

# Android Root Detection Evasion

Roberto Castellotti

Supervised by: prof. Giovanni Lagorio

Università degli Studi di Genova

2022

# Introduction to Android

- ▶ introduzione android
- ▶ introduzione "apk" <https://developer.android.com/guide/components/fundamentals> mobisec native code
- ▶ spiegare come si passa da sorgente java a bytecode
- ▶ setup root phone (lineage, magisk, twrp, tool vari)
- ▶ root: perche' farlo
- ▶ root detection: perche'
- ▶ root detection: come <https://github.com/scottyab/rootbeer>
- ▶ approfondimento signature app (in particolare patchare con frida meglio perche' firma vera)
- ▶ tool usati

# patch-and-reinstall

- ▶ `adb pull /data/com/app/com.package.name app.apk`
- ▶ `apktool d app.apk -output app`
- ▶ `patch`
- ▶ `apktool b app -output rebuilt-app.apk`
- ▶ `zipalign -f -p 4 rebuilt-app.apk aligned-rebuilt-app.apk`
- ▶ `apksigner sign --ks /key.jks aligned-rebuilt-app.apk`
- ▶ `adb install aligned-rebuilt-app.apk`
- ▶ this approach does not require the android device to be rooted

## patch-and-reinstall: an example

The patching process is usually a matter of finding the methods performing root detection and patching them, here is an example from a popular financial services company's application:

```
Author: rcastellotti <me@rcastellotti.dev> 2022-06-29 01:17:32
Committer: rcastellotti <me@rcastellotti.dev> 2022-06-29 01:17:32
Parent: 4e6c812897a3d9e1187e9fafd77619b8f2b0540a (decompiled)
Child: 28844736e60b1fb98792750e4d4ed48afd41b143 (added source patch)
Branches: main, remotes/origin/main
Follows:
Precedes:
```

```
    i cant really believe this is the patch
```

```
                /smali_classes2/com/                                /util/WuRootUtil.smali
index 6460600e0..866aaf619 100644
```

```
@@ -329,7 +329,7 @@
```

```
        :cond_1
        :goto_0
-       const/4 v0, 0x1
+       const/4 v0, 0x0
```

```
        :goto_1
        return v0
```

Figure: patching `.method public static isDeviceRooted()`

# frida: a dynamic toolkit

- ▶ provides ability to inject scripts into black box processes.
- ▶ portable (Windows, macOS, GNU/Linux, iOS, Android)
- ▶ **injected mode:** `frida-server`: `frida-core` over TCP
- ▶ `frida-core`: a layer that packages up GumJS into a shared library that it injects into existing software, and provides a two-way communication channel for talking to your scripts.
- ▶ **on device:** `./frida-server-15.1.27-android-arm64`
- ▶ **on computer:** `frida -U -l script.js -f com.package.name`
- ▶ this approach could be better since it allows to patch applications without losing original package signature

## tools used

- ▶ [Android Studio](#)
- ▶ [iBotPeaches/Apktool](#): A tool for reverse engineering Android apk files
- ▶ [skylot/jadx](#): Dex to Java decompiler
- ▶ [scottyab/rootbeer](#): Simple to use root checking Android library and sample app