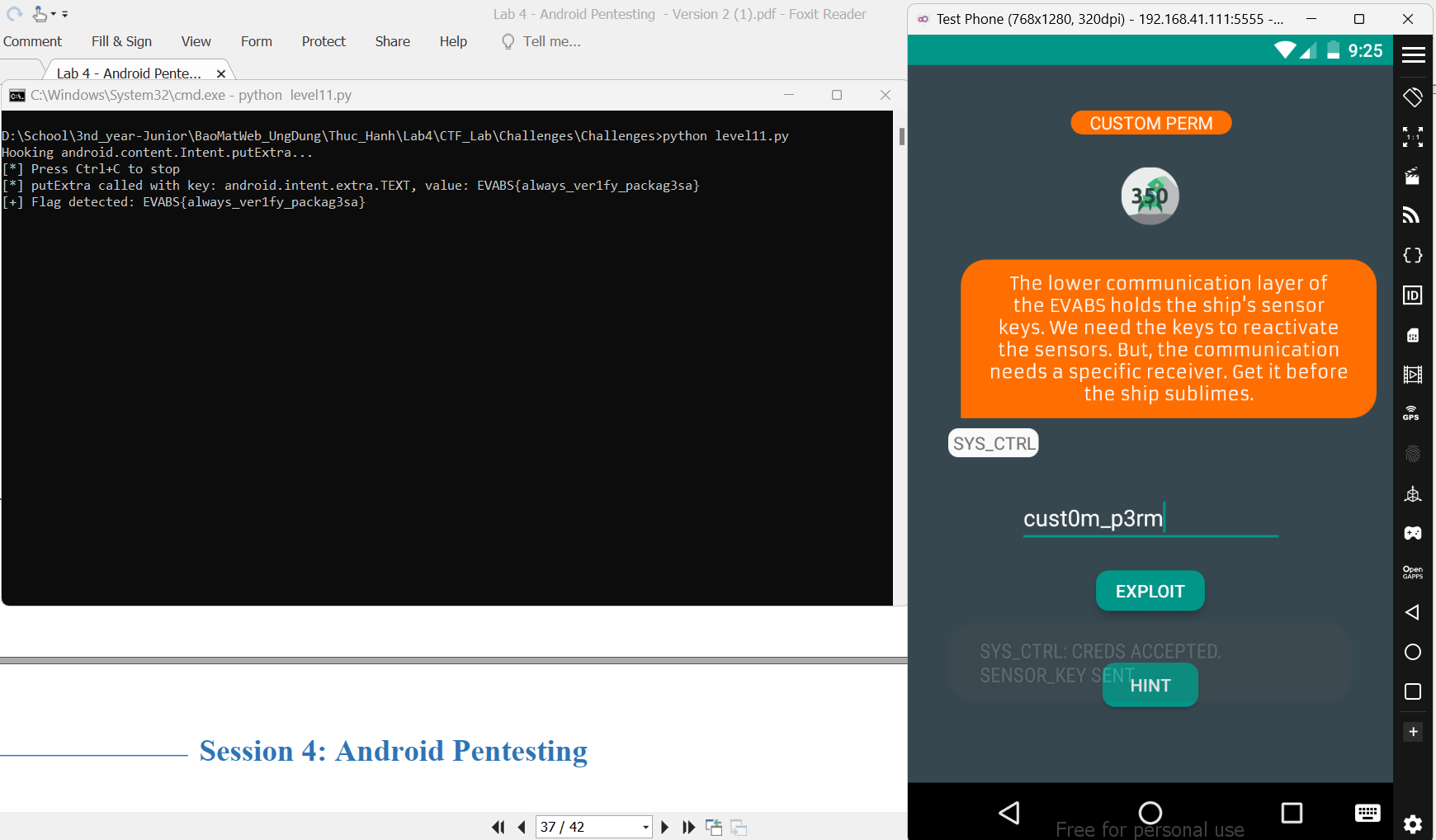
"""

script = session.create\_script(hook\_script)

script.load()

print("[\*] Press Ctrl+C to stop")

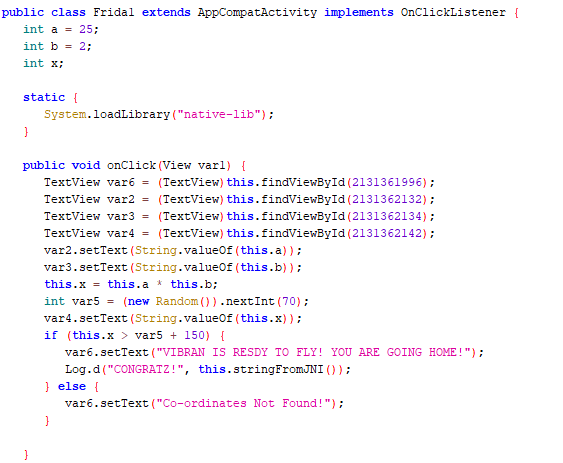
sys.stdin.read()



1. Kết quả flag level 11

### Level 12

Ý tưởng và giải thích



1. Code level 12

* Flag sẽ được in ra nếu a \* b > var 5 + 150 (với var 5 là số ngẫu nhiên từ 1 đến 70). Vì a, b không đổi nên x luôn là 50, chỉ cần dùng Frida để sửa var5 bằng -200

Thực hiện

import frida

import sys

import time

device = frida.get\_usb\_device()

pid = device.spawn(["com.revo.evabs"])

device.resume(pid)

time.sleep(1)

session = device.attach(pid)

hook\_script = """

Java.perform(function () {

    console.log("Hooking java.util.Random.nextInt");

    var random = Java.use("java.util.Random");

    random.nextInt.overload("int").implementation = function(var\_1) {

        return -200;

    };

});

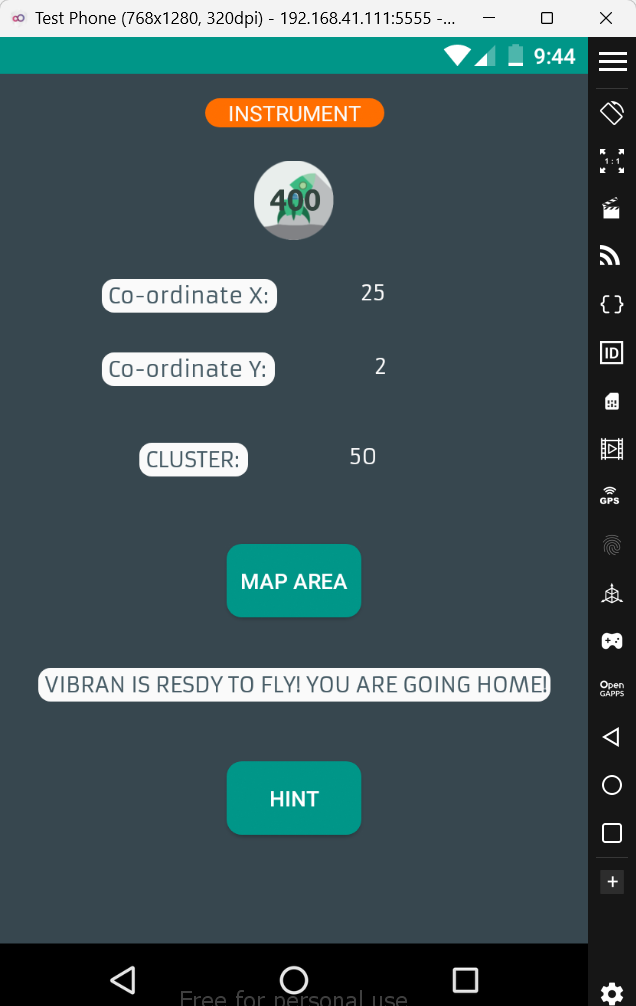
"""

script = session.create\_script(hook\_script)

script.load()

print("[\*] Press Ctrl+C to stop")

sys.stdin.read()



1. Kết quả hook



1. Kết quả flag level 12 (xem bằng logcat tương tự level 1)

## Droid

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

Ý tưởng

Sử dụng logcat để quan sát nhật ký

Thực hiện

1. Tìm pid của picoCTF

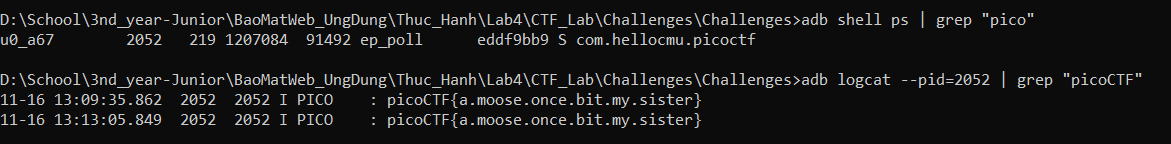
adb shell ps | grep “pico”

1. Xem nhật ký bằng logcat

adb logcat --pid=2052 | grep “picoCTF”

1. Click vào button của ứng dụng

Kết quả



1. Kết quả flag one.apk

### Tìm kiếm và lấy flag. Bạn có thể tìm thấy tại: two.apk.

Ý tưởng và thực hiện

* Vì đề chỉ yêu cầu lấy flag, ta không có gợi ý gì nên sẽ bắt đầu bằng việc phân tích code với Bytecode-Viewer

A black text on a white background

Description automatically generated

1. Code ở đường dẫn com/hellocmu/picoCTF/MainActivity.class

* Hàm getFlag được định nghĩa trong lớp FlagstaffHill sẽ cung cấp flag

A close up of text

Description automatically generated

1. Code ở đường dẫn com/hellocmu/picoCTF/FlagstaffHill.class

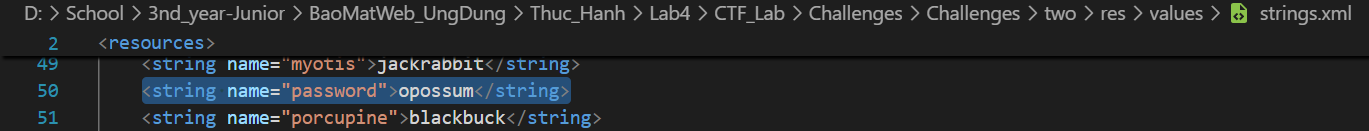
* Ta thấy, getFlag sẽ trả về flag nếu bằng với chuỗi nhập input, không thì sẽ in ra NOPE

A screenshot of a computer

Description automatically generated

1. Kết quả nhập chuỗi sai

* Để tìm chuỗi, ta sẽ thử decompile ứng dụng và vào two/res/values/strings.xml vì file này chứa các chuỗi được sử dụng trong ứng dụng



1. Chuỗi nghi ngờ password

Kết quả

A screenshot of a computer

Description automatically generated

1. Kết quả flag two.apk

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

Ý tưởng và thực hiện

* Tương tự two.apk, ta cũng sẽ đọc code three.apk bằng Bytecode-Viewer

A black text on a white background

Description automatically generated

1. Code ở đường dẫn com/hellocmu/picoCTF/MainActivity.class

A screenshot of a computer code

Description automatically generated

1. Code ở đường dẫn com/hellocmu/picoCTF/FlagstaffHill.class của three.apk

* Phân tích các biến ta có:
  + var2 = 0
  + var3 = 1
  + var4 = 2
  + var5 = 5
* Cụ thể dòng code return

return var0.equals("".concat(var6[var5]).concat(".").concat(var6[var3]).concat(".").concat(var6[var2]).concat(".").concat(var6[var5 + var2 - var3]).concat(".").concat(var6[3]).concat(".").concat(var6[var4]))

* Phân tích dòng return:
  + var6[var5] = var6[5] = “dismass”
  + var6[var3] = var6[1] = “ogg”
  + var6[var2] = var6[0] = “weatherwax”
  + var6[var5 + var2 - var3] = var6[4] = “aching”
  + var6[3] = “nitt”
  + var6[var4] = var6[2] = “galick”
  + Các chuỗi phân cách bằng dấu “.”
* Nếu nhập chuỗi **“dismass.ogg.weatherwax.aching.nitt.garlick”** thì sẽ nhận được flag

Kết quả

A screenshot of a computer

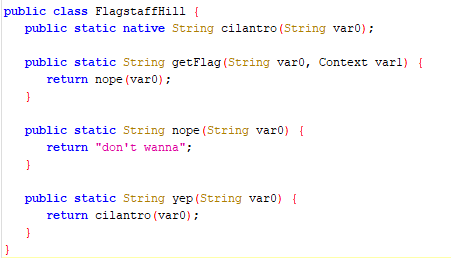
Description automatically generated

1. Kết quả three.apk

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

Ý tưởng và thực hiện

* Phân tích code four.apk bằng Bytecode-Viewer



1. Code Java ở đường dẫn com/hellocmu/picoCTF/FlagstaffHill.class của four.apk

A screenshot of a computer program

Description automatically generated

1. Code Smali ở đường dẫn com/hellocmu/picoCTF/FlagstaffHill.class của four.apk

* Ta thấy hàm getFlag hiện tại đang trả về chuỗi **“don’t wanna”**, vì vậy sau khi decompile four.apk chúng ta cần sửa code smali từ **“nope”** thành **“yep**” ở hàm getFlag

A screenshot of a computer program

Description automatically generated

1. Code Smali đã sửa nope sang yep

* Recompile, ký và cài đặt ứng dụng

A computer screen with white text

Description automatically generated

1. Recompile, ký và cài đặt ứng dụng (đặt tên là fourv2.apk)

Kết quả

A screenshot of a computer

Description automatically generated

1. Kết quả flag four.apk

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

Ý tưởng và thực hiện

* Phân tích code bằng Bytecode-Viewer

A white background with colorful text

Description automatically generated with medium confidence

1. Code Java ở đường dẫn com/hellocmu/picoCTF/FlagstaffHill.class của five.apk

* Ta thấy nếu nhập đúng chuỗi input thì sẽ in ra chuỗi “call it”, vậy ta cần làm hai việc là tìm chuỗi input và sửa dòng code return để thay vì in “call it” nó sẽ gọi cardamom để in flag
* Đầu tiên là chuỗi input, chuỗi input là “alphabetsoup”
* var2 ban đầu là “aaa”:
  + Ký tự đầu tiên tăng 4: ‘a’ + 4 = 'e'
  + Ký tự thứ hai tăng 19: ‘a’ + 19 = 't'
  + Ký tự thứ ba tăng 18: ‘a’ + 18 = 's'

=> Kết quả: var2 = “ets”

* var3 ban đầu là “aaa”:
  + Ký tự đầu tiên tăng 7: ‘a’ + 7 = ‘h’
  + Ký tự thứ hai không thay đổi: ‘a’ + 0 = ‘a’
  + Ký tự thứ ba tăng 1: ‘a’ + 1 = ‘b’

=> Kết quả: var3 = "hab"

* var5 ban đầu là “aaa”:
  + Ký tự đầu tiên không thay đổi: ‘a’ + 0 = ‘a’
  + Ký tự thứ hai tăng 11: ‘a’ + 11 = ‘l’
  + Ký tự thứ ba tăng 15: ‘a’ + 15 = ‘p’

=> Kết quả: var5 = "alp"

* Var4 ban đầu là “aaa”:
  + Ký tự đầu tiên tăng 14: ‘a’ + 14 = ‘o’
  + Ký tự thứ hai tăng 20: ‘a’ + 20 = ‘u’
  + Ký tự thứ ba tăng 15: ‘a’ + 15 = ‘p’

=> Kết quả: var4 = "oup"

A screenshot of a phone

Description automatically generated

1. Chuỗi input đúng

* Sau khi decompile five.apk, sửa code smali



1. Code Smali cần sửa ở đường dẫn com/hellocmu/picoCTF/FlagstaffHill.class của five.apk

A screenshot of a computer

Description automatically generated

1. Code smali đã sửa

* Recompile, ký và cài đặt ứng dụng

A computer screen with text on it

Description automatically generated

1. Recompile, ký và cài đặt ứng dụng (đặt tên là fivev2.apk)

Kết quả

A screenshot of a chat

Description automatically generated

1. Kết quả flag five.apk