6-FRIDA

1-Introduction

هي عبارة عن Dynamic tool بنستخدمها علشان نحلل التطبيق من غير ما نعمل decompile للتطبيق يعني مثلا عاوزين نغير في الكود والتطبيق شغال ب Frida بتسمح بكده يبقي كده بتستخدم في

- **reverse engineering**
- **Mobile penetration testing**

Frida contain 2 component

1- server ---> on phone (andorid)

2- client ----> on computer (ubuntu)

برتبط بقي بين server و client ياستخدام Python or JavaScript وبكده عرفنا ايه هي الاداة دلوقتي هنشوف ازاي هنزل الاداة

2- Install

1- Frida Client (install on ubuntu)

```
➜  ~ pip install frida-tools --break-system-packages
```

```
➜ ~ pwd
/home/ubuntu
➜ ~ cd .local/bin
➜ bin ls
alembic       f2py       frida-apk       frida-itrace  frida-ls-devices  frida-rm     numpy-config
androguard    flask      frida-compile   frida-join    frida-ps          frida-trace  pyftmerge
apkInspector  fonttools  frida-create    frida-kill    frida-pull        ipython      pyftsubset
cheroot       frida      frida-discover  frida-ls      frida-push        ipython3     ttx
➜ bin
```

2- for server install from github and the link of installation

https://github.com/frida/frida/releases/download/16.7.14/frida-server-16.7.14-android-x86_64.xz

after install extract it and upload to emulator on /data/local/bin --> because this path use to run binary file

```
➜  Downloads adb push frida-server-16.7\ \(2\).14-android-x86_64
/data/local/tmp
➜  cd /data/local/tmp
➜  chmod +x frida-server-16.7\ \(2\).14-android-x86_64
```

```
➜  su
➜  ./frida-server-16.7\ \(2\).14-android-x86_64
```

on ubuntu connect client with server

-U --> to connect to adb

```
➜  Downloads frida-ps -U
```

3- Hooking with Frida

اعتراض استدعاء دوال أو وظائف معينة في تطبيق قيد التشغيل، بحيث يمكن التعديل عليها أو مراقبة كيفية عملها في الوقت الفعلي.

مثال: اعتراض دالة `onClick` في تطبيق Android باستخدام Frida

لنفرض أنك تريد اعتراض دالة `onClick` في تطبيق Android عند الضغط على زر معين. إليك كيفية فعل ذلك باستخدام Frida:

1. تحديد الدالة المستهدفة

في هذا المثال، نحن نريد اعتراض الدالة `onClick` الخاصة بزر في التطبيق.

2. استخدام Frida للاعتراض على الدالة

هذا هو الكود باستخدام Frida لكتابة سكربت بلغة JavaScript لاعتراض دالة `onClick` في تطبيق Android:

`Java.perform()` is where we tell Frida to hook itself to the JVM, and it takes a function as a parameter, in this function we write our code.

```
Java.perform(function() {
    // get clas
    var MainActivity = Java.use("com.example.myapp.MainActivity"); // استبدل المستهدف بالـ class

    // اعتراض دالة onClick
    MainActivity.onClick.implementation = function(view) {
        console.log("Successful!"); // طباعة رسالة في الـ console

        // يمكنك هنا التعديل أو القيام بأي عملية قبل استدعاء الدالة الأصلية
        this.onClick(view); // استدعاء الدالة الأصلية بعد التعديل
    };
});
```

3. تشغيل السكربت باستخدام Frida

لتشغيل السكربت، تحتاج إلى توصيل جهازك أو محاكي Android، ثم استخدام Frida لحقن السكربت في التطبيق الذي تريد اختباره:

```
frida -U -f com.example.myapp -l hook_script.js --no-pause
```

دلوقتي ده كان شرح مثال يعني بيشرح ازاي هنتعامل مع Frida دلوقتي بقي تعالي نشوف امثلة تانية وده هيكون من https://frida.re/docs/javascript-api/ : Frida.re documentation javascript

java.perform --> useing onResume method

```
Java.perform(function(){
  const Activity = Java.use('android.app.Activity');
  Activity.onResume.implementation = function () {
    send('onResume() got called! Let\'s call the original implementation');
    this.onResume();
  };
});
```

دلوقتي هننزل التطبيق و نشوف ازاي نستخدم معاه Frida

```
adb install AES.apk

➜  ~ frida-ps -U |grep -i aes
4032  AES
```

open AES with frida but must to turn on firda server

```
➜  ~ frida -U AES
     ____
    / _  |   Frida 16.7.10 - A world-class dynamic instrumentation toolkit
   | (_| |
    > _  |   Commands:
   /_/ |_|       help      -> Displays the help system
   . . . .       object?   -> Display information about 'object'
   . . . .       exit/quit -> Exit
   . . . .
   . . . .   More info at https://frida.re/docs/home/
   . . . .
   . . . .   Connected to Android Emulator 5554 (id=emulator-5554)

[Android Emulator 5554::AES ]->
```

paste the code here

هنا OnResume method هتشتغل لما نخلي التطبيق run in background فدلوقتي هنخرج من التطبيق بس مش هنقفله خالص ونشوف هل method run ولا لا

- **`onResume`:**

  الكود بيستعمل Frida علشان يوصل للكلاس اللي اسمه **`Activity`** اللي هو مسؤول عن إدارة الـ Activity (الواجهة) في تطبيقات Android.

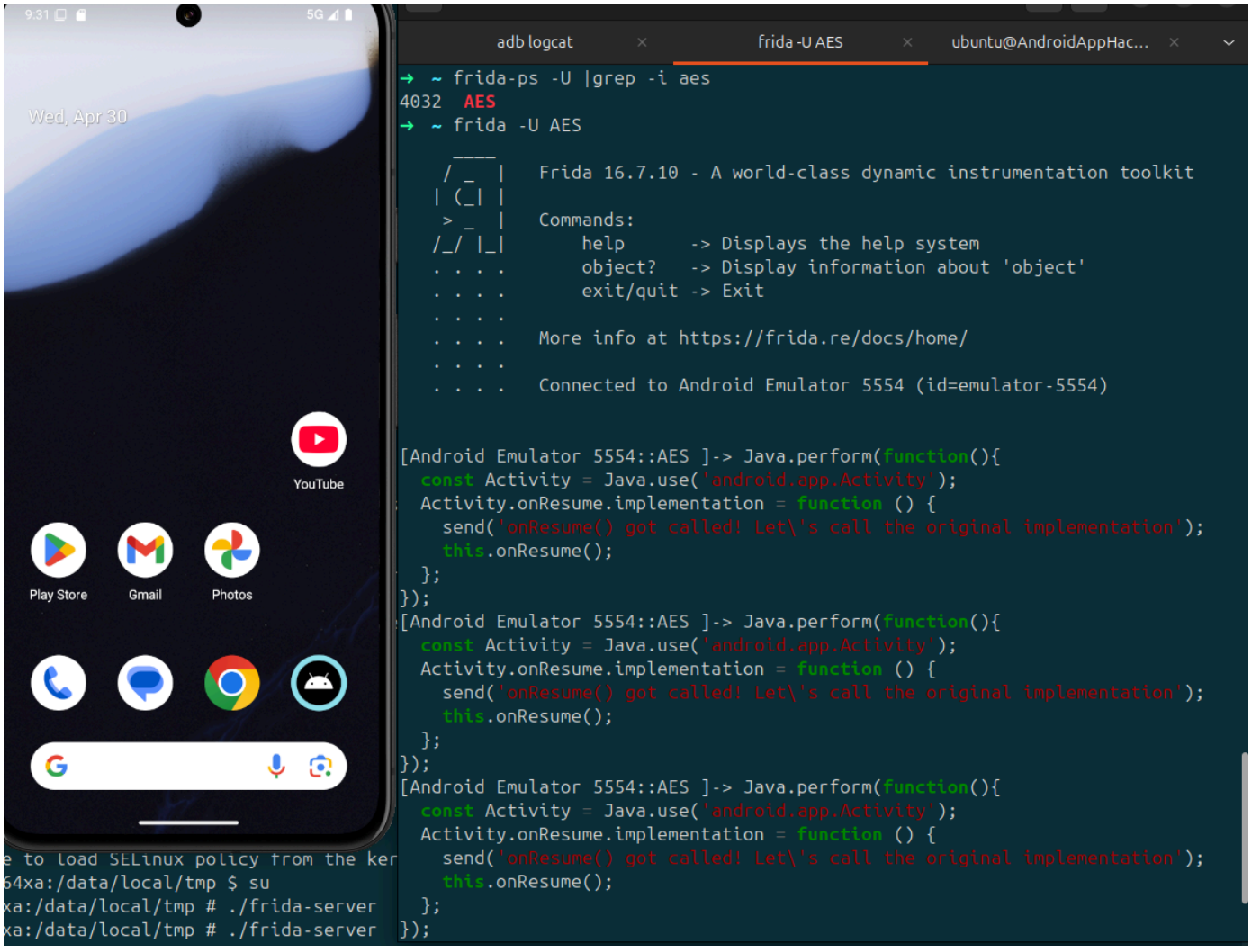
- **تغيير سلوك `onResume`:**

  لما الدالة **`onResume`** تتنفذ، الكود بيعرض رسالة (logs) بتقول إن الدالة اشتغلت:

  > **"onResume() got called! Let's call the original implementation"**

- **android.app.Activity --> class name**

Method run successfully



4- Dize Game : another example for frida with this app

دي عبارة عن تطبيق انتا بتحدد عدد النرد وععد الاوجه وهو بيطلع رقم احنا بقي عاوزين نعرف ايه هو الرقم ده

1- decompile app with jadx-gui

```
➜  frida jadx-gui Privacyfriendlydicer.apk
```

2- identify the method that handle the "dice" result

```
package org.secuso.privacyfriendlydicer.dicer;

import java.security.SecureRandom;

/* loaded from: classes.dex */
public class Dicer {
    private static final SecureRandom random = new SecureRandom();

    public int[] rollDice(int poolSize, int faceNum) {
        int[] dice = new int[poolSize];
        for (int i = 0; i < dice.length; i++) {
            dice[i] = random.nextInt(faceNum) + 1;
        }
        return dice;
    }
}
```

3- try to understand the implementation of it

1. تعريف الكلاس:

```
public class Dicer {
```

- هذا هو تعريف الكلاس واسمه `Dicer`.
- الكلاس يستخدم لمحاكاة عملية رمي النرد (Dice Rolling).

---

2. إنشاء object :

```
private static final SecureRandom random = new SecureRandom();
```

- يتم إنشاء كائن ثابت من نوع `SecureRandom`.
- `SecureRandom` هو مولد أرقام عشوائية آمن (cryptographically secure)، يُستخدم لتوليد أرقام عشوائية بجودة عالية.

---

3. تعريف الدالة `rollDice`:

```
public int[] rollDice(int poolSize, int faceNum) {
```

- `rollDice`: هي دالة تُحاكي رمي عدد معين من النرد.
- `int poolSize`: يمثل عدد النردات التي سيتم رميها.
- `int faceNum`: يمثل عدد الأوجه لكل نرد (مثل 6 أوجه للنرد التقليدي).

---

4. إنشاء مصفوفة لتخزين النتائج:

```
int[] dice = new int[poolSize];
```

- يتم إنشاء مصفوفة من الأعداد الصحيحة لتخزين نتائج رمي النرد.

---

5. ملء المصفوفة بالأرقام العشوائية:

```
for (int i = 0; i < dice.length; i++) { dice[i] = random.nextInt(faceNum) + 1;}
```

- يتم تكرار الحلقة `for` بعدد النردات (`poolSize`).
- في كل تكرار:
  1. يتم توليد رقم عشوائي باستخدام `random.nextInt(faceNum)`.
     - هذا يولد رقمًا عشوائيًا بين 0 و`faceNum - 1`.**
  2. يتم إضافة 1 للرقم العشوائي الناتج ليكون بين 1 و**faceNum**.

3. يتم تخزين الرقم العشوائي في المصفوفة `dice`.

---

6. إعادة المصفوفة:

`return dice;`

- بعد انتهاء الحلقة، يتم إرجاع المصفوفة `dice` التي تحتوي على نتائج رمي النرد.

هنا بقي اللي ممكن نعمل هو نعدل الرقم اللي هو ده random.nextInt(faceNum) + 1 نخليه بدل 1 يكون مثلا رقم كبير علشان يجمع عليه دي بقي ممكن نعمله ب smali

chagne 0x1 to 0x5 or any nubmer

```
.line 15
sget-object v2, Lorg/secuso/privacyfriendlydicer/dicer/Dicer;   ->random:Ljava/security/SecureRandom;

invoke-virtual {v2, p2}, Ljava/security/SecureRandom;  ->nextInt(I)I

move-result v2

add-int/lit8 v2, v2, 0x1

aput v2, v0, v1
```

دلوقتي احنا عايزين ننشئ كود نستخدمه مع Frida : هنا اه الكود ده هيطبع pramater اول ما استخدم الدالة دي اللي هي rollDice بس خلي بالك هنا مش هينفذ الدالة في الكود ولكن هنيفذ الدالة اللي كتبناها وهيطبع القيم

```
Java.perform(function(){
    var myDer=Java.use('org.secuso.privacyfriendlydicer.dicer.Dicer');
    myDer.rollDice.implementation=function(param1,param2){
        send("Parmater 1 is : "+param1);
        send("Parmater 2 is : "+param2);
        return this.rollDice(param1,param2);
    }
});
```

هنا اهو طبع القيم ولكن حصل error علشان غيرنا الفانكشن المفروض كانت بترجع قيمة ممكن بقي نستدعي الفانكشن باستخدام this.roolDice(param1,param2).f

```
[Android Emulator 5554::Dicer ]-> Java.perform(function(){var myDicer =Java.use('org.secuso.privacyfriendlydicer.dicer.D
    icer'); myDicer.rollDice.implementation=function(param1,param2){ send("Parameter 1= "+param1); send("Parameter 2 = "
+param2);}});

SyntaxError: unexpected end of string
    at <input>:1
[Android Emulator 5554::Dicer ]-> Java.perform(function(){var myDicer=Java.use('org.secuso.privacyfriendlydicer.dicer.Di
cer'); myDicer.rollDice.implementation=function(param1,param2){ send("Parameter 1= "+param1); send("Parameter 2 = "
+param2);}});
[Android Emulator 5554::Dicer ]-> message: {'type': 'send', 'payload': 'Parameter 1= 5'} data: None
message: {'type': 'send', 'payload': 'Parameter 2 = 6'} data: None
Error: Implementation for rollDice expected return value compatible with [I
    at re (frida/node_modules/frida-java-bridge/lib/class-factory.js:678)
    at <anonymous> (frida/node_modules/frida-java-bridge/lib/class-factory.js:655)
Process crashed: java.lang.NullPointerException: Attempt to get length of null array

***
FATAL EXCEPTION: main
Process: org.secuso.privacyfriendlydicer, PID: 14475
java.lang.NullPointerException: Attempt to get length of null array
        at org.secuso.privacyfriendlydicer.ui.MainActivity.displaySum(MainActivity.java:240)
        at org.secuso.privacyfriendlydicer.ui.MainActivity.access$000(MainActivity.java:40)
        at org.secuso.privacyfriendlydicer.ui.MainActivity$1.onChanged(MainActivity.java:73)
        at org.secuso.privacyfriendlydicer.ui.MainActivity$1.onChanged(MainActivity.java:70)
        at androidx.lifecycle.LiveData.considerNotify(LiveData.java:131)
        at androidx.lifecycle.LiveData.dispatchingValue(LiveData.java:149)
        at androidx.lifecycle.LiveData.setValue(LiveData.java:307)
        at androidx.lifecycle.MutableLiveData.setValue(MutableLiveData.java:50)
        at androidx.lifecycle.LiveData$1.run(LiveData.java:91)
        at android.os.Handler.handleCallback(Handler.java:959)
        at android.os.Handler.dispatchMessage(Handler.java:100)
        at android.os.Looper.loopOnce(Looper.java:232)
```

true code is

```
Java.perform(function(){
    var myDer=Java.use('org.secuso.privacyfriendlydicer.dicer.Dicer');
    myDer.rollDice.implementation=function(param1,param2){
        send("Parmater 1 is : "+param1);
        send("Parmater 2 is : "+param2);
        return this.rollDice(param1,param2);
    }
});
```

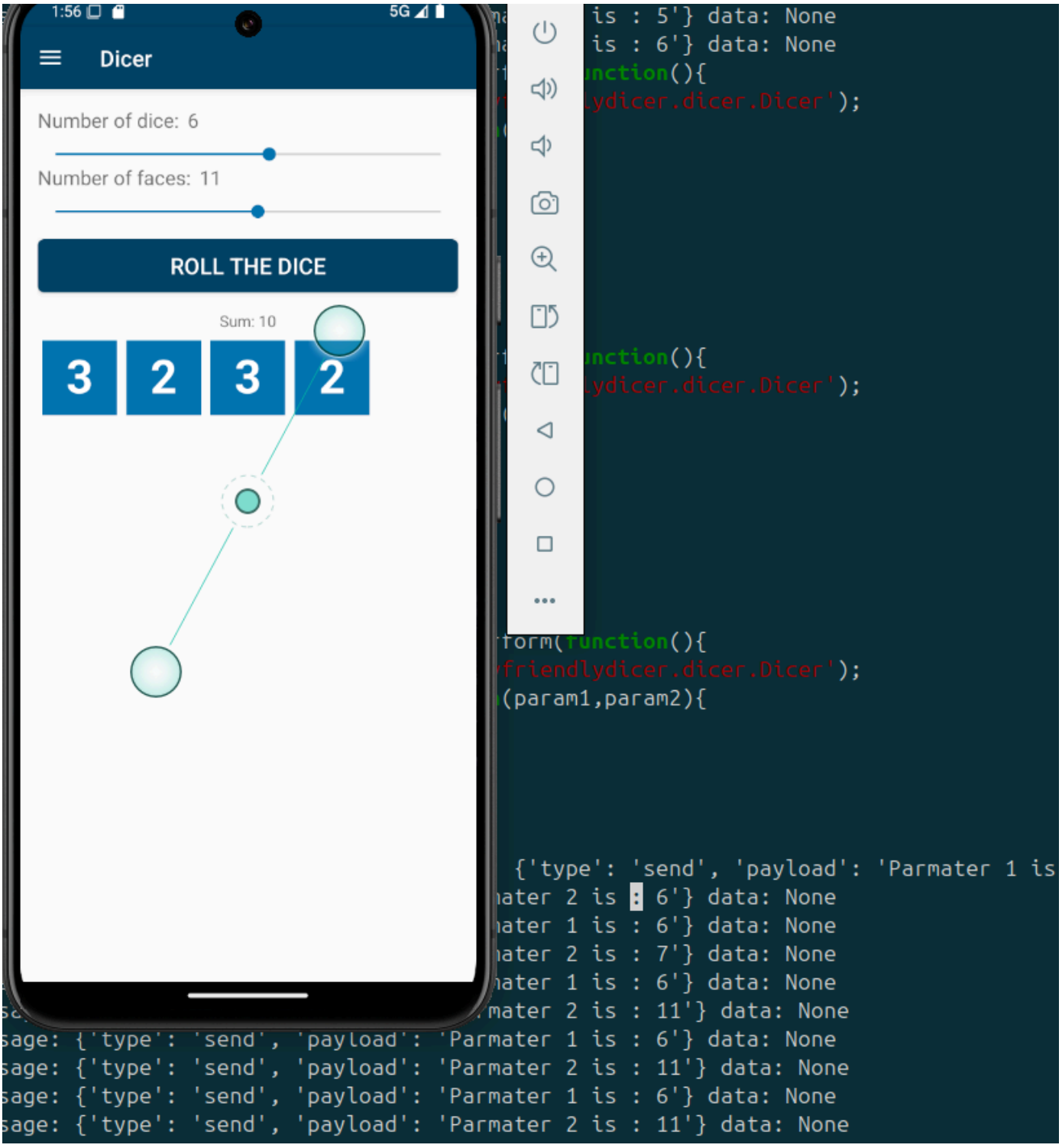
```
});
[Android Emulator 5554::Dicer ]-> message: {'type': 'send', 'payload': 'Parmater 1 is : 5'} data: None
message: {'type': 'send', 'payload': 'Parmater 2 is : 6'} data: None
message: {'type': 'send', 'payload': 'Parmater 1 is : 5'} data: None
message: {'type': 'send', 'payload': 'Parmater 2 is : 6'} data: None
message: {'type': 'send', 'payload': 'Parmater 1 is : 5'} data: None
message: {'type': 'send', 'payload': 'Parmater 2 is : 6'} data: None
```

5- Modifying Parameters

هنتكلم بقي ازاي اعدل ونغير في parameters اللي في method : اللي عاوزين نعمله هو ان نستخدم 4 نرد دائما ويكونوا بين 1 و 3 : هو ان لما نستدعي الدالة نحط احنا الارقام

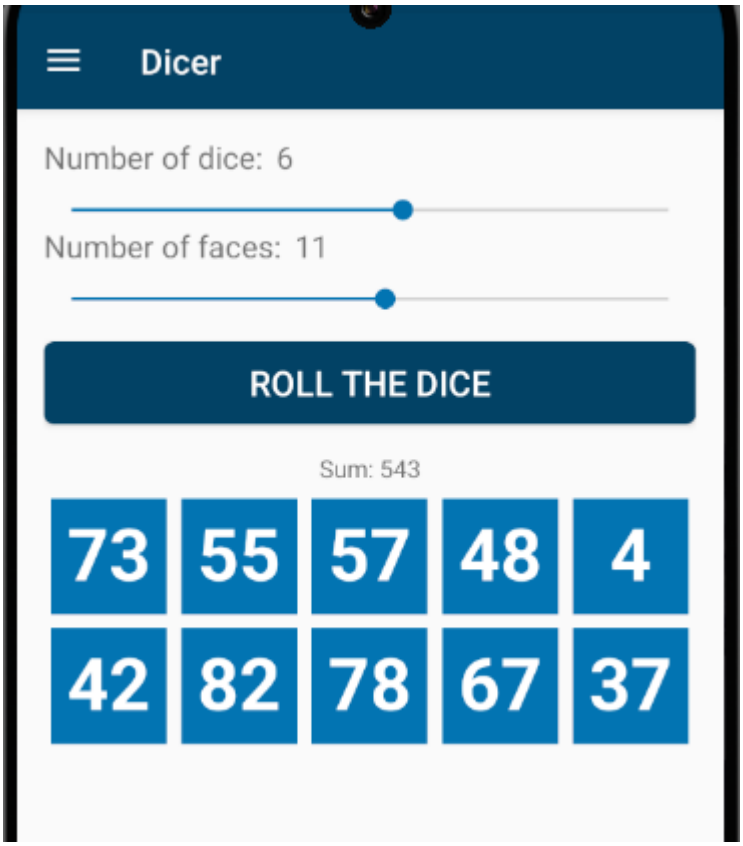
```
return this.rollDice(4,3);
```

هنا اهو مهما زودة اي قيمة هتطلع برضه نفس القيم اللي مطوطه اللي هما 4 نرد ويكونو بين 1و3



لو غيرنا مثلا يكونوا 10 نرد ويكونوا بين 1و100

```
return this.rollDice(10,100);
```



6-Function overloading

هنا علشان احنا مش محددين فانكشن فممكن يكون ميعرفش هستخدم ان فانكشن فاحنا هنا هنحدد احنا فانكشن : مثلا هنا في java.security.SecureRandom بيبقي في اكتر من method هنا اهو فانكشن بتاخد قيمة وفانكشن مش بتاخد قيمة فاحنا عايزين نحدد احنا الفانشكن

nextInt

Added in API level 1

```
public int nextInt ()
```

Returns the next pseudorandom, uniformly distributed `int` value from this random number generator's sequence. The general contract of `nextInt` is that one `int` value is pseudorandomly generated and returned. All 2^{32} possible `int` values are produced with (approximately) equal probability.

Implementation Requirements:

The method `nextInt` is implemented by class `Random` as if by:

```
public int nextInt() {
   return next(32);
 }
```

هنا هيخلي function اللي مش بتاخد قيم ترجع 0

```
Java.perform(function(){
    var random=Java.use('java.security.SecureRandom');
    random.nextInt.implementation=function(param){
        return 0;
    }
});
```

هنا هيدي error علشان هو مش عارف يستخدم انهي function قي الاتنين لا في واحدة بتاخد paramter and return random number والتانية مش بتاهد parameter return random number

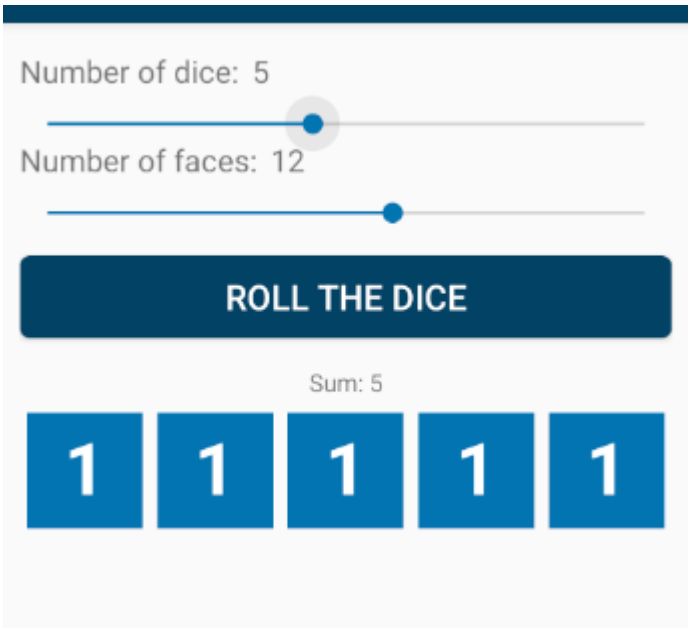
nextInt() --> return random number

nextInt(param) return random number between 0 and param

```
[Android Emulator 5554::Dicer ]-> Java.perform(function(){
    var random=Java.use('java.security.SecureRandom');
    random.nextInt.implementation=function(param){
        return 0;
    }
});
Error: nextInt(): has more than one overload, use .overload(<signature>) to choose from:
    .overload()
    .overload('int')
    at Q (frida/node_modules/frida-java-bridge/lib/class-factory.js:626)
    at X (frida/node_modules/frida-java-bridge/lib/class-factory.js:621)
    at set (frida/node_modules/frida-java-bridge/lib/class-factory.js:1103)
    at <anonymous> (<input>:2)
    at <anonymous> (frida/node_modules/frida-java-bridge/lib/vm.js:12)
    at _performPendingVmOps (frida/node_modules/frida-java-bridge/index.js:248)
    at <anonymous> (frida/node_modules/frida-java-bridge/index.js:223)
    at <anonymous> (frida/node_modules/frida-java-bridge/lib/vm.js:12)
    at _performPendingVmOpsWhenReady (frida/node_modules/frida-java-bridge/index.js:242)
    at perform (frida/node_modules/frida-java-bridge/index.js:202)
    at <eval> (<input>:6)
    at eval (native)
    at <anonymous> (/frida/repl-1.js:34)
    at evaluate (/frida/repl-1.js:57)
    at fridaEvaluateExpression (/frida/repl-1.js:34)
    at call (native)
    at <anonymous> (frida/runtime/message-dispatcher.js:11)
    at o (frida/runtime/message-dispatcher.js:23)
[Android Emulator 5554::Dicer ]->
```

احنا بقي عاوزين نحدد يستخدم ايه : هو ان هنضيف .overload('int') بعد ما نحدد الدالة فكده لما نخلي القمة ترجع 0 يبقي اي حاجة في هنرميها لازم ترجع 1 علشان هو بيجمع عليها 1 في الكود الاصلي

```
Java.perform(function(){
    var random=Java.use('java.security.SecureRandom');
    random.nextInt.overload('int').implementation=function(param){
        return 0;
    }
});
```



if we want to return 11 we want to change the return value to be 10

```
Java.perform(function(){
    var random=Java.use('java.security.SecureRandom');
    random.nextInt.overload('int').implementation=function(param){
        return 10;
    }
});
```

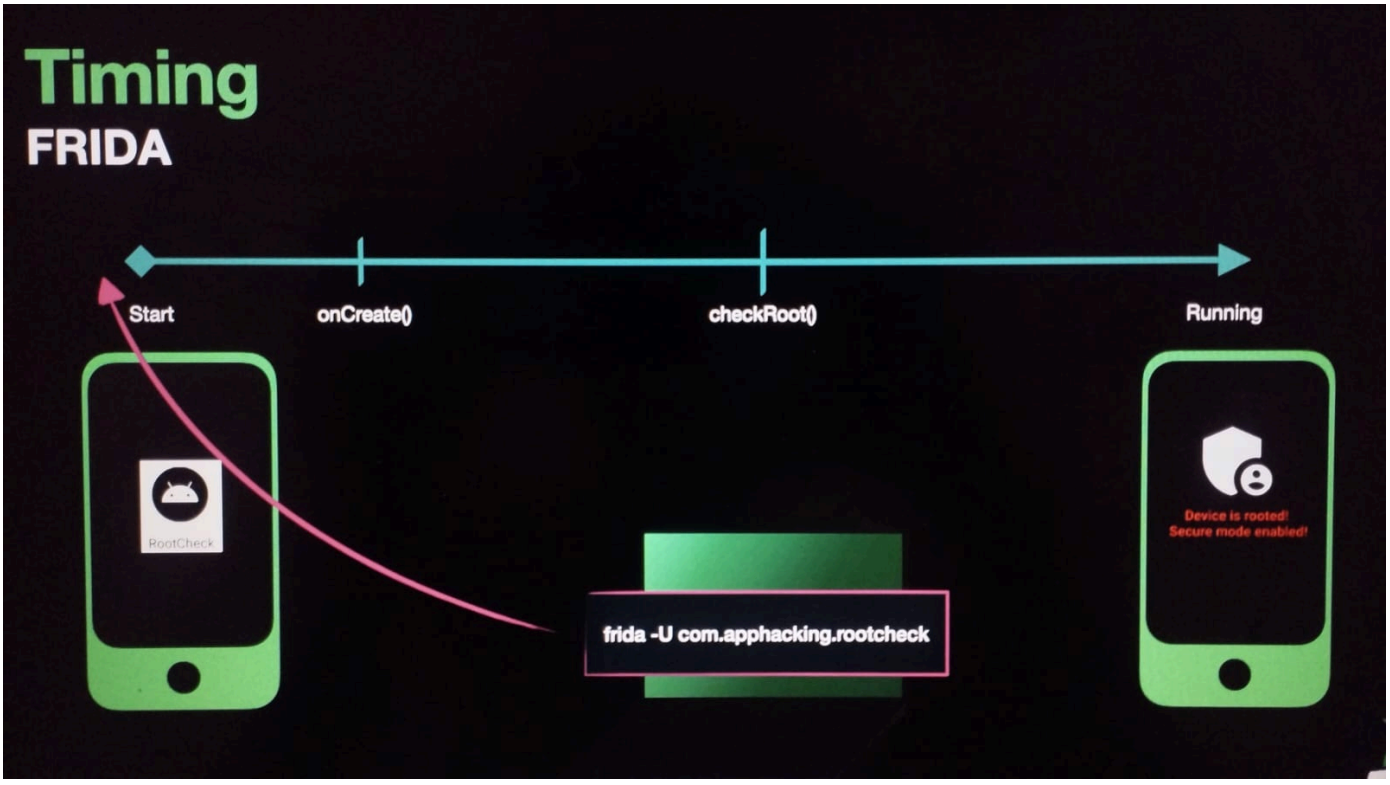
يبقي كده دلوقتي لازم تعرف اهم حاجتين وانتا بتعمل code javascript

1- use Java.perform() : JVM ب Frida علشان تربط

2- Java.use('class name') علشان تعرف class اللي هتستخدمه

7-Timing

هنا بقي منا بنستخدم Frida وهو التطبيق معموله running فدلوقتي لو في تطبيق زي اللي في الصورة ده بيعمل checkroot فلما نجي نستخدمه وهو running مش هنعرف نستخدمه علشان كده بقي عاوزين نستخدم frida من قبل ما يعمل running يعني لما يعمل start



if we want to print 1 2 3 4 5 6 on Dicer apk

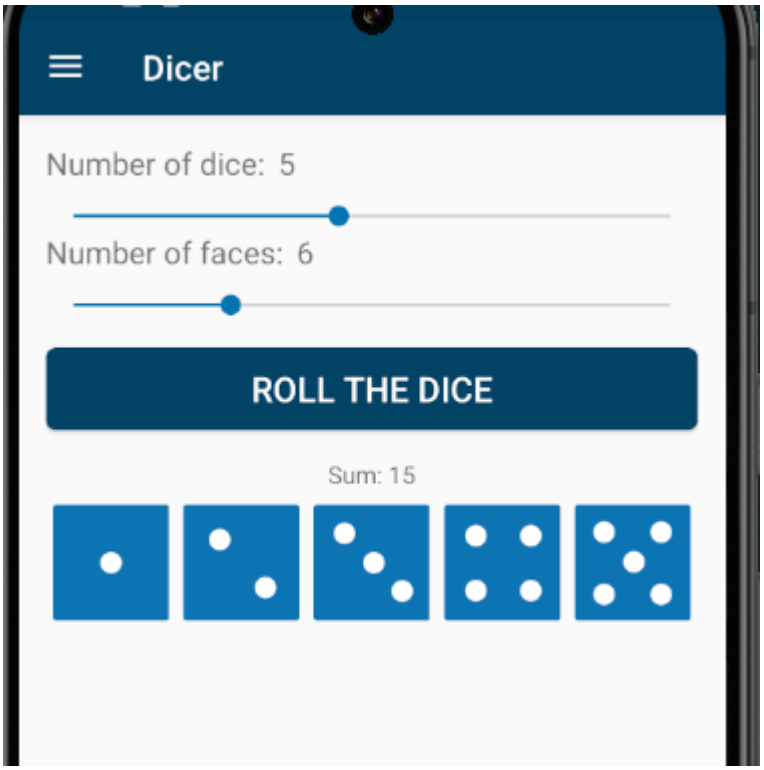
1- code javascript

```
Java.perform(function(){
    var myClass=Java.use('org.secuso.privacyfriendlydicer.dicer.Dicer');
    myClass.rollDice.implementation=function(param1,parma2){
        var arr=Java.array('int',new Array(param1).fill(0));
        for (var i=0;i<param1;i++){
            arr[i]=i+1;
        }
        return arr;
 };
});
```

2- save it on file.js and run frida with script

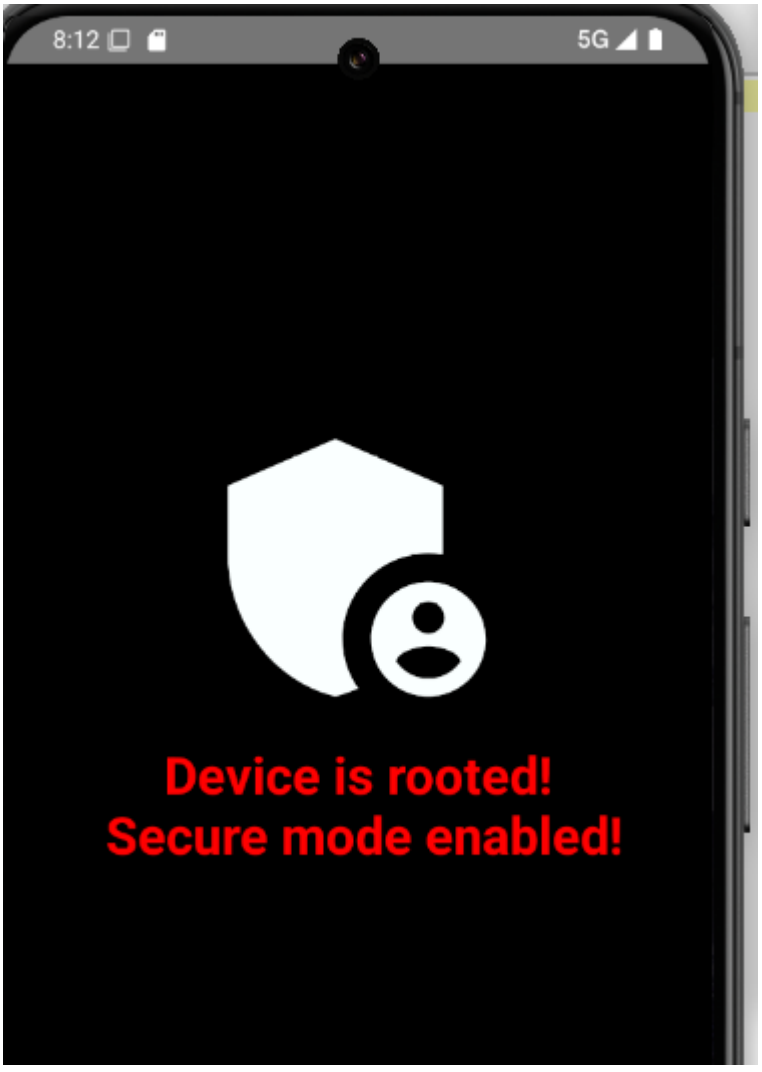
-f --> to run script when app starting

```
frida -U -f org.secuso.privacyfriendlydicer -l script2.js
```



8- RootCheck Bypass

هنا اهو احنا عاوزين منبقاش root



هنا بقي احنا عايزين return false علشان ميتحققش من root

```java
package com.apphacking.rootcheck;

import android.os.Build;
import java.io.BufferedReader;
import java.io.File;
import java.io.InputStreamReader;

/* loaded from: classes.dex */
public class RootUtil {
    public static boolean isDeviceRooted() {
        return checkRootMethod1() || checkRootMethod2() || checkRootMethod3();
    }
```

هنا هو ده الكود بس مش هيتنفذ برضه علشان احنا بنعمله وهو التطبيق running

```
Java.perform(function(){
    var RootCheck=Java.use('com.apphacking.rootcheck.RootUtil');
    RootCheck.isDeviceRooted.implementation=function(){
        return false;
    };
});
```

دلوقتي بقي احنا عاوزين نخبر frida ان هي تستخدمه اول ما يعمل start <--- بنستخدم -f

-f ---> to tell frida to run script with app when start

```
frida -U -f com.apphacking.rootcheck -l bypassRoot.js
```

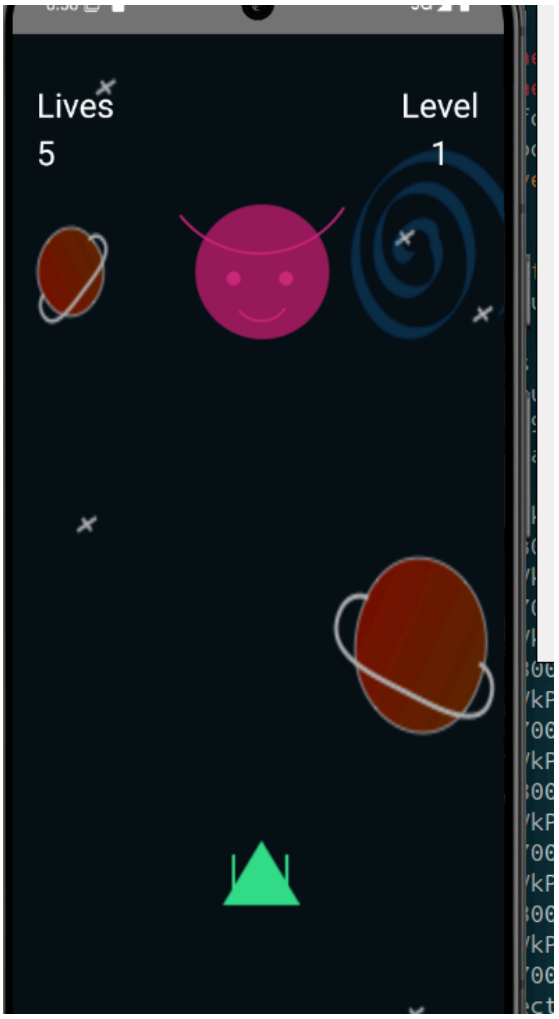
another code

```
Java.perform(function(){
    var RootCheck=Java.use('com.apphacking.rootcheck.RootUtil');
    RootCheck.checkRootMethod1.implementation=function(){
        return false;
    }
    RootCheck.checkRootMethod2.implementation=function(){
        return false;
    }

    RootCheck.checkRootMethod3.implementation=function(){
        return false;
    }
});
```

9- Static Method

هو ازاي نستدعي الدوال الثابتة في Frida

هنا في التطبيق ده احنا عاوزين نعلي level بتاعه



```
Player × Item × MainActivity ×
import android.content.Context;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

/* loaded from: classes2.dex */
30 public class MainActivity extends AppCompatActivity {
    public static ImageView alien;
    public static Context mainContext;
    public static ConstraintLayout mainLayout;
    public static TextView txtViewLevel;
    Button button;
    Player player;
    TextView txtViewLive;
    public static int level = 1;
    public static int highScore = 100;

    @Override // androidx.fragment.app.FragmentActivity, androidx.activity.ComponentActivity, androidx.
31  protected void onCreate(Bundle savedInstanceState) {
32      super.onCreate(savedInstanceState);
33      setContentView(C0459R.layout.activity_main);
35      this.button = (Button) findViewById(C0459R.id.imgButton);
36      mainContext = getApplicationContext();
37      mainLayout = (ConstraintLayout) findViewById(C0459R.id.mainLayout);
39      this.txtViewLive = (TextView) findViewById(C0459R.id.textViewLives);
40      TextView textView = (TextView) findViewById(C0459R.id.textViewLevel);
        txtViewLevel = textView;
42      textView.setText(String.valueOf(level));
44      this.player = new Player();
45      new Alien(320);
    }

    public static void nextLevel() {
        level++;
52      System.out.println("Current Level is = " + level);
53      System.out.println("Current HighScore is = " + highScore);
    }
```

احنا عاوزين نستدعي الدالة اللي هي next level

```
Java.perform(function(){
    var increase_level=Java.use('com.apphacking.fridafunc.MainActivity');
    increase_level.nextLevel();
});
```

```
adb logcat | grep --color=auto -i "curre...
→ ~ adb logcat |grep -i "current level is "
05-01 09:03:21.231  4312  4796 I System.out: Current Lev
el is = 2
05-01 09:08:45.142  8804  9222 I System.out: Current Lev
el is = 2
05-01 09:08:53.220  8804  9222 I System.out: Current Lev
el is = 3
05-01 09:08:53.995  8804  9222 I System.out: Current Lev
el is = 4
05-01 09:08:54.420  8804  9222 I System.out: Current Lev
el is = 5
05-01 09:08:54.725  8804  9222 I System.out: Current Lev
el is = 6
05-01 09:08:55.008  8804  9222 I System.out: Current Lev
el is = 7
05-01 09:08:55.278  8804  9222 I System.out: Current Lev
el is = 8
```

```
    var increase_level=Java.use('com.apphacking.fridafun
c.MainActivity');
    increase_level.nextLevel();
});
[Android Emulator 5554::FridaFunc ]-> Java.perform(funct
ion(){
    var increase_level=Java.use('com.apphacking.fridafun
c.MainActivity');
    increase_level.nextLevel();
});
[Android Emulator 5554::FridaFunc ]-> Java.perform(funct
ion(){
    var increase_level=Java.use('com.apphacking.fridafun
c.MainActivity');
    increase_level.nextLevel();
});
[Android Emulator 5554::FridaFunc ]-> Java.perform(funct
ion(){
    var increase_level=Java.use('com.apphacking.fridafun
c.MainActivity');
    increase_level.nextLevel();
});
[Android Emulator 5554::FridaFunc ]->
```

دلوقتي بقي هنا احنا عاوزين نستخدم اي method من دول اللي في Class Player

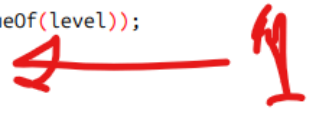
Player | Item | MainActivity | Alien | BuildConfig

```java
package com.apphacking.fridafunc;

/* loaded from: classes2.dex */
public class Player {
    int lives = 5;
    int power = 10;

    Player() {
        System.out.println("A new player object has been created!");
    }

    void increaseLive() {
        this.lives++;
        System.out.println("This is impossible? This method does not get called anywhere in this code!");
        System.out.println("Player lives INCREASING? Lives = " + this.lives);
    }

    void decreaseLive() {
        this.lives--;
    }
}
```

بس هيبقي في error علشان لو روحنا للكود الاصلي هنلاقيه هو في الاول بينشء player بعد كده بيستدعي method

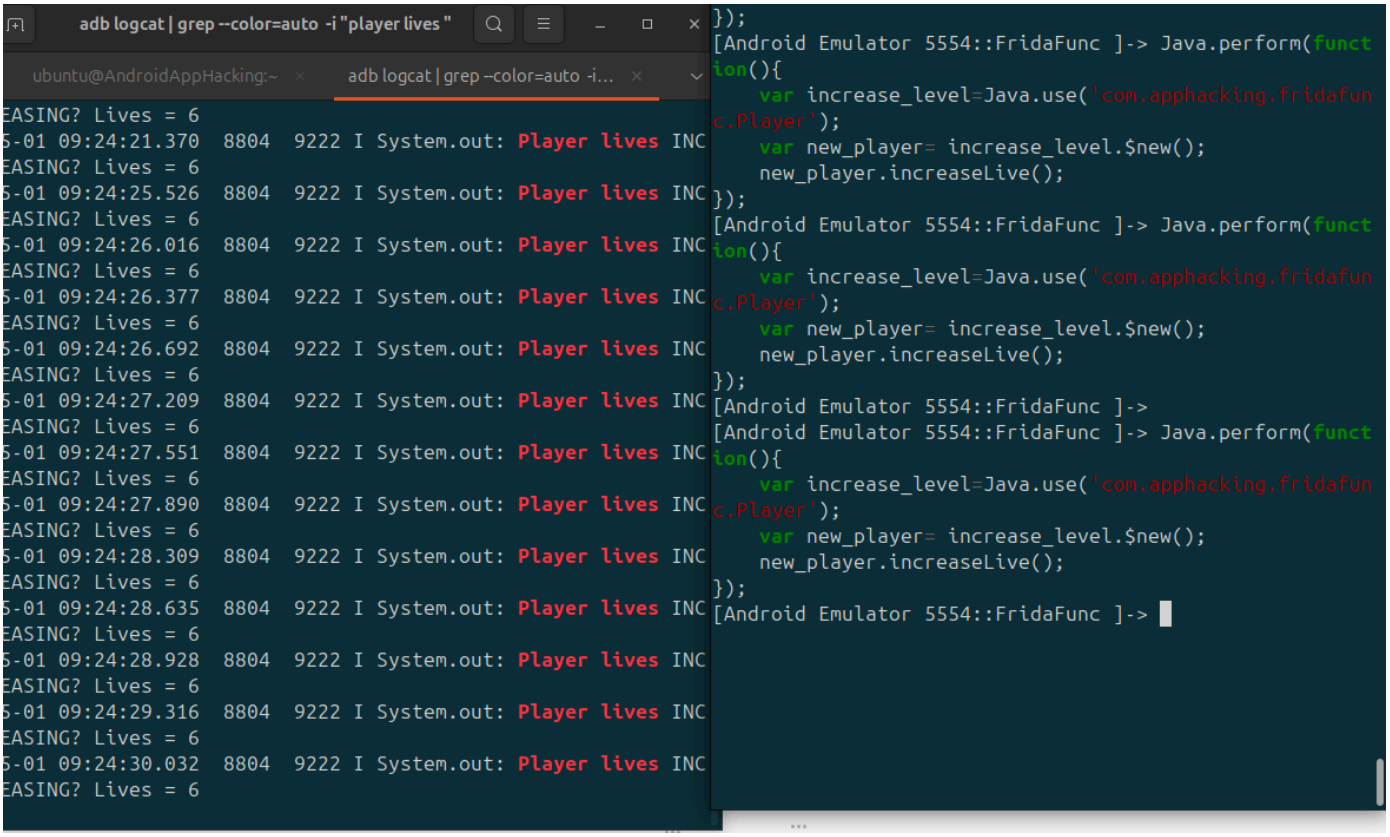
Player | Item | MainActivity | Alien | BuildConfig

```java
import android.content.Context;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

/* loaded from: classes2.dex */
public class MainActivity extends AppCompatActivity {
    public static ImageView alien;
    public static Context mainContext;
    public static ConstraintLayout mainLayout;
    public static TextView txtViewLevel;
    Button button;
    Player player;
    TextView txtViewLive;
    public static int level = 1;
    public static int highScore = 100;

    @Override // androidx.fragment.app.FragmentActivity, androidx.activity.ComponentActivity, androidx.core.app.Co
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(C0459R.layout.activity_main);
        this.button = (Button) findViewById(C0459R.id.imgButton);
        mainContext = getApplicationContext();
        mainLayout = (ConstraintLayout) findViewById(C0459R.id.mainLayout);
        this.txtViewLive = (TextView) findViewById(C0459R.id.textViewLives);
        TextView textView = (TextView) findViewById(C0459R.id.textViewLevel);
        txtViewLevel = textView;
        textView.setText(String.valueOf(level));
        this.player = new Player();
        new Alien(320);
    }

    public static void nextLevel() {
        level++;
        System.out.println("Current Level is = " + level);
        System.out.println("Current HighScore is = " + highScore);
    }

    public void playerCollision(View view) {
        this.player.decreaseLive();
        if (this.player.lives <= 0) {
            this.txtViewLive.setText("GAME OVER!");
        } else {
            this.txtViewLive.setText(String.valueOf(this.player.lives));
        }
    }
}
```



فدلوقتي عاوزين ننشئ player

```
Java.perform(function(){
    var increase_level=Java.use('com.apphacking.fridafunc.Player');
    var new_player= increase_level.$new();
    new_player.increaseLive();
});
```



بس دلوقتي هيبقي في مشكلة هو ان كل ما اذود هيروي ينشئ واحد جديد ويذود 1 وبع كده برجع يعمل نفس الفانشكن ينشئ واحد جديد ويذود 1 فدلوقتي عاوزين نعمل حاجة تنشء مرة واحدة بس هنعلمها في next topic

---

10- Working with instance

هنا بقي احنا عايزين نغير Instance(object) اللي في Memory

```
Java.perform(function(){
    Java.choose('Class_name',{
        onMatch:function(instance){

        },
        onComplete:function(){

        }
    })
});
```

`Java.choose`:

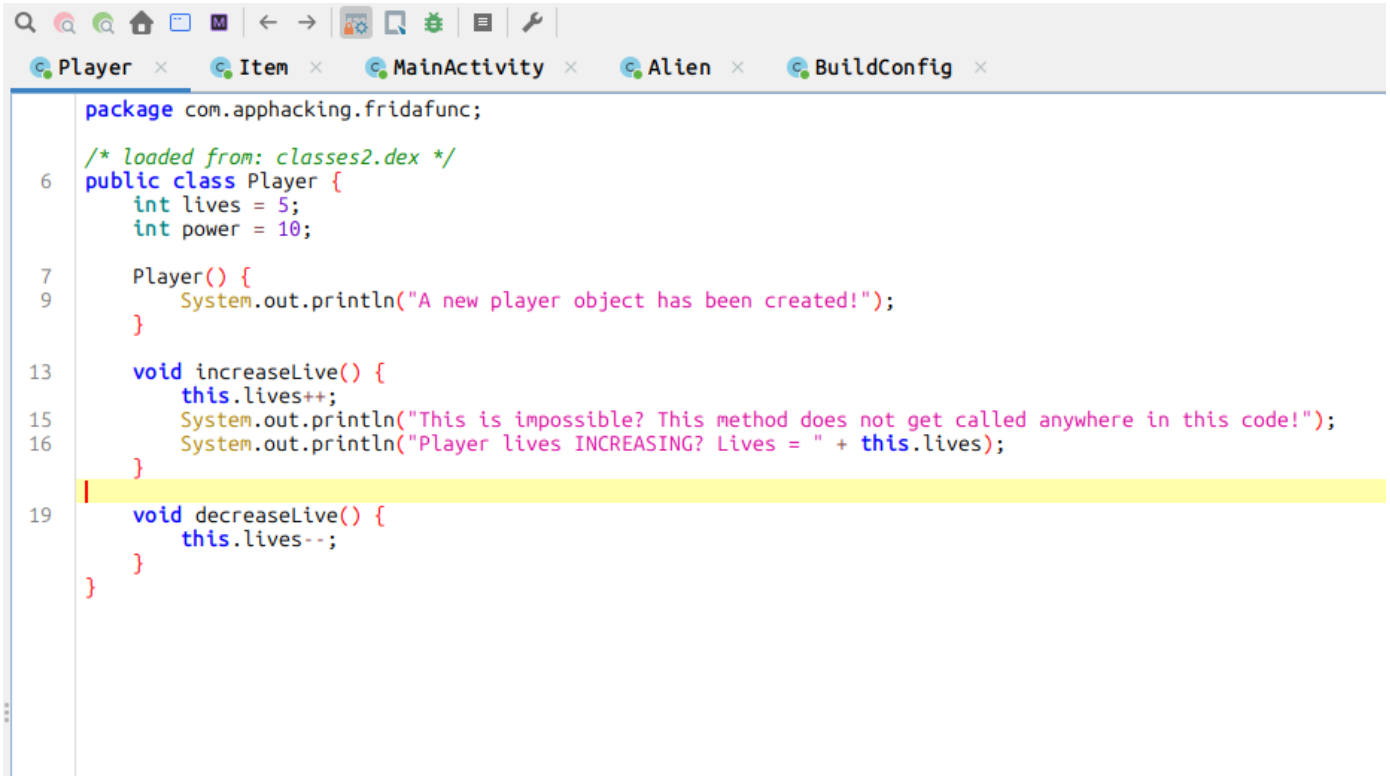
- **الوصف:**
  - تُستخدم للعثور على جميع الكائنات النشطة (instances) لفئة معينة (`Class_name`) في التطبيق.
  - يمكن أن تكون هذه الكائنات موجودة في الذاكرة أثناء تشغيل التطبيق.
- **المعلمات:**
- **Callback Object :**
  - **onMatch : يتم استدعائها عند العثور علي كاءن نشط**
  - **onComplete : after complete onMatch Operation**

```
Java.perform(function(){
    Java.choose('com.apphacking.fridafunc.Player',{
        onMatch: function(instance){
            send("An instance of  the player class has been found : "+instance);
        },
        onComplete: function(){
            send("Frida has finshed scanning the application memory for player instance");
        }
    })
});
```

هنا اهو جاب كل المتغيرات الخاصة ب player اللي متخزنة في Memory

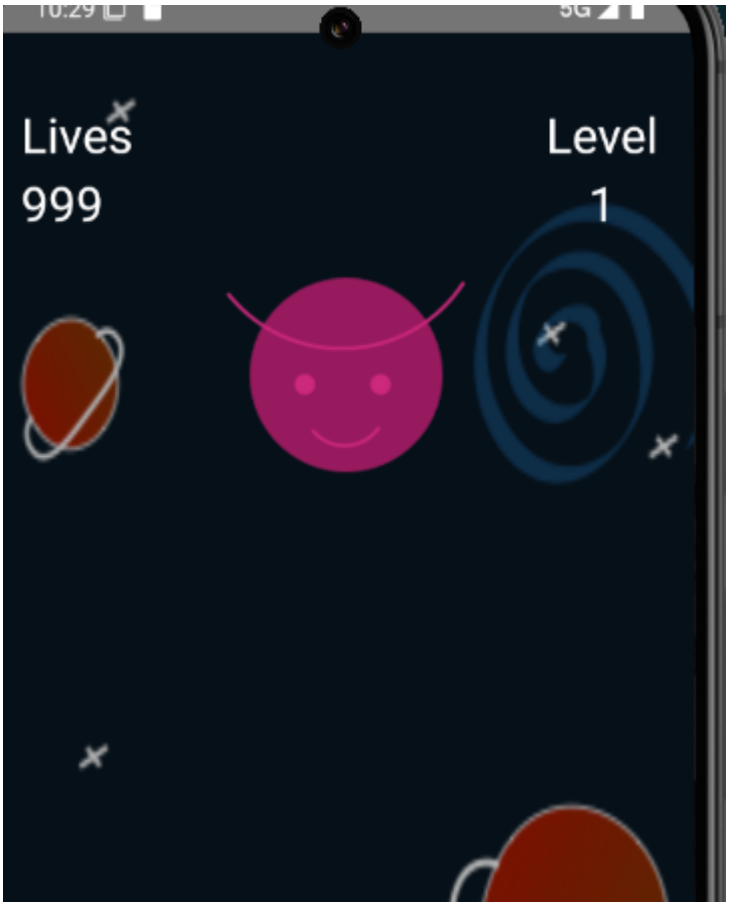
```
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@e3caba'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@dd95fe5'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@beb2dc'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@ee02b4f'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@c141ae'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@abf4a29'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@8a48bb0'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@373d886'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@1f03547'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@3397162'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@f6c8174'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@438bf44'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@e928857'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@3cd52f3'} data: None
message: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@908582d'} data: None
message: {'type': 'send', 'payload': 'Frida has finshed scanning the application memory for player instance'} data: None
[Android Emulator 5554::FridaFunc ]->
```

دلوقتي بقي هنا لو عاوزين نغير قيم lives , power باستخدام instance




```
Java.perform(function(){
    Java.choose('com.apphacking.fridafunc.Player',{
        onMatch: function(instance){
            send("the player lives is  = "+instance.lives.value);
            instance.lives.value=1000;
            instance.power.value=1000;
            send("the player lives is  = "+instance.lives.value);
            send("An instance of  the player class has been found :
"+instance);
        },
        onComplete: function(){
            send("Frida has finshed scanning the application memory for
player instance");
        }
    })
});
```

```
essage: {'type': 'send', 'payload': 'the player lives is  = 1000'} data: None
essage: {'type': 'send', 'payload': 'the player lives is  = 1000'} data: None
essage: {'type': 'send', 'payload': 'An instance of  the player class has been found : com.apphacking.fridafunc.Player@
e3caba'} data: None
```



if we want to increase lives

```
Java.perform(function(){
    Java.choose('com.apphacking.fridafunc.Player',{
        onMatch: function(instance){
            instance.increaseLive();
        },
        onComplete: function(){
        }
    })
});
```

```
05-01 10:37:11.400 10611 12164 I System.out: Player lives INCREASING? Lives = 1001
```

11- Instance as a parameter

دلوقتي عاوزين نشوف لو Instance(objcet) بياخد parameter من نوع class زي كده مثلا :

```
Public class Item{
    int itemPower=1;
    public string name ="skillZap";
    Item(int itemPower){
        System.out.println("A new skillZap has been created");
```

```
        this.itemPower=itemPower;
    }
}
public class Boss{
    int highpoint=10;
    int power=9000;
    Item itemDrop;
    Boss(Item item){
         System.out.println("A new Boss has been created");
         this.itemDrop=item;
    }
}
```

Boss class take another class in parameter (Item) هنا اه

```
54          this.player = new Player();
56          Item item = new Item(1000);
57          this.boss = new Boss(item);
        }
```

عايزيزن نشوف بقي ازاي نستخدمه في Frida يعني نكتب code javascript يعرف يتحكم في ده

1- create new instance for Boss object take parameter of type class (Item)

2- create new instance for Item to sent as a parameter when we declare new Boss and take int value as a parameter

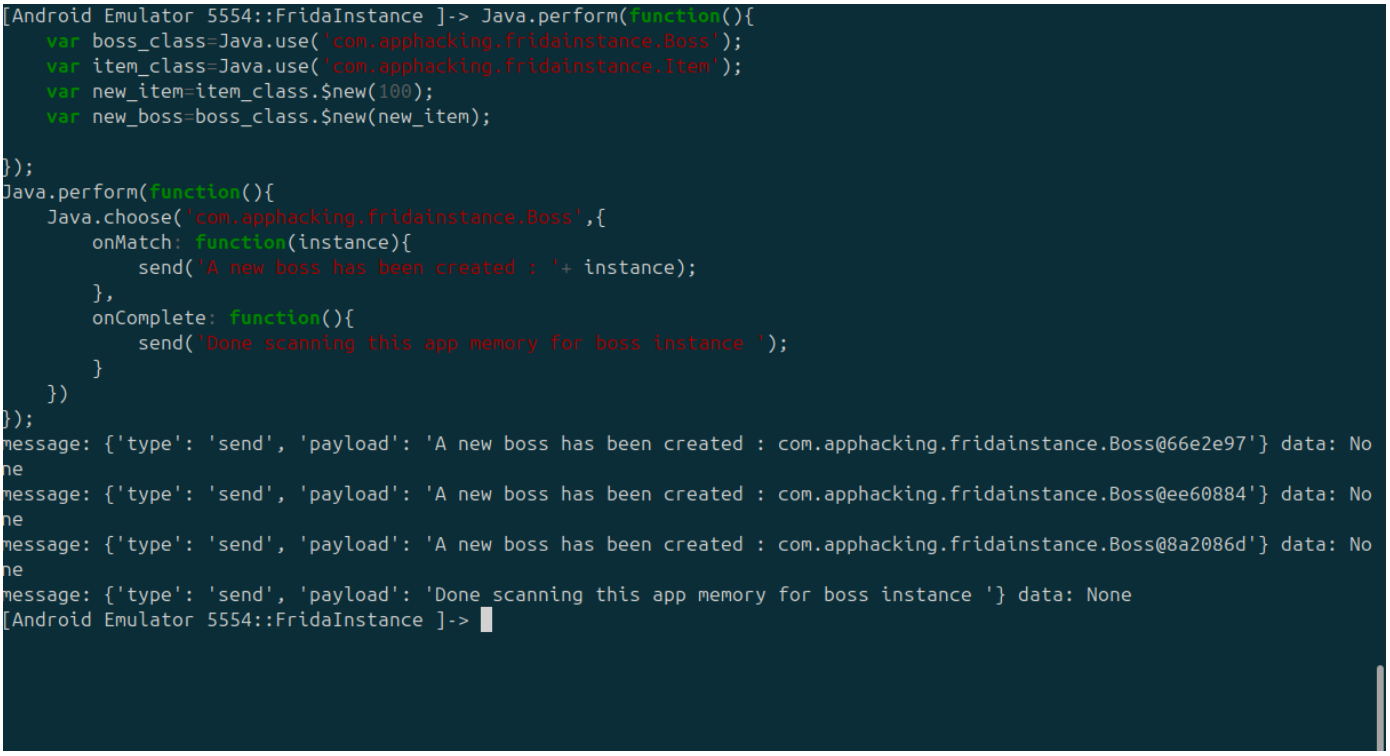
```
Java.perform(function(){
    // initializing Item class take new int value
    var item_class=Java.use('com.apphacking.fridainstance.Item');
        var new_item=item_class.$new(100);

    // initializing Boss class take class (Item)
    var boss_class=Java.use('com.apphacking.fridainstance.Boss');
    var new_boss=boss_class.$new(new_item);

});
```

```
05-02 08:22:13.244  6551  6551 I System.out: A new player object has been created!
05-02 08:22:13.245  6551  6551 I System.out: A new boss has been created!
05-02 08:22:57.620  6551  6643 I System.out: A new boss has been created!
```

```
Java.perform(function(){
    var boss_class=Java.use('com.apphacking.fridainstance.Boss');
    var item_class=Java.use('com.apphacking.fridainstance.Item');
    var new_item=item_class.$new(100);
    var new_boss=boss_class.$new(new_item);

});
```

```
Java.perform(function(){
    Java.choose('com.apphacking.fridainstance.Boss',{
        onMatch: function(instance){
            send('A new boss has been created : '+ instance);
        },
        onComplete: function(){
            send('Done scanning this app memory for boss instance ');
        }
    })
});
```

هنا احنا بنستخدم send علشان الرسالة تيجي لما run frida مش محتاجين نشوف logcat



لو عاوزين بقي نستخدم الطريقة التانية اللي هي Java.choose

```
Java.perform(function(){
    Java.choose('com.apphacking.fridainstance.Item',{
        onMatch: function(new_item){
            var boss_class=Java.use('com.apphacking.fridainstance.Boss');
            var new_boss=boss_class.$new(new_item);
            send('A new boss has been created : '+ new_boss);
        },
        onComplete :function(){
            send('Done scanning this app memory for boss instance ');
        }
    })
})
```

```
[Android Emulator 5554::FridaInstance ]-> Java.perform(function(){
    Java.choose('com.apphacking.fridainstance.Item',{
        onMatch: function(new_item){
            var boss_class=Java.use('com.apphacking.fridainstance.Boss');
            var new_boss=boss_class.$new(new_item);
            send('A new boss has been created : '+ new_boss);
        },
        onComplete :function(){
            send('Done scanning this app memory for boss instance ');
        }
    })
})
message: {'type': 'send', 'payload': 'A new boss has been created : com.apphacking.fridainstance.Boss@cdcfca2'} data: N
ne
message: {'type': 'send', 'payload': 'A new boss has been created : com.apphacking.fridainstance.Boss@65d1d33'} data: N
ne
message: {'type': 'send', 'payload': 'A new boss has been created : com.apphacking.fridainstance.Boss@cfd28f0'} data: N
ne
message: {'type': 'send', 'payload': 'Done scanning this app memory for boss instance '} data: None
[Android Emulator 5554::FridaInstance ]->
```

Constructor Hooking

لو عاوز استخدم constructor بنستخدم $init

```
Java.perform(function(){
    var increase_level=Java.use('com.apphacking.fridafunc.Player');
    var new_player= increase_level.$init.implementation=function(){



    }


});
```

12- UI Thread

هنا عايزين نتحكم في UI for app

```
package com.apphacking.fridafunc;

import android.view.View;
import android.widget.ImageView;
import androidx.constraintlayout.widget.ConstraintSet;

/* loaded from: classes2.dex */
public class Alien {
    Alien(int margin) {
        System.out.println("A new Alien object has been created");
        MainActivity.alien = new ImageView(MainActivity.mainContext);
        MainActivity.alien.setBackgroundResource(R.drawable.alien);
        MainActivity.alien.setId(View.generateViewId());
        MainActivity.mainLayout.addView(MainActivity.alien, 0);
        ConstraintSet set = new ConstraintSet();
        set.clone(MainActivity.mainLayout);
        set.connect(MainActivity.alien.getId(), 3, MainActivity.mainLayout.getId(), 3, margin);
        set.connect(MainActivity.alien.getId(), 1, MainActivity.mainLayout.getId(), 1);
        set.connect(MainActivity.alien.getId(), 2, MainActivity.mainLayout.getId(), 2);
        set.applyTo(MainActivity.mainLayout);
    }
}
```

هنا اه لو عاوزين نغير margin بتاع التطبيق

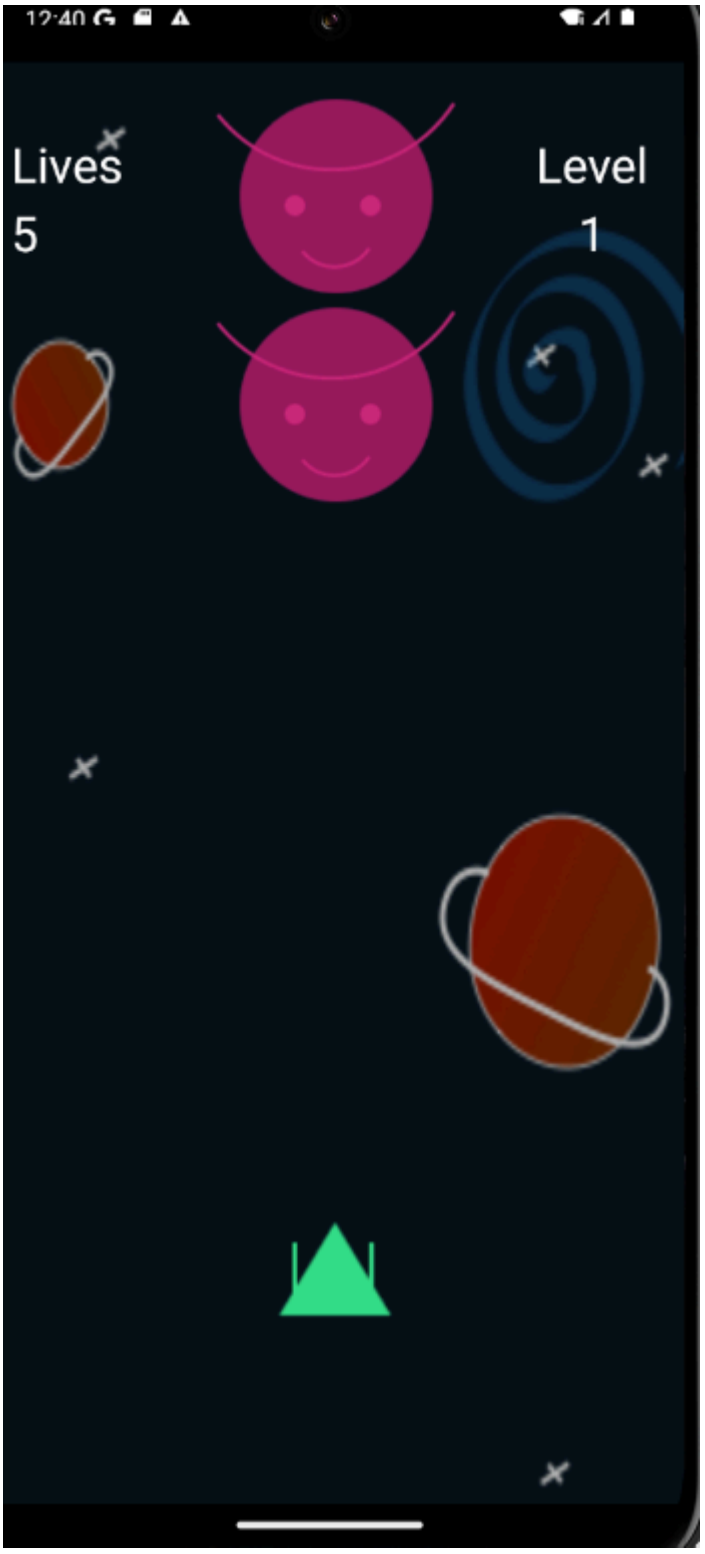
```
Java.perform(function(){
    var align_class=Java.use('com.apphacking.fridafunc.Algin');
    align_instance=align_class.$new(300);
})
```

```
[Android Emulator 5554::FridaFunc ]-> Java.perform(function(){
    var align_class=Java.use('com.apphacking.fridafunc.Alien');
    align_instance=align_class.$new(300);
})

[Android Emulator 5554::FridaFunc ]-> Error: android.view.ViewRootImpl$CalledFromWrongThreadException: Only the original
thread that created a view hierarchy can touch its views. Expected: main calling: fr.ca
    at <anonymous> (frida/node_modules/frida-java-bridge/lib/env.js:124)
    at value (frida/node_modules/frida-java-bridge/lib/class-factory.js:1237)
    at e (frida/node_modules/frida-java-bridge/lib/class-factory.js:643)
    at apply (native)
```

بس هنا مفيش حاجة هتتغير علشان لازم ننشيء Thread

-
  - `Java.scheduleOnMainThread` **ensures the code runs on the UI Thread** (Main Thread), **which is responsible for updating and interacting with the user interface in Android.**

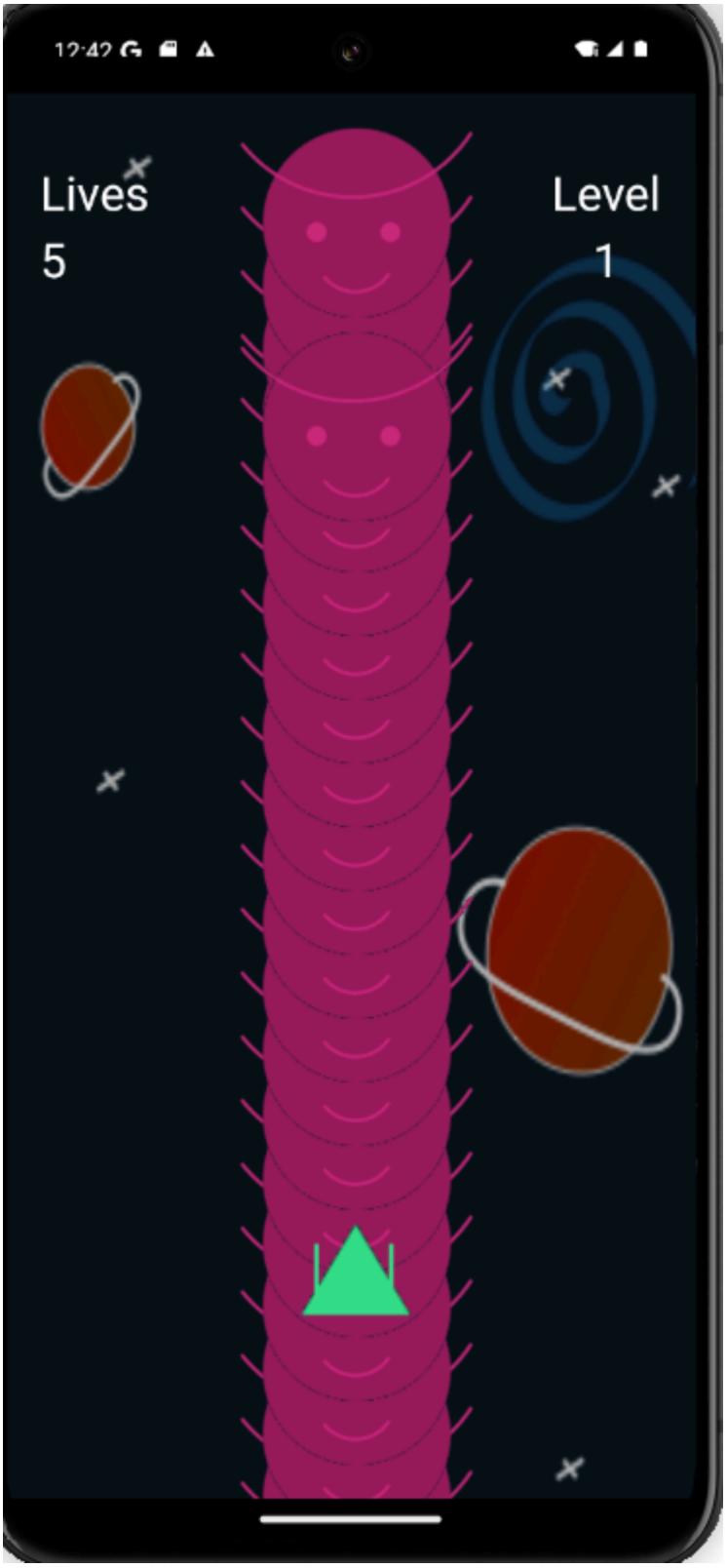
```
Java.perform(function(){
    var align_class=Java.use('com.apphacking.fridafunc.Alien');
    Java.scheduleOnMainThread(function(){
        align_instance=align_class.$new(500);

    })
})
```



if we write this code

```
Java.perform(function(){
    var align_class=Java.use('com.apphacking.fridafunc.Alien');
    Java.scheduleOnMainThread(function(){
        for (var i=0;i<10000;i++){
            align_instance=align_class.$new(i);

        }
    })
```

```
})
```



if we want to create text on the screen :

method to want to call (toastClassReference) public static Toast makeText(Context context, CharSequance text,int duration)

```
Java.perform(function(){
    var string_reference=Java.use('java.lang.String');
```

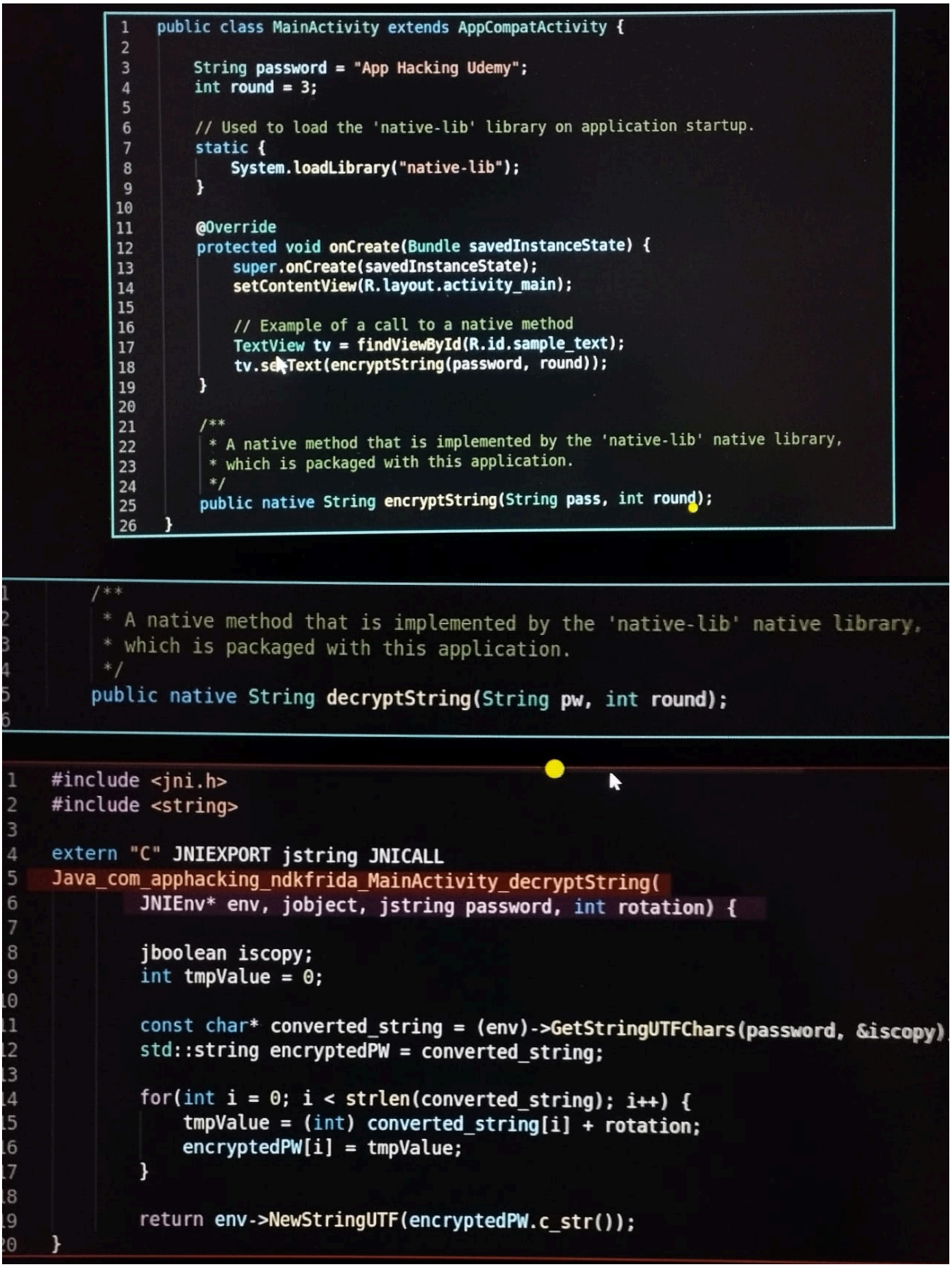
```
    var DisplayText=string_reference.$new("Frida is amazing also with the
UI")
    var align_class=Java.use('com.apphacking.fridafunc.Alien');
    var toastClassReference=Java.use('android.widget.Toast');
    var
context=Java.use('android.app.ActivityThread').currentApplication().getAppli
cationContext();
    Java.scheduleOnMainThread(function(){
        for (var i=0;i<10000;i++){
            align_instance=align_class.$new(i);

        }
        toastClassReference.makeText(context,DisplayText,1).show();
    })
})
```

13- Frida NDK (Native Development Kit)

هنا هنشوف ازاي نربط Frida with C++/C وده لو لاقينا في اللكود
System.loadLibrary("native_lib")

Frida

```
Frida with C++/C  --> System.loadLibrary("native-lib")

Interceptor(target,callback)

target --> pointer
callback --> onEnter : to access parameters ,  onLeave : to return value
```



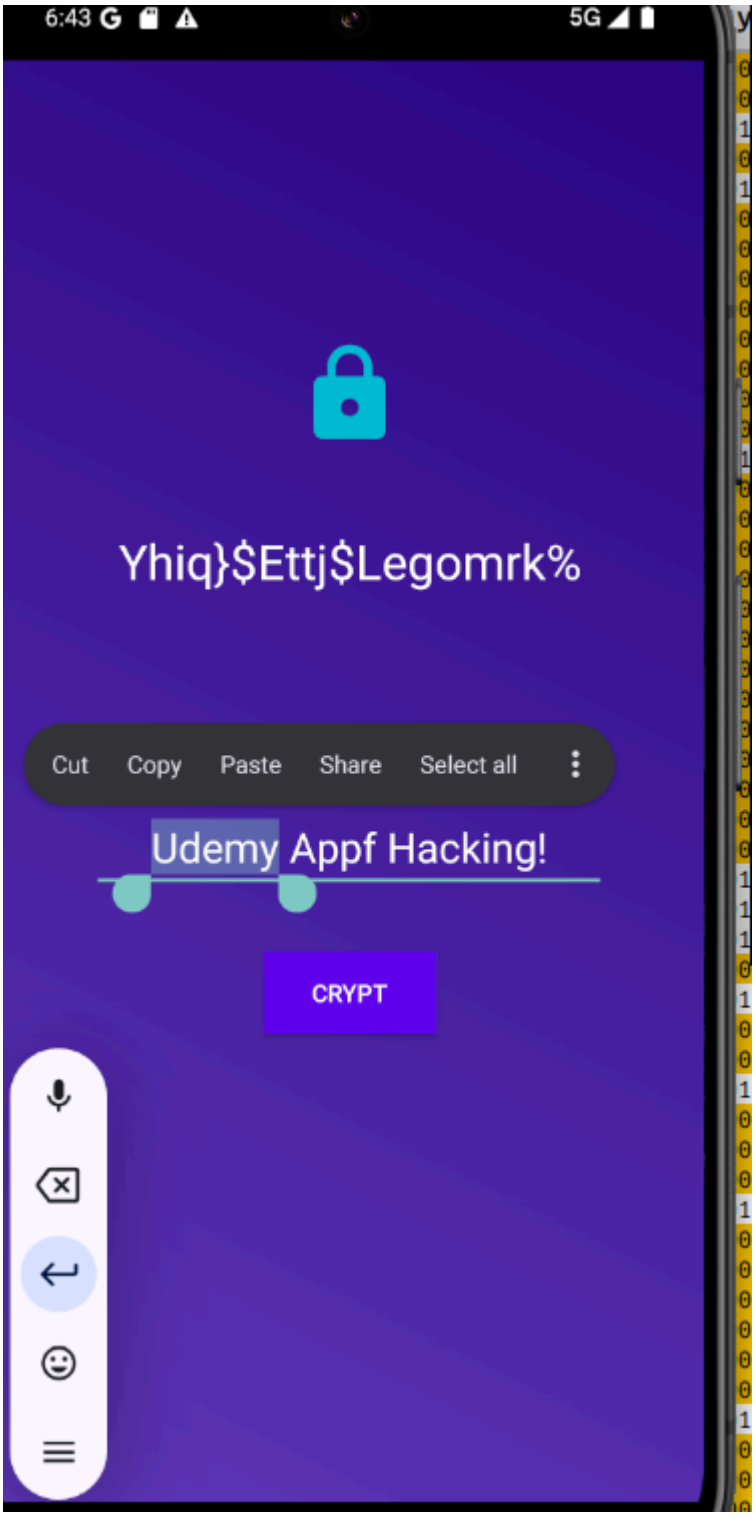
```java
public class MainActivity extends AppCompatActivity {

    String password = "App Hacking Udemy";
    int round = 3;

    // Used to load the 'native-lib' library on application startup.
    static {
        System.loadLibrary("native-lib");
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // Example of a call to a native method
        TextView tv = findViewById(R.id.sample_text);
        tv.setText(encryptString(password, round));
    }

    /**
     * A native method that is implemented by the 'native-lib' native library,
     * which is packaged with this application.
     */
    public native String encryptString(String pass, int round);
}
```

```java
/**
 * A native method that is implemented by the 'native-lib' native library,
 * which is packaged with this application.
 */
public native String decryptString(String pw, int round);
```

```cpp
#include <jni.h>
#include <string>

extern "C" JNIEXPORT jstring JNICALL
Java_com_apphacking_ndkfrida_MainActivity_decryptString(
        JNIEnv* env, jobject, jstring password, int rotation) {

    jboolean iscopy;
    int tmpValue = 0;

    const char* converted_string = (env)->GetStringUTFChars(password, &iscopy);
    std::string encryptedPW = converted_string;

    for(int i = 0; i < strlen(converted_string); i++) {
        tmpValue = (int) converted_string[i] + rotation;
        encryptedPW[i] = tmpValue;
    }

    return env->NewStringUTF(encryptedPW.c_str());
}
```

14- NDK hooking

هنا اهو في التطبيق ده بياخد text وبيعمله encrypt



in MainActivity code he contain (native-lib)

R | MainActivity | lib/arm64-v8a/libnative-lib.so

```java
package com.apphacking.ndkfrida;

import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

/* loaded from: classes2.dex */
public class MainActivity extends AppCompatActivity {
    EditText edittext;
    String password = "Udemy App Hacking!";
    TextView txtview;

    public native String decryptString(String str, int i);

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

    @Override // androidx.appcompat.app.AppCompatActivity, androidx.fragment.app.FragmentActivity,
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        this.edittext = (EditText) findViewById(R.id.editText);
        this.txtview = (TextView) findViewById(R.id.textView);
    }

    public void doThemagic(View view) {
        this.password = String.valueOf(this.edittext.getText());
        System.out.println(this.password);
        this.txtview.setText(decryptString(this.password, 4));
    }
}
```

first method : frida with java

if we use this code : will print the password and the int value when decrypt function implement

```
Java.perform(function(){

    var main_class=Java.use('com.apphacking.ndkfrida.MainActivity');
    main_class.decryptString.implementation=function(param1,param2){
        send("Param 1 is = : "+param1);
        send("Param 2 is = : "+param2);
        var returnValue=this.decryptString(param1,param2);
        send("return value is : "+returnValue);
        return returnValue;
    }
```

```
})
```

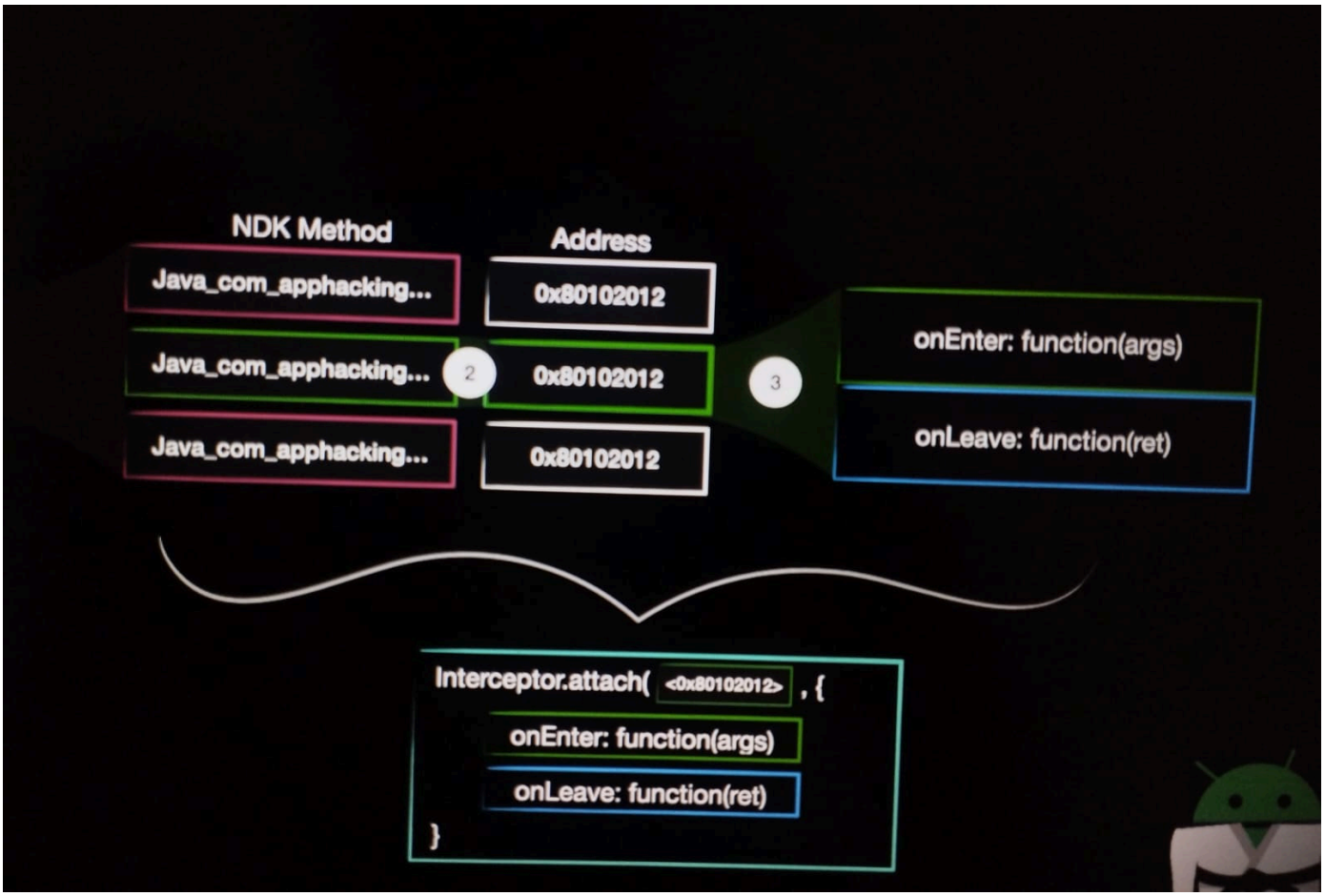
```
[Android Emulator 5554::ndkfrida ]-> Java.perform(function(){

    var main_class=Java.use('com.apphacking.ndkfrida.MainActivity');
    main_class.decryptString.implementation=function(param1,param2){
        send("Param 1 is = : "+param1);
        send("Param 2 is = : "+param2);
        var returnValue=this.decryptString(param1,param2);
        send("return value is : "+returnValue);
        return returnValue;
    }

})
[Android Emulator 5554::ndkfrida ]-> message: {'type': 'send', 'payload': 'Param 1 is = : Udemy Appf Hacking!'} data: No
ne
message: {'type': 'send', 'payload': 'Param 2 is = : 4'} data: None
message: {'type': 'send', 'payload': 'return value is : Yhiq}$Ettj$Legomrk%'} data: None
message: {'type': 'send', 'payload': 'Param 1 is = : Udemy Appf Hacking!'} data: None
message: {'type': 'send', 'payload': 'Param 2 is = : 4'} data: None
message: {'type': 'send', 'payload': 'return value is : Yhiq}$Ettj$Legomrk%'} data: None
[Android Emulator 5554::ndkfrida ]->
```

when we change the param2 to 0 he give me the plain text

```
Java.perform(function(){

    var main_class=Java.use('com.apphacking.ndkfrida.MainActivity');
    main_class.decryptString.implementation=function(param1,param2){
        send("Param 1 is = : "+param1);
        send("Param 2 is = : "+param2);
        var returnValue=this.decryptString(param1,0);
        send("return value is : "+returnValue);
        return returnValue;
    }

})
```

```
[Android Emulator 5554::ndkfrida ]-> Java.perform(function(){

    var main_class=Java.use('com.apphacking.ndkfrida.MainActivity');
    main_class.decryptString.implementation=function(param1,param2){
        send("Param 1 is = : "+param1);
        send("Param 2 is = : "+param2);
        var returnValue=this.decryptString(param1,0);
        send("return value is : "+returnValue);
        return returnValue;
    }

})
[Android Emulator 5554::ndkfrida ]-> message: {'type': 'send', 'payload': 'Param 1 is = : Udemy Appf Hacking!'} data: No
ne
message: {'type': 'send', 'payload': 'Param 2 is = : 4'} data: None
message: {'type': 'send', 'payload': 'return value is : Udemy Appf Hacking!'} data: None
```

second method : frida with C++/C



To get address :

Module.enumerateImports();

Module.enumerateExports();

Module.enumerateSymbols();

```
[Android Emulator 5554::ndkfrida ]-> Module.enumerateImports("libnative-lib.so")
[
    {
        "address": "0x70e6bc72e3b0",
        "module": "/apex/com.android.runtime/lib64/bionic/libc.so",
        "name": "__cxa_atexit",
        "slot": "0x70e40a87dca8",
        "type": "function"
    },
    {
        "address": "0x70e6bc72e660",
        "module": "/apex/com.android.runtime/lib64/bionic/libc.so",
        "name": "__cxa_finalize",
        "slot": "0x70e40a87dca0",
        "type": "function"
    },
```

```
    {
        "address": "0x70e6bc73cf80",
        "module": "/apex/com.android.runtime/lib64/bionic/libc.so",
        "name": "__register_atfork",
        "slot": "0x70e40a87dcb0",
        "type": "function"
    },
    {
        "address": "0x70e6bc75c210",
        "module": "/apex/com.android.runtime/lib64/bionic/libc.so",
        "name": "__stack_chk_fail",
        "slot": "0x70e40a87dce8",
        "type": "function"
    },
    {
        "address": "0x70e6bc773cf0",
        "module": "/apex/com.android.runtime/lib64/bionic/libc.so",
        "name": "__memcpy_chk",
        "slot": "0x70e40a87dd58",
        "type": "function"
    },
    {
        "address": "0x70e6bc753130",
        "module": "/apex/com.android.runtime/lib64/bionic/libc.so",
        "name": "memcpy",
        "slot": "0x70e40a87dd60",
        "type": "function"
    },
    {
```


to get address run Module.enumerateExports("libnative-lib.so")


```
    {
        "address": "0x70e40a853230",
        "name": "Java_com_apphacking_ndkfrida_MainActivity_decryptString",
        "type": "function"
    },
```

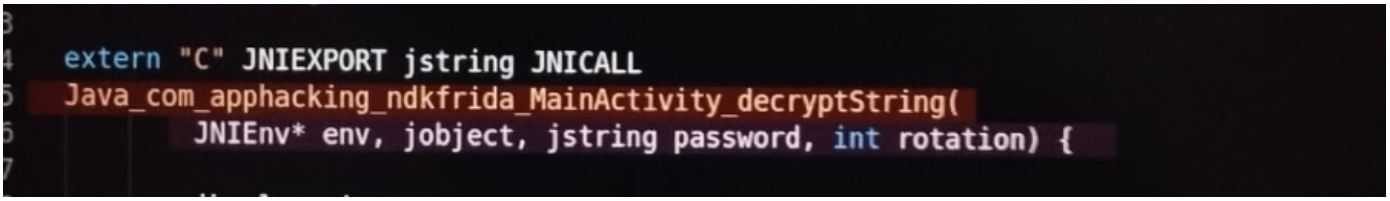

```
var exportString=Module.enumerateExports('libnative-lib.so');
exportString.forEach(function(element,index){
    if(element.name.includes('decryptString')){
        console.log(element.name);
        console.log(element.address);
    }
})
```

```
[Android Emulator 5554::ndkfrida ]-> var exportString=Module.enumerateExports('libnative-lib.so');
exportString.forEach(function(element,index){
    if(element.name.includes('decryptString')){
        console.log(element.name);
        console.log(element.address);
    }
})
Java_com_apphacking_ndkfrida_MainActivity_decryptString
0x70e40a853230
[Android Emulator 5554::ndkfrida ]->
```

now use Interceptor.attach

```
var address=0x0;
var exportString=Module.enumerateExports('libnative-lib.so');
exportString.forEach(function(element,index){
    if(element.name.includes('decryptString')){
        console.log(element.name);
        console.log(element.address);
        address=element.address;
    }
})
Interceptor.attach(address,{
    onEnter: function(arg){
        send("the decrypt string");

    },
    onLeave: function(ret){
        send("this is a return ");
    }
})
```

print the parameters : index 2 for password and 3 for rotation




```
var address=0x0;
var exportString=Module.enumerateExports('libnative-lib.so');
exportString.forEach(function(element,index){
    if(element.name.includes('decryptString')){
        console.log(element.name);
        console.log(element.address);
        address=element.address;
    }
```

```
})
Interceptor.attach(address,{
    onEnter: function(arg){
        var stringClassref=Java.use('java.lang.String');
        var stringInstance=Java.cast(ptr(arg[2]),stringClassref) ;
        send("the decrypt string");
        send("first paramter is "+stringInstance);
        send("second paramter is "+arg[3]toInt32());
    },
    onLeave: function(ret){
        send("this is a return ");
        send("the return value is "+ret);
    }
})
```

```
[Android Emulator 5554::ndkfrida ]-> var address=0x0;
var exportString=Module.enumerateExports('libnative-lib.so');
exportString.forEach(function(element,index){
    if(element.name.includes('decryptString')){
        console.log(element.name);
        console.log(element.address);
        address=element.address;
    }
})
Interceptor.attach(address,{
    onEnter: function(arg){
        var stringClassref=Java.use('java.lang.String');
        var stringInstance=Java.cast(ptr(arg[2]),stringClassref) ;
        send("the decrypt string");
        send("first paramter is "+stringInstance);
        send("second paramter is "+arg[3]);
    },
    onLeave: function(ret){
        send("this is a return ");
        send("the return value is "+ret);
    }
})
Java_com_apphacking_ndkfrida_MainActivity_decryptString
0x70e40a853230
{}
[Android Emulator 5554::ndkfrida ]-> message: {'type': 'send', 'payload': 'the decrypt string'} data
: None
message: {'type': 'send', 'payload': 'first paramter is eUdemy App Hacking '} data: None
message: {'type': 'send', 'payload': 'second paramter is 0x4'} data: None
message: {'type': 'send', 'payload': 'this is a return '} data: None
message: {'type': 'send', 'payload': 'the return value is 0x70e6d6926025'} data: None
```

هشنوف بقي ازاي نعدل علي القيم اللي هي0 to set it parmater2 بس لازم علشان نغيرها مش وهو في حالة running لازم نعملها start وبعد كده run script

1- start the app

```
➜  fridaNDK frida -U -f com.apphacking.ndkfrida
```

2- running this script

```
Interceptor.attach(address,{
    onEnter: function(arg){
        var class_string1=Java.use('java.lang.String');
        var print_string=Java.cast(ptr(arg[2]),class_string1);
        send("We are in the decryptstirng function right now,");
        send("the string we enter it to function decryptString is :
"+print_string);
        send("the int value before change it : "+arg[3].toInt32());
        arg[3]=ptr(0);
        send("the int value after  change it : "+arg[3].toInt32());
    },
    onLeave: function(ret){
        var class_string=Java.use('java.lang.String');
        var return_value=Java.cast(ret,class_string)
        send("We are leaving  the decryptstirng function right now,");
        send("the return value is : "+return_value);
    }
})
```

Ghidra

Ghidra is a software reverse engineering (SRE) framework. It is an open-source tool widely used by cybersecurity professionals, penetration testers, malware analysts, and software engineers to analyze binaries, decompile code, and study the inner workings of software.

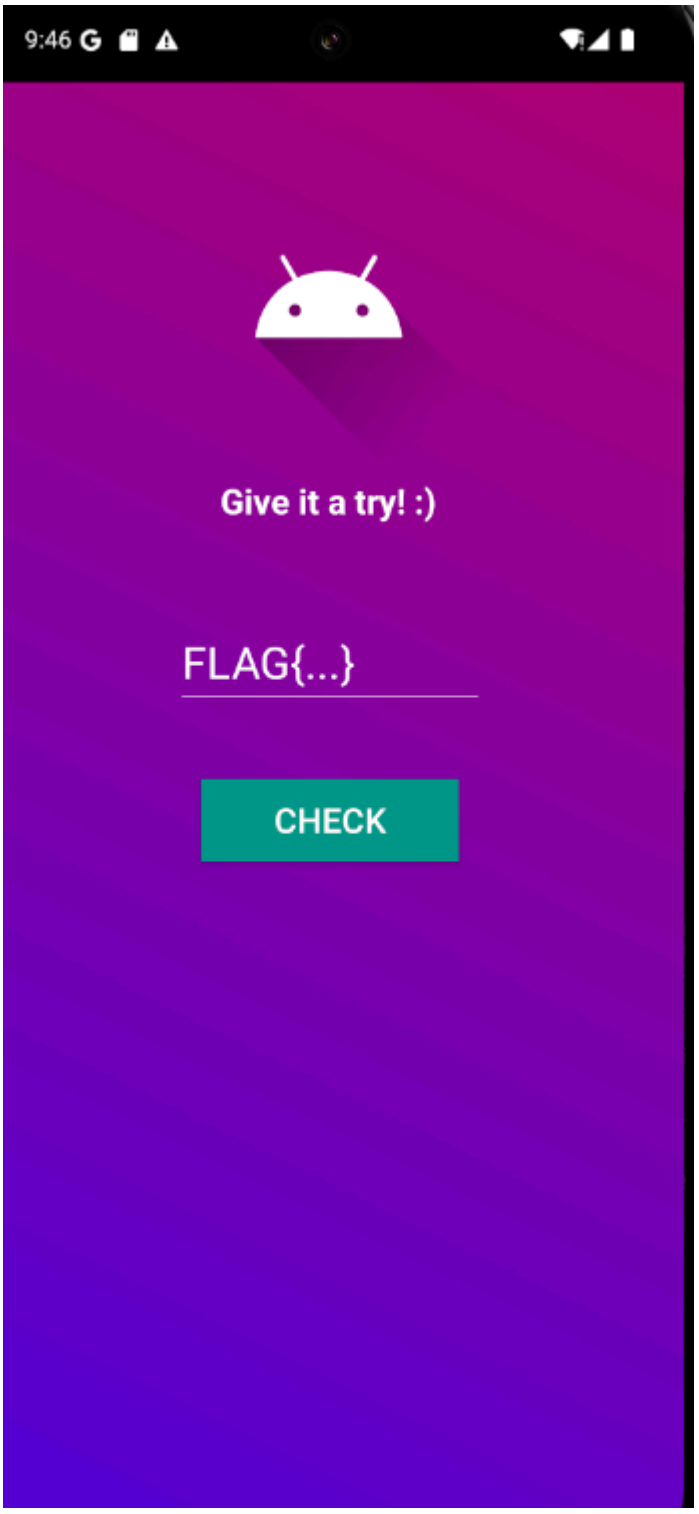
دي من خلالها نقدر نتعامل مع ملفات اللي هي library ومن خلالها نقدر نحلل التطبيق

install ghidra

```
-> wget
https://github.com/NationalSecurityAgency/ghidra/releases/download/Ghidra_11
.3.2_build/ghidra_11.3.2_PUBLIC_20250415.zip

-> unzip folder
-> ./tool
```

opne ghidra and select the library file after decompile the app --> /app/lib/lib_name

دلوقتي عاوزين نعرف ايه هو flag وهنا هو بياخد text وبيقارنها في الكلمة اللي هو حاطهها في library file



decompile this app and open library file with ghidra

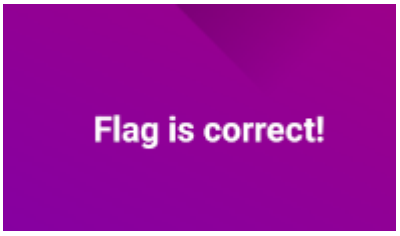
here : hideAndSeek take string and return true or fasle

```java
package com.apphacking.cfuncfrida;

import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

/* loaded from: classes2.dex */
public class MainActivity extends AppCompatActivity {
    EditText editText;
    String password;
    TextView txtView;

    public native String hideAndSeek(String str);

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

    @Override // androidx.fragment.app.FragmentActivity, androidx.activity.ComponentActivity, androidx.core.
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        this.editText = (EditText) findViewById(R.id.edtText);
        this.txtView = (TextView) findViewById(R.id.textView);
    }

    public void checkPW(View view) {
        String obj = this.editText.getText().toString();
        this.password = obj;
        this.txtView.setText(hideAndSeek(obj));
    }
}
```

open library file wiht ghidra and find flag

FLAG{INSANE_FRIDA_HOOKS}

```
undefined8
Java_com_apphacking_cfuncfrida_MainActivity_hideAndSeek
          (_JNIEnv *param_1,undefined8 param_2,_jstring *param_3)

{
  long lVar1;
  int iVar2;
  char *pcVar3;
  undefined8 local_58;
  basic_string<> abStack_50 [24];
  basic_string<> abStack_38 [28];
  uchar auStack_1c [4];
  long local_18;

  lVar1 = tpidr_el0;
  local_18 = *(long *)(lVar1 + 40);
  pcVar3 = (char *)_JNIEnv::GetStringUTFChars(param_1,param_3,auStack_1c);
  std::__ndk1::basic_string<>::basic_string<>(abStack_38,"Flag is correct!");
                    // try { // try from 0010f090 to 0010f09f has its CatchHandler @ 0010f110
  std::__ndk1::basic_string<>::basic_string<>(abStack_50,"Flag is wrong!");
  iVar2 = strcmp(pcVar3,"FLAG{INSANE_FRIDA_HOOKS}");
  if (iVar2 == 0) {
    pcVar3 = (char *)FUN_0010f29c(abStack_38);
                    // try { // try from 0010f0e0 to 0010f157 has its CatchHandler @ 0010f11c
    local_58 = _JNIEnv::NewStringUTF(param_1,pcVar3);
  }
  else {
    pcVar3 = (char *)FUN_0010f29c(abStack_50);
    local_58 = _JNIEnv::NewStringUTF(param_1,pcVar3);
  }
  std::__ndk1::basic_string<>::~basic_string(abStack_50);
  std::__ndk1::basic_string<>::~basic_string(abStack_38);
  lVar1 = tpidr_el0;
  if (*(long *)(lVar1 + 40) == local_18) {
    return local_58;
  }
                    // WARNING: Subroutine does not return
  __stack_chk_fail();
}
```



دلوقتي بقي احنا عرفنا نجيب Flag باستخدام ghidra دلوقتي عاوزين نستخدم Frida

we need to get :

1-Get the address of the strcmp function

2-interceptor.attach(address,callback)

3-print the paramters

1- get the address

```
[Android Emulator 5554::cfuncfrida ]-> var address;
var exportString=Module.enumerateImports('libnative-lib.so');
```

```
exportString.forEach(function(element,index){
    if(element.name.includes('strcmp')){
        console.log(element.name);
        console.log(element.address);
        address=element.address;
    }
})

strcmp
0x71fc256e6790
```

2-interceptor.attach(address,callback)

```
Interceptor.attach(address,{
    onEnter:function(arg){
        var userInput=Memory.readUtf8String(arg[0]);
        if(userInput.includes('FLAG')){
            send("Parmater 1 : "+Memory.readUtf8String(arg[0]));
            send("Parmater 2: "+Memory.readUtf8String(arg[1]));

        }
    },
    onLeave:function(ret){
        // send("the end of function");
    }
})
```

3-print the paramters

```
[Android Emulator 5554::cfuncfrida ]-> var address;
var exportString=Module.enumerateImports('libnative-lib.so');
exportString.forEach(function(element,index){
    if(element.name.includes('strcmp')){
        console.log(element.name);
        console.log(element.address);
        address=element.address;
    }
})
Interceptor.attach(address,{
    onEnter:function(arg){
        var userInput=Memory.readUtf8String(arg[0]);
        if(userInput.includes('FLAG')){
            send("Parmater 1 : "+Memory.readUtf8String(arg[0]));
            send("Parmater 2: "+Memory.readUtf8String(arg[1]));

        }
    },
    onLeave:function(ret){
        // send("the end of function");
    }
})

strcmp
0x71fc256e6790
{}
[Android Emulator 5554::cfuncfrida ]-> message: {'type': 'send', 'payload': 'Parmater 1 : FLAG{INSANE_FRIDA_HOOKS}'} dat
a: None
message: {'type': 'send', 'payload': 'Parmater 2: FLAG{INSANE_FRIDA_HOOKS}'} data: None
message: {'type': 'send', 'payload': 'Parmater 1 : FLAG{INSANE_FRIDA_HOOKS}'} data: None
message: {'type': 'send', 'payload': 'Parmater 2: FLAG{INSANE_FRIDA_HOOKS}'} data: None
```

Full code

```
var addressOfStrcmp;

// Get the address of the strcmp function
var importFunctionArray = Module.enumerateImports("libnative-lib.so");

importFunctionArray.forEach(function(element) {

    if (element.name.includes("strcmp")) {
        console.log(element.name);
        console.log(element.address);

        addressOfStrcmp = element.address;
    }

})

// introducing a new state to only print out the return value of our strcmp
var isOurFunction = false;

// Interceptor.attach(address, callbacks)
Interceptor.attach(addressOfStrcmp, {

    // args[0] --> our value
    // args[1] --> value of the flag
```

```
    onEnter: function(args) {

        var userInput =  Memory.readUtf8String(args[0]);

        if (userInput.includes('1234567980')) {
            //send("param1 = " + Memory.readUtf8String(args[0]));
            //send("param2 = " + Memory.readUtf8String(args[1]));
            send("param1 = " + Memory.readCString(args[0]));
            send("param2 = " + Memory.readCString(args[1]));

            isOurFunction = true;
        }
    },
    onLeave: function(ret) {
        // verifying that this is our strcmp which we are intersted in.
        if(isOurFunction) {
            send("return value " + ret.toInt32());
        }

        // setting it back to false --> otherwise all return values
afterwards would be printed out.
        isOurFunction = false;
    }

})

// Print out these parameters.
```