BÁO CÁO THỰC HÀNH

**Môn học: Pháp chứng kỹ thuật số**

**Kỳ báo cáo: Buổi 05 (Session 05)**

**Tên chủ đề: Mobiles Forensics**

*GVHD: Đoàn Minh Trung*

*Ngày báo cáo: 07/05/2024*

**Nhóm: 07**

1. **THÔNG TIN CHUNG:**

*(Liệt kê tất cả các thành viên trong nhóm)*

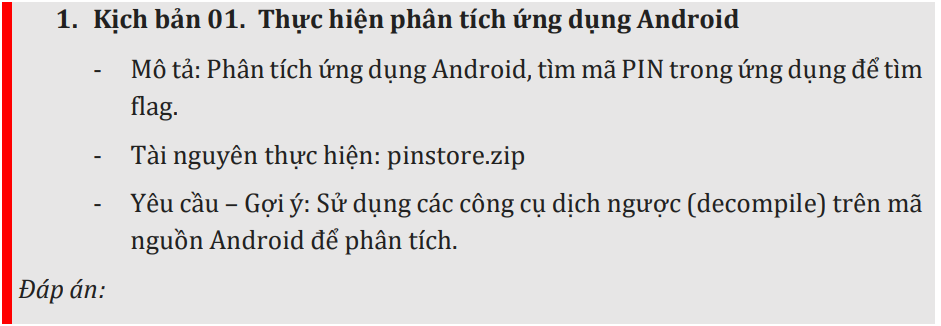
Lớp: NT334.O21.ATCL.1

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Nguyễn Đình Bảo Long | 21522303 | [21522303@gm.uit.edu.vn](mailto:21522303@gm.uit.edu.vn) |
| 2 | Nguyễn Tấn Phát | 21522447 | [21522447@gm.uit.edu.vn](mailto:21522447@gm.uit.edu.vn) |
| 3 | Ngô Minh Thiên | 21522623 | [21522623@gm.uit.edu.vn](mailto:21522623@gm.uit.edu.vn) |
| 4 | Đào Vĩnh Thịnh | 21522632 | [21522632@gm.uit.edu.vn](mailto:21522632@gm.uit.edu.vn) |

1. **NỘI DUNG THỰC HIỆN:[[1]](#footnote-1)**

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Kết quả tự đánh giá** |
| 1 | Kịch bản 01 | 100% |
| 2 | Kịch bản 02 | 100% |
| 3 | Kịch bản 03 | 100% |
| 4 | Kịch bản 04 | 100% |

**BÁO CÁO CHI TIẾT**



A screenshot of a computer

Description automatically generated

- Khi cài đặt thì nó yêu cầu ta nhập pin

- Nếu sau thì in ra “Incorrect Pin, try again”

- Ta dùng công cụ jadx để decompile file apk

A screen shot of a computer

Description automatically generated

Trong đường dẫn “/home/lixsong/Desktop/sources/pinlock/ctf/pinlock/com/pinstore/” ta thấy có các file dưới đây.

A screenshot of a computer

Description automatically generated

- Thường trong MainActitivy sẽ là chứa các Activity chính của chương trình

A screenshot of a computer

Description automatically generated

- Đoạn code trong MainActivity

- Ta có thể thấy

+ Chương trình khởi tạo biến để lấy thông tin pin ta nhập vào sau đó khởi tạo biến pinFromDB là null và biến hashOfEnteredPin là null

+ Ta có thể đoán là nó sẽ lấy mã pin người dùng nhập và mã PIN từ database. Sau đó nó sử dụng lớp CryptoUilitiese để băm mã PIN nhập vào và so sánh với mã PIN từ database

+ Nếu 2 mã PIN khớp thì nó sẽ chuyển qua màn hình SecretDisplay còn nếu không khớp thì sẽ có thông báo được hiển thị

- Vậy ta chỉ cần vào xem database của nó để xem mã hash là được

A screen shot of a computer code

Description automatically generated

- Trong CryptoUtilities.Class ta thấy nó sử dụng SHA-1 để hash mã PIN của ta

- Khi decompile các file và xem trong các thư mục ta thấy trong thư mục “/resources/assets/” có 1 file tên là pinlock.db có vẻ như đây sẽ là file database của ta

A screenshot of a computer

Description automatically generated

- Dùng sqlite3 để xem các tables trong databse

- Xem qua các tables thì tables pinDB có vẻ là khớp với SHA-1 của ta

A screenshot of a computer

Description automatically generated

Thử dùng tool decode SHA1 online thì có được kết quả là 7498 <https://www.dcode.fr/sha1-hash>



Dùng mã pin 7498 login thành công ứng dụng pinstore

A screenshot of a computer

Description automatically generated

- Kết quả khi ta nhập vào mã PIN đúng

- Nhưng có vẻ là chưa ra được flag của ta

A screenshot of a computer program

Description automatically generated

- Ta xem code của secretdisplay

- Ta thấy sau khi ta nhập mã pin vào thì nó sẽ lấy giá trị đó và dùng CryptoUtilities để thực hiện mã hóa và giải mãm bằng cách truyền phiên bản “v1” và “pin” vào

- Sau đó nó sử dụng hàm fetchSecret() để lấy chuỗi bí mật từ database sau đó chuỗi này được decrypt và kết quả giải mã được sẽ hiển thị lên giao diện

A computer screen shot of code

Description automatically generated

- Đây là hàm fetchSecret() nhưng ta nhập nó lại không phải là flag

A computer screen shot of a program code

Description automatically generated

- Ta xem hàm getkey trong CyptoUtilities class

- Phương thức getKey(String version) trong lớp CryptoUtilities được sử dụng để tạo SecretKeySpec dựa trên phiên bản và pin.

- Nếu phiên bản là "v1", phương thức tạo một SecretKeySpec từ một chuỗi cố định "t0ps3kr3tk3y" và sử dụng thuật toán mã hóa AES.

- Nếu phiên bản không phải "v1", phương thức sử dụng thuật toán PBKDF2 và HMAC-SHA1 để tạo một SecretKey từ pin và salt. Sau đó, tạo một SecretKeySpec từ khóa đã được mã hóa và sử dụng thuật toán mã hóa AES.

- Kết quả là SecretKeySpec tạo ra dựa trên phiên bản và pin được sử dụng cho việc mã hóa và giải mã dữ liệu.

➔ Ý tưởng là ta sẽ sử dụng pin và chuỗi secret từ secretsDBv2 là sẽ có flag của ta

Ta có đoạn code khai thác

A screenshot of a computer program

Description automatically generated

Dưới đây là ý nghĩa của từng phần trong đoạn code:

1. Khai báo các thư viện và các ngoại lệ có thể xảy ra trong quá trình mã hóa và giải mã.

2. Khai báo chuỗi `pin` là một mật khẩu đơn giản, có thể sử dụng để tạo ra khóa mã hóa cho dữ liệu.

3. `secret1` và `secret2` là các chuỗi đã được mã hóa lấy trong pinlock.db

4. Trong phần `try`, đoạn code thực hiện các bước sau:

- Giải mã `secret1` sử dụng một khóa cụ thể được tạo ra từ chuỗi "t0ps3kr3tk3y" thông qua thuật toán SHA-1. Sau đó, dữ liệu được giải mã bằng thuật toán AES.

- Tạo một khóa mới từ mật khẩu (`pin`) bằng cách sử dụng phương pháp PBKDF2 với HMAC-SHA1 và một salt cố định là "SampleSalt".

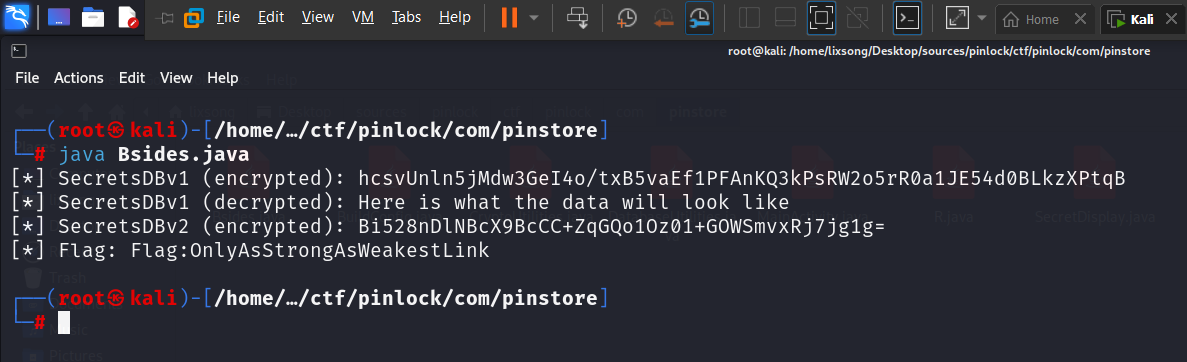
- Sử dụng khóa mới để giải mã `secret2`.

- In ra kết quả giải mã của `secret1` và `secret2`.

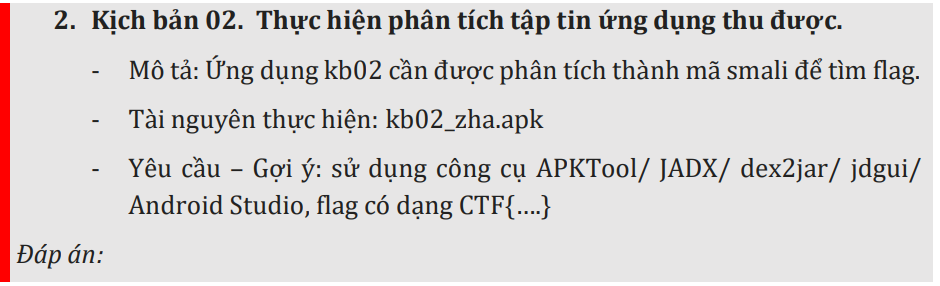
5. Trong trường hợp xảy ra các ngoại lệ như không thể tìm thấy thuật toán hoặc khóa không hợp lệ, chương trình sẽ in ra thông báo lỗi.

Tóm lại, đoạn code này giải mã hai chuỗi đã được mã hóa bằng AES, một bằng một khóa được tạo ra từ một chuỗi cố định và một bằng một mật khẩu được nhập vào. Điều này giúp bảo vệ dữ liệu nhạy cảm khỏi việc truy cập trái phép.

Kết quả :



Flag: OnlyAsStrongAsWeakestLink



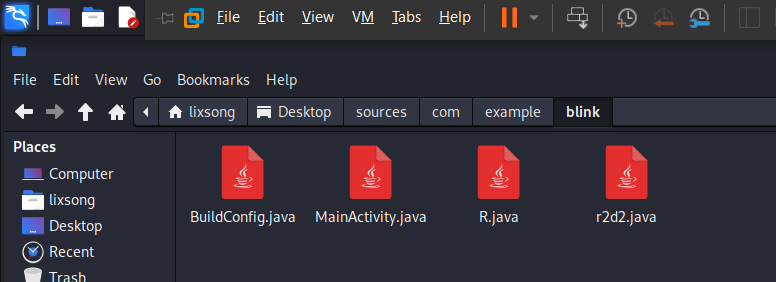
A screenshot of a video game

Description automatically generated

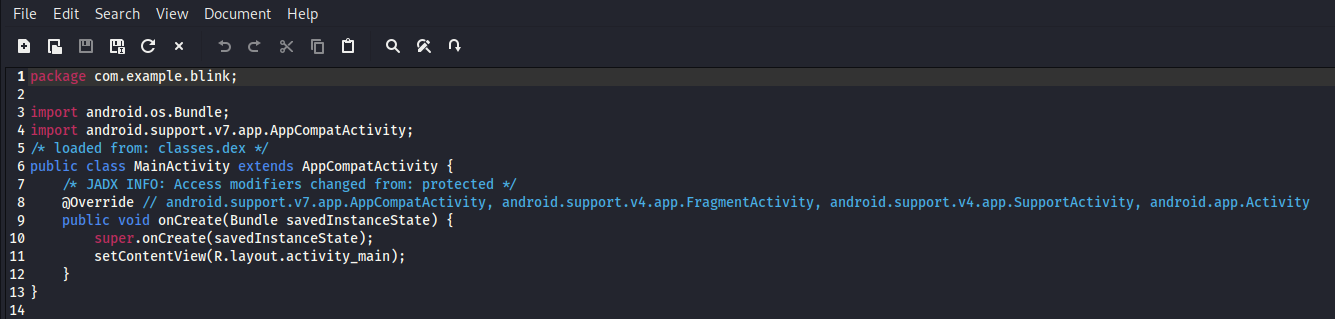
- Khi cài đặt vào truy cập vào ứng dụng ta được như hình trên

- Dùng jadx để decompile file apk

- Ta thấy có 4 file.



Kiểm tra code MainActivity.java thì không thấy có gì đặc biệt



Kiểm tra code r2d2.java thì thấy có đoạn mã base64 khá dài.

Ta thấy đoạn mã base64 này có dạng image jpeg.

A screenshot of a computer screen

Description automatically generated

Thực hiện chuyển base 64 thành hình bằng lệnh: echo “/9j/4AA...” > base64 –d flag.png

A screenshot of a computer screen

Description automatically generated

Thì ta lấy được flag

A screenshot of a computer

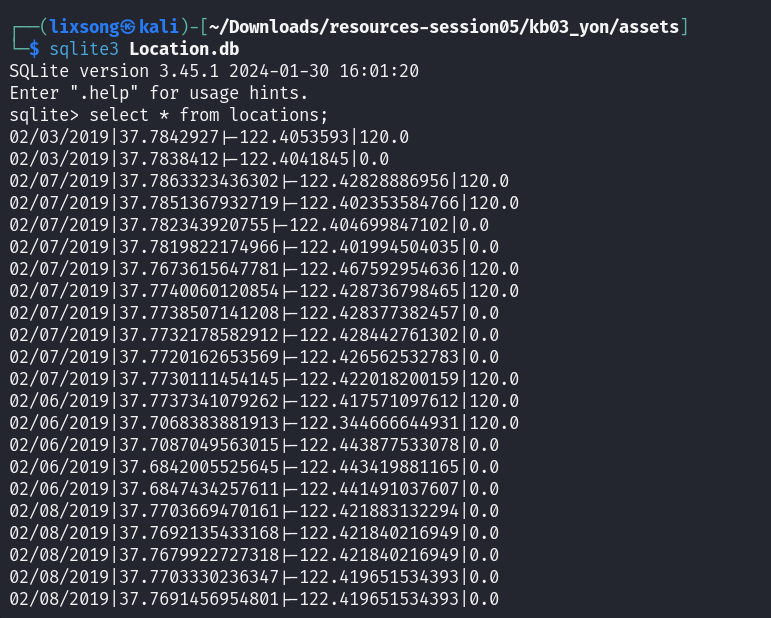
Description automatically generated

Flag: CTF{PUCKMAN}

A close-up of a message

Description automatically generated

Đầu tiên ta dùng apktool để decompile và mở file assets để đọc Locations.db



Xem thông tin bên trong thì ta thấy được thông tin ngày tháng, toạ độ X, toạ độ Y và màu 120 và 0

Thực hiện code để đọc data từ file Location và kết hợp api của google maps để thực hiện visualize lên google maps

A screenshot of a computer program

Description automatically generated

Thực hiện chạy code thì tạo ra file map.html

A black background with white text

Description automatically generated

Mở file map.html

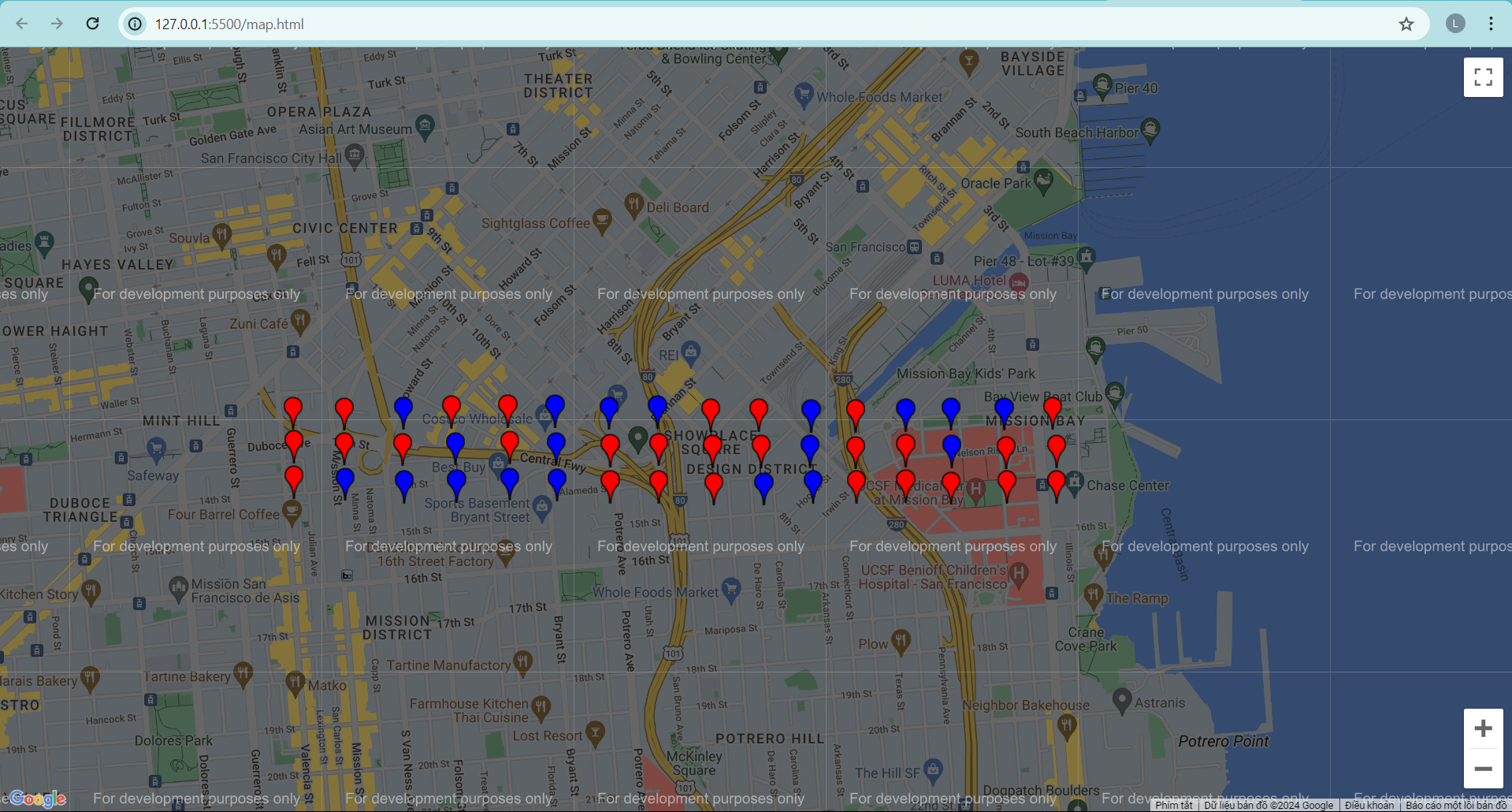
A map with red and blue pins

Description automatically generated

- Khi ta load các điểm lên thì ta được các điểm như trên

- Tới đây ta vẫn chưa sử dụng tới dữ kiện cột ngày trong database

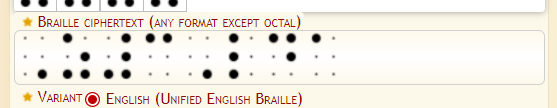
- Khi lọc các điểm theo ngày thì ngày “02/08/2019” cho ta các điểm khá là thú vị

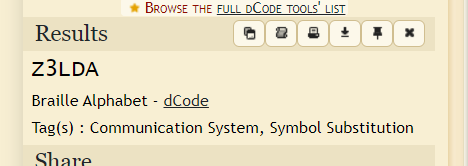


- Đây có vẻ như là “Braille”

- Ta có thể coi điểm màu đỏ là những chấm đen còn màu xanh là chấm trắng

Dùng <https://www.dcode.fr/braille-alphabet> để decode



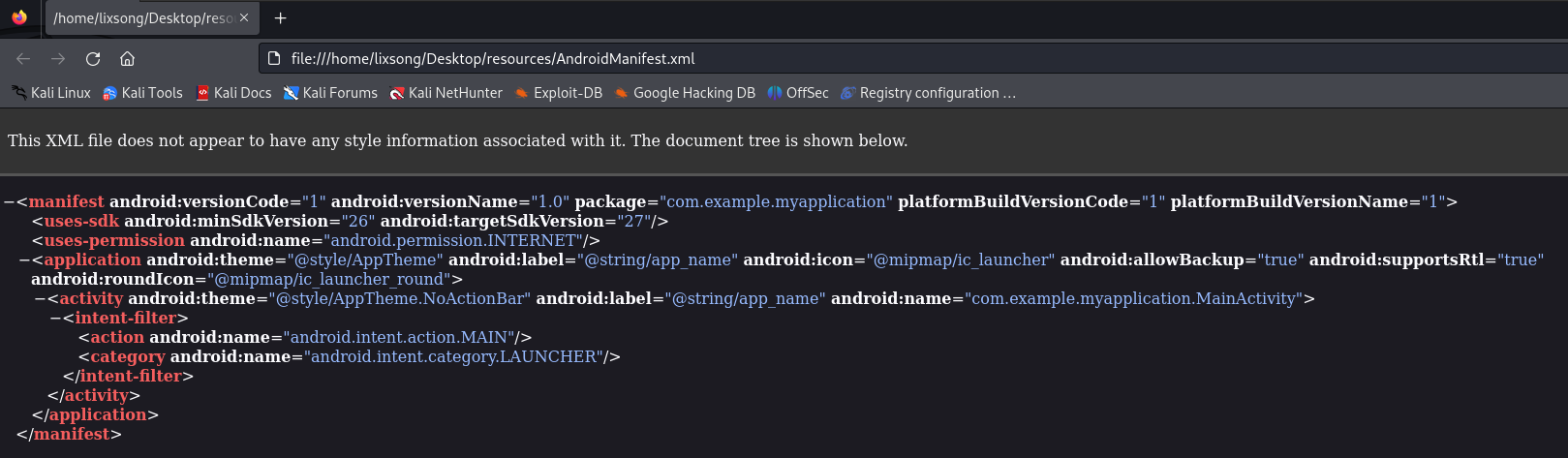


Flag: Z3LDA

A screenshot of a computer

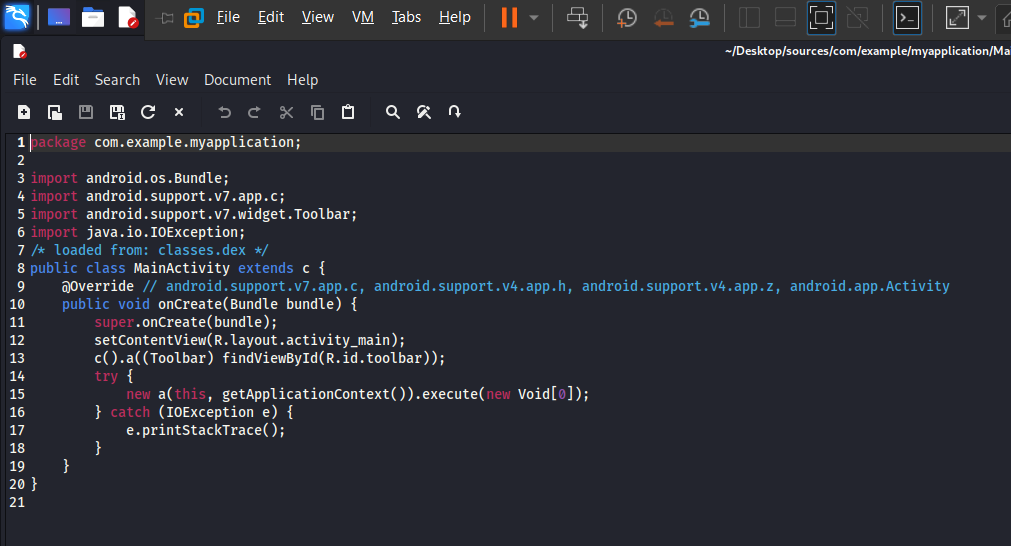
Description automatically generated

Dùng jadx decompile file apk và đọc file AndroidManifest.xml để check version



Check thấy androidsdk version = 26 hoặc 27 nghĩa là android 8 hoặc android 8.1

Ta thấy chương trình có activity là com.example.myapplication.MainActivity



Đọc sơ qua thì đoạn mã cố gắng thực thi 1 tác vụ bằng cách sử dụng lớp “a”, truyền hoạt động “this” và ngữ cảnh “getApplicationContext()” làm đối số.

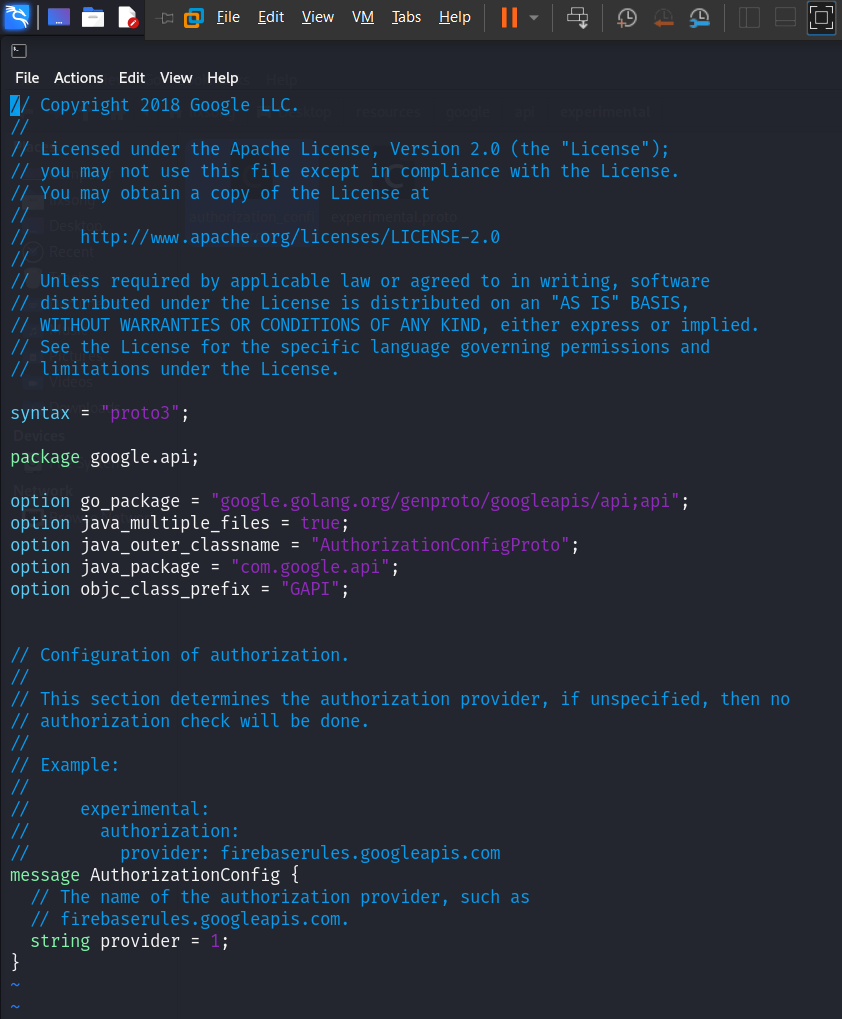
Sau đó dòng execute để khởi động thực thi tác vụ.

Ta thì thấy MainActivity.java gợi ý cho chúng ta về a.java, cái mà mở rộng AsyncTask, cho thấy quá trình sẽ diễn ra ở nền

A screen shot of a computer program

Description automatically generated

Tiếp tục ta sẽ thực hiện xem code thì ta thấy chương trình đang sử dụng api key của google thì đây là lab bị outdate nên dùng được nữa.



Sau khi phân tích code thì ta có attack vector

Bypass SSL Unpinning 🡪 Hook toString 🡪 Monitor toString

Thực hiện viết đoạn code để hooking.

A screen shot of a computer program

Description automatically generated

Đoạn code trên là một ví dụ về việc sử dụng Frida, một công cụ được sử dụng để hook và thay đổi hoạt động của ứng dụng chạy trên Android. Dưới đây là giải thích từng bước của đoạn code:

1. Step 1:

- Sử dụng `Java.perform()` để thực hiện một hàm callback trên các đối tượng Java.

- Lấy ra các class cần hook: `ArrayList` và `TrustManagerImpl` từ gói `java.util` và `com.android.org.conscrypt` tương ứng.

- Ghi đè vào phương thức `checkTrustedRecursive` của `TrustManagerImpl`. Trong hàm ghi đè này, một đối tượng mới của `ArrayList` được tạo và trả về.

2. Step 2:

- In ra thông báo "Hooking Java".

- Lấy ra class `StringBuilder` từ `java.lang`.

- Ghi đè vào hàm khởi tạo của `StringBuilder` nhận một đối số là một chuỗi. Trong hàm ghi đè này, một chuỗi rỗng và một đối tượng `StringBuilder` mới được khởi tạo. Thông điệp "new StringBuilder()" cùng với đối số được in ra.

- In ra thông báo "Hooking new StringBuilder(java.lang.String)".

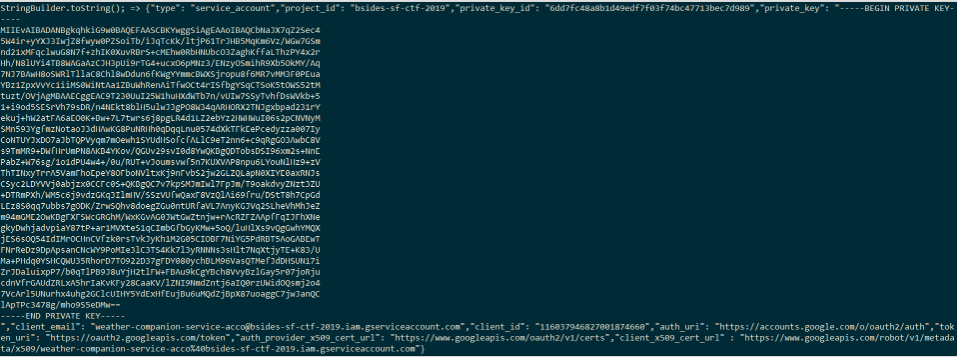
3. Step 3:

- Ghi đè vào phương thức `toString()` của `StringBuilder`. Trong hàm ghi đè này, giá trị của chuỗi `StringBuilder` được lấy ra và in ra thông điệp "StringBuilder.toString(); => [giá trị của chuỗi]".

- In ra thông báo "Hooking StringBuilder.toString() hooked".

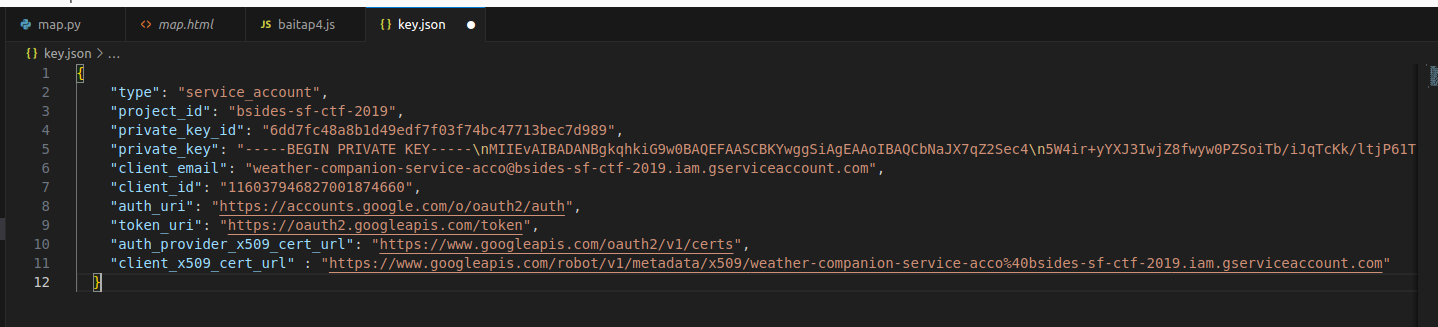
Như vậy, đoạn code trên đang thực hiện việc ghi đè vào các phương thức của các class trong Java như `ArrayList` và `StringBuilder`, và sử dụng Frida để theo dõi và in ra các thông điệp khi các phương thức này được gọi trong ứng dụng Android.

Thực hiện chạy file frida -U -f com.example.myapplication -l /home/lixsong/Desktop/baitap4.js

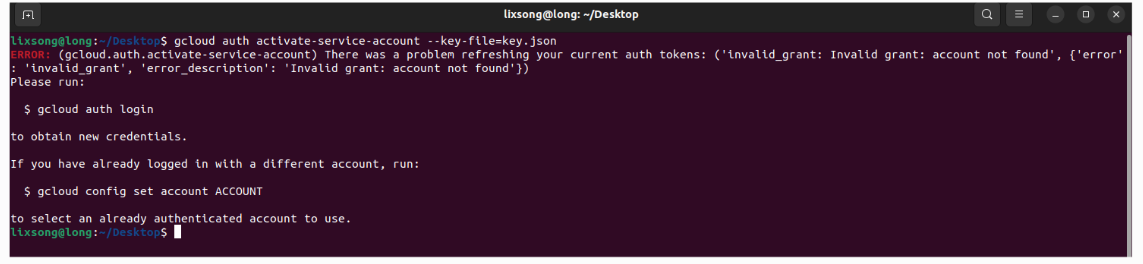


- Chuỗi có định dạng JSON này là key được sử dụng để ủy quyền trong google cloud storage

- Và file JSON sẽ có định dạng như sau, khi đã sắp xếp lại



Thực hiện cài đặt account theo file key.json này nhưng ko thành công có thể do bài này bị outdate rồi .



Sau khi tham khảo writeups trên mạng thì dưới đây là các bước nếu đăng nhập thành công.

A screen shot of a computer code

Description automatically generated



Flag: CTF{buck3t\_s3at5}

1. Ghi nội dung công việc, các kịch bản trong bài Thực hành, [↑](#footnote-ref-1)