Android Undercover: Native Code Translation for AV Stealth

LaurieWired

whoami

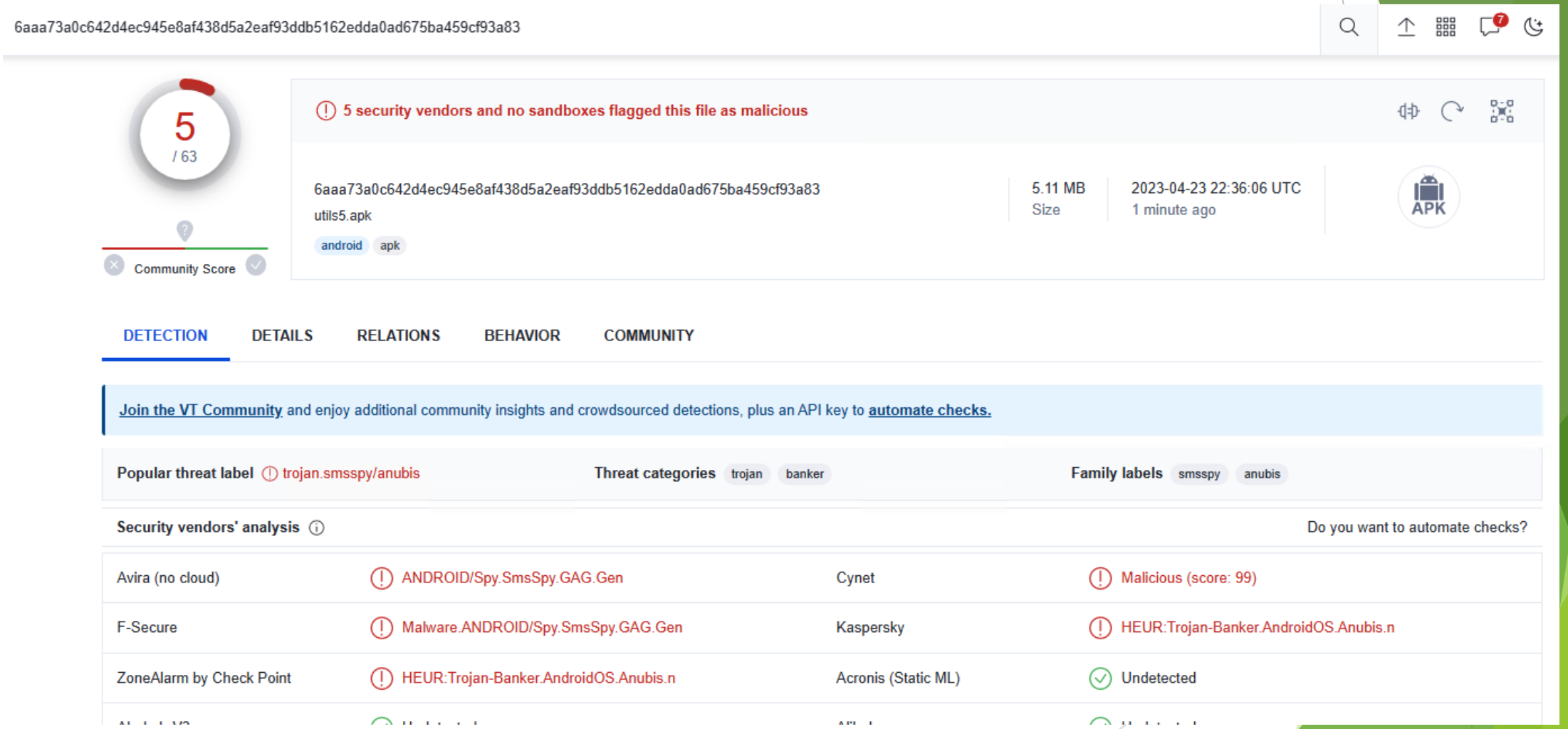
- Reverse Engineer
- Specialize in cross-platform malware with a focus on mobile malware
- Run YouTube channel @lauriewired
  - Explores mobile reverse engineering
- Representing myself as an individual security researcher today

@lauriewired

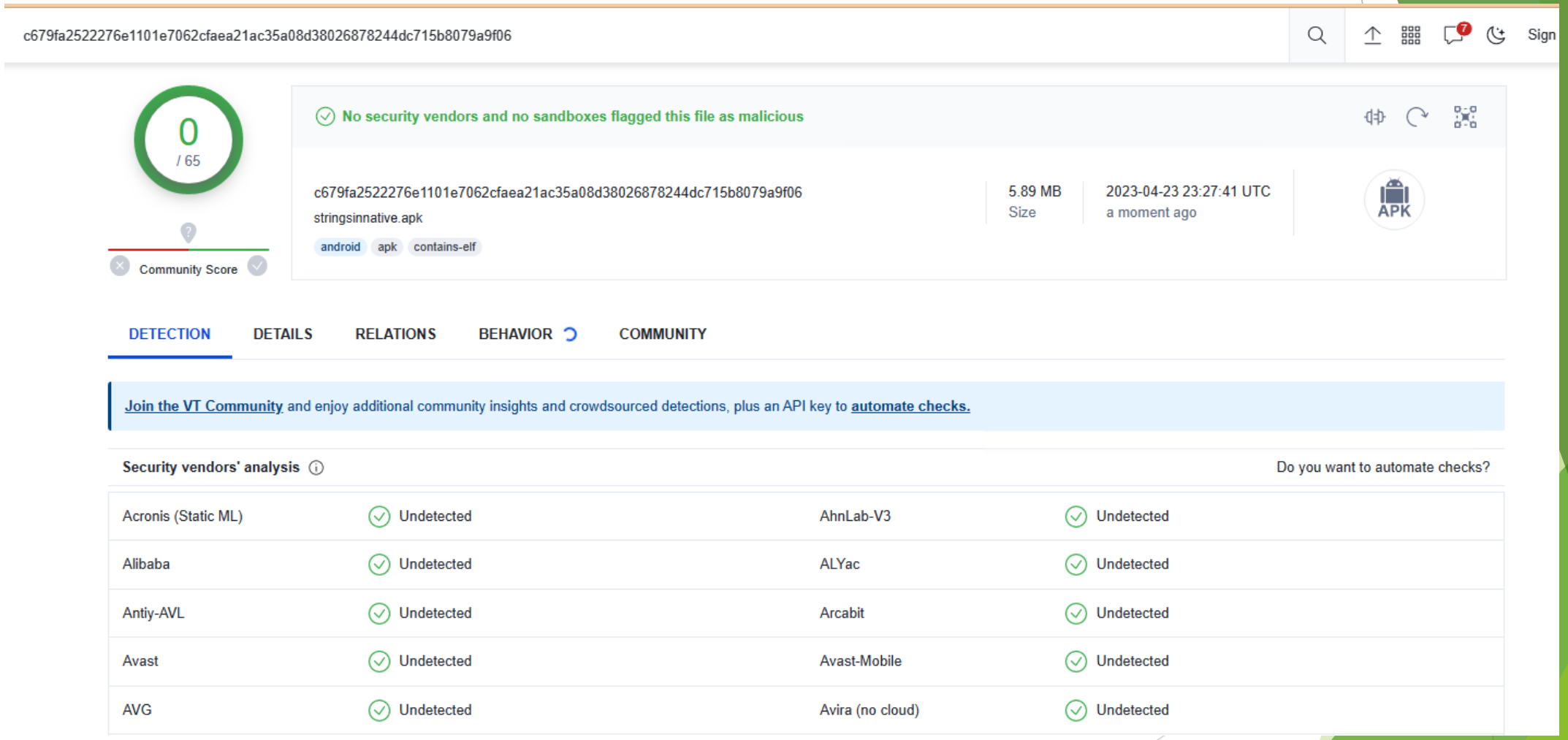
Follow Along with GitHub

- https://github.com/LaurieWired/AndroidNativeObfuscation_defcon615

The only difference between this app...



… and this app


c679fa2522276e1101e7062cfaea21ac35a08d38026878244dc715b8079a9f06
Sign
0
/ 65
Community Score
No security vendors and no sandboxes flagged this file as malicious
c679fa2522276e1101e7062cfaea21ac35a08d38026878244dc715b8079a9f06
stringsinnative.apk
android
apk
contains-elf
5.89 MB
Size
2023-04-23 23:27:41 UTC
a moment ago
APK
DETECTION
DETAILS
RELATIONS
BEHAVIOR
COMMUNITY
Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.
Security vendors' analysis
Do you want to automate checks?
Acronis (Static ML) Undetected
AhnLab-V3 Undetected
Alibaba Undetected
ALYac Undetected
Antiy-AVL Undetected
Arcabit Undetected
Avast Undetected
Avast-Mobile Undetected
AVG Undetected
Avira (no cloud) Undetected

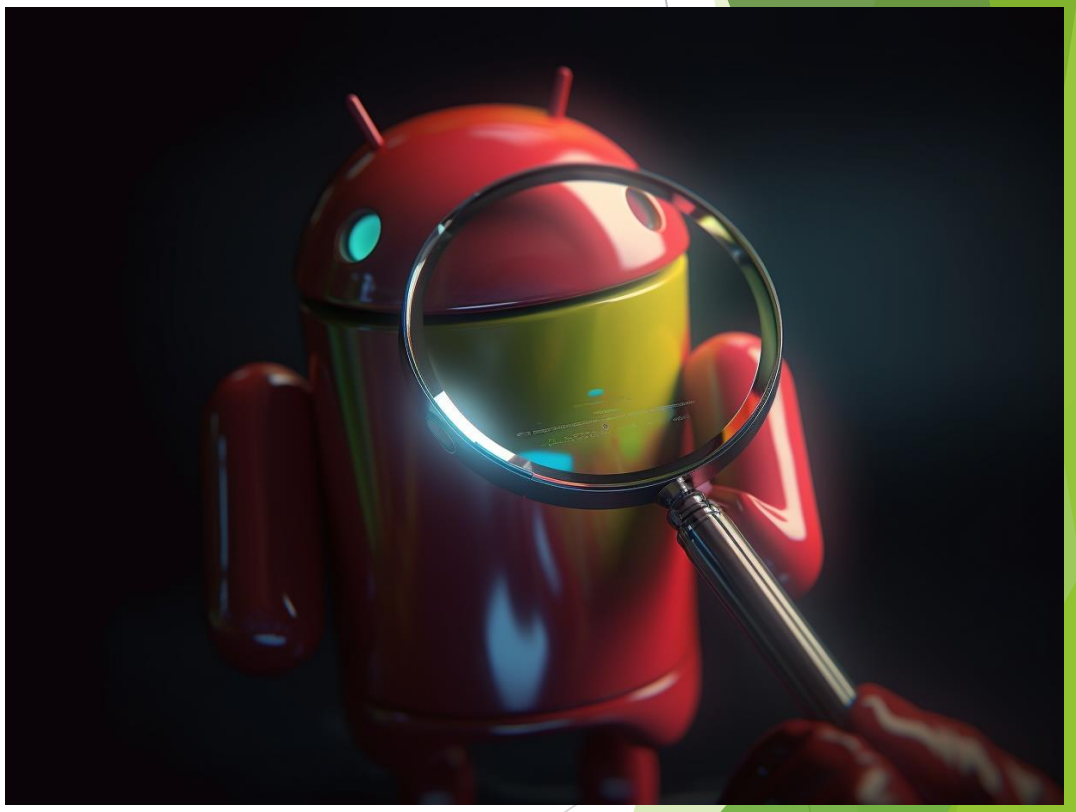
is partial native code implementation.

Quick background: let's define obfuscation

- Obfuscation obscures app data and functionality
- Common among all platforms
- Offensive and defensive motivations for obfuscation
- Essential for Android
  - Decompiled into pretty Java code

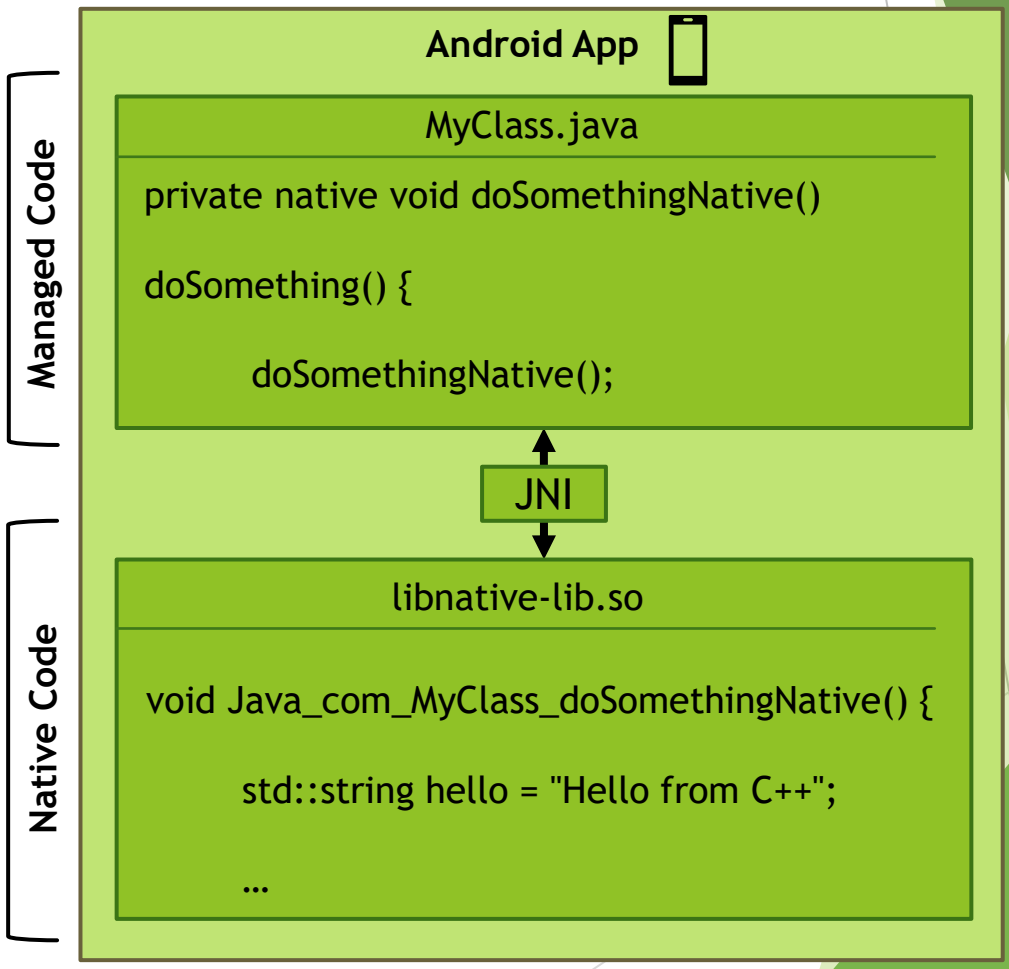
Native as an Obfuscation Technique

- Native code can hide malicious code
- More challenging to reverse engineer
  - Read assembly instead of Java
  - Understand JNI invocations
- Limit support for particular instruction set architectures
  - Remove x86 support



Android Native Code

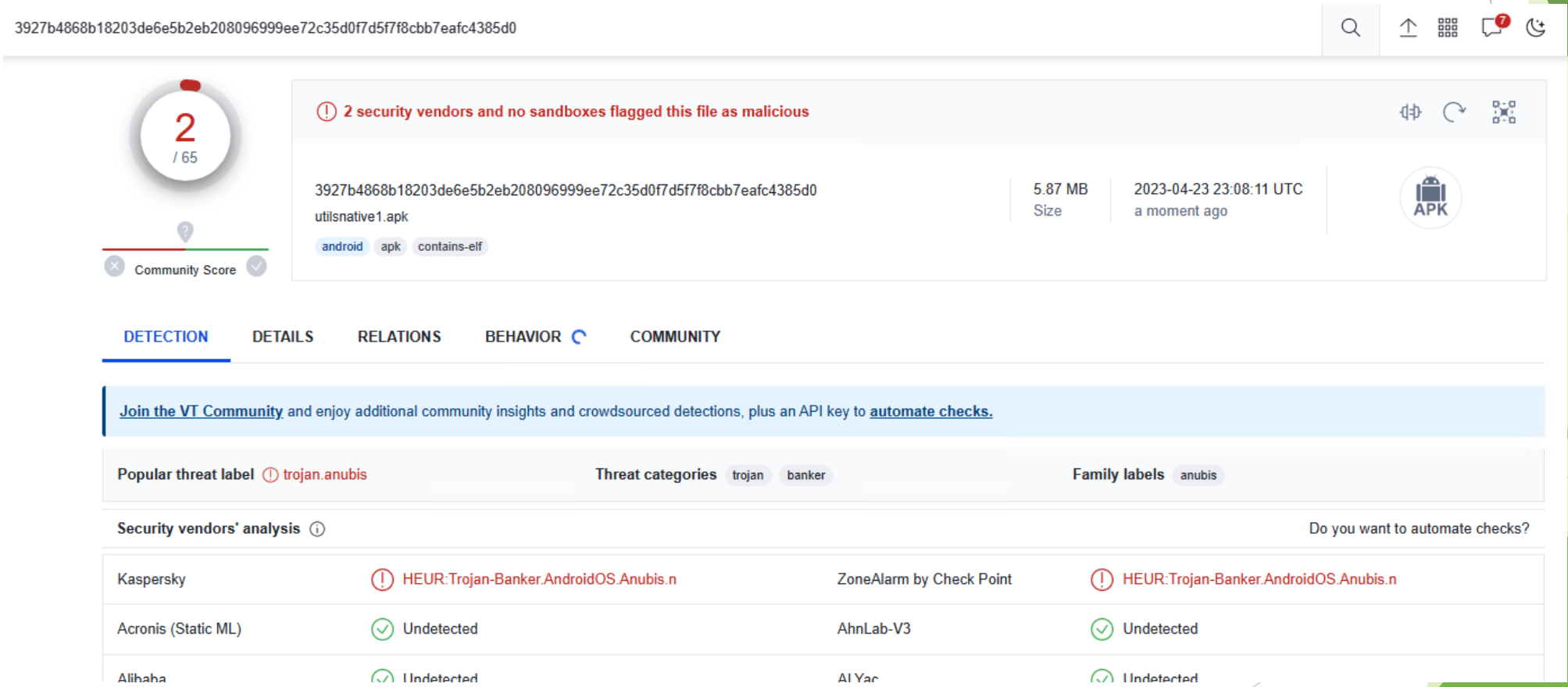
- Android apps are written in Java / Kotlin
- Apps can also include native code
  - C/C++ code
- Communicate via the JNI



Hands-on time: Let’s dive in!

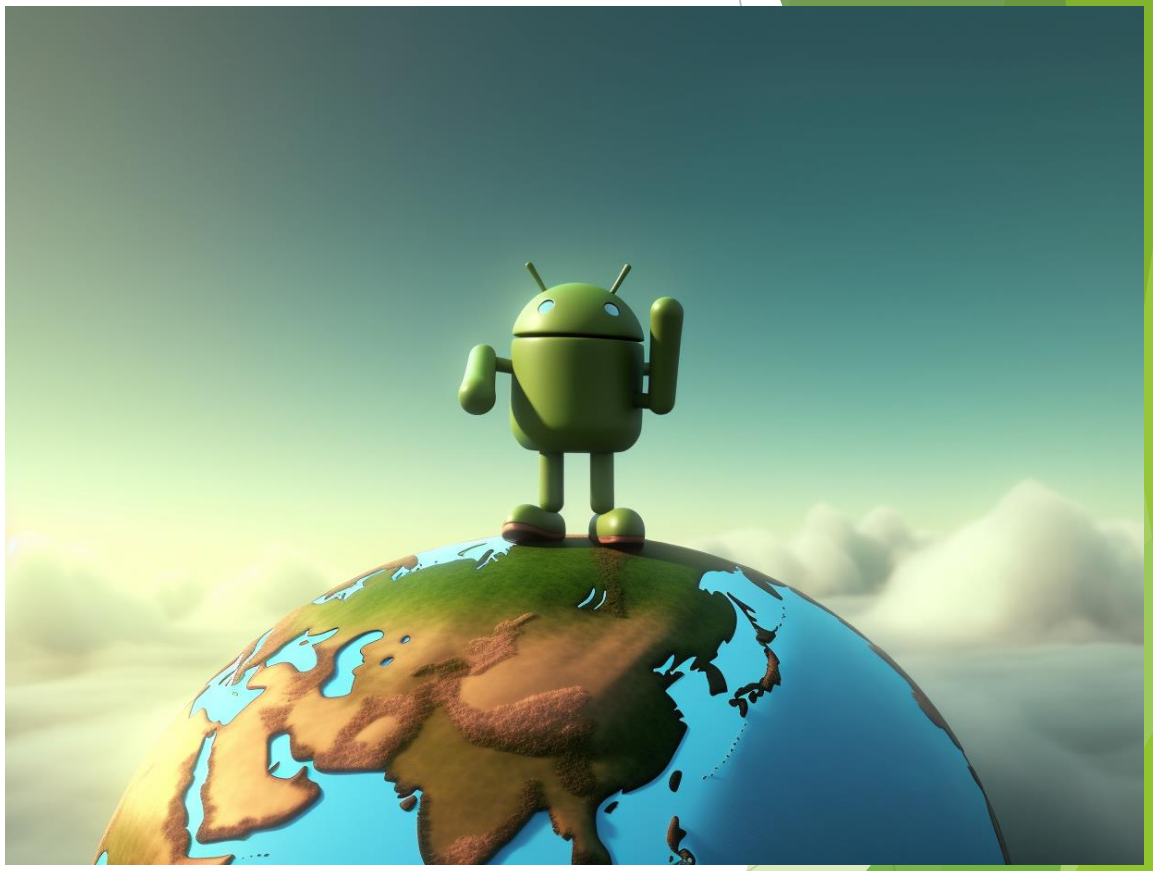
1. Put on our black hat to understand native code authoring
    a) Using Android Studio
2. Upload to VirusTotal to check detections (or lack thereof)
3. Put back on our blue hat to reverse engineer the native code
    a) Using JADX and Ghidra

Funnily, just the presence of native code removes some smaller AV detections



Real-World Use-Cases

- Commonly used in custom packers
  - Use native code to decrypt additional DEX file
  - Hide additional Java code invocations
- Not generally used by script kiddies
  - Too complex to implement



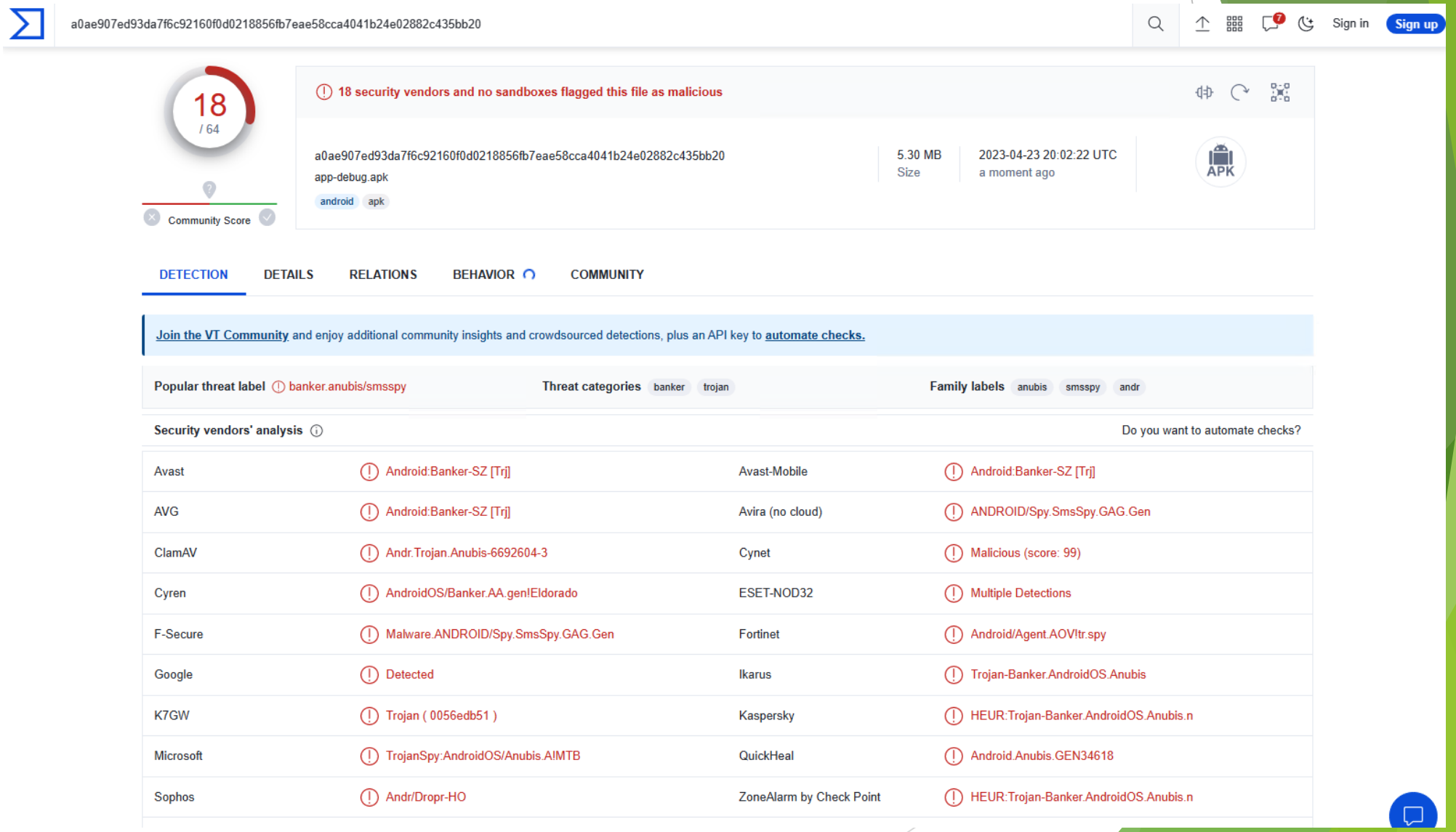
Next Steps

If you want more, try taking another Anubis method and converting it to native code.

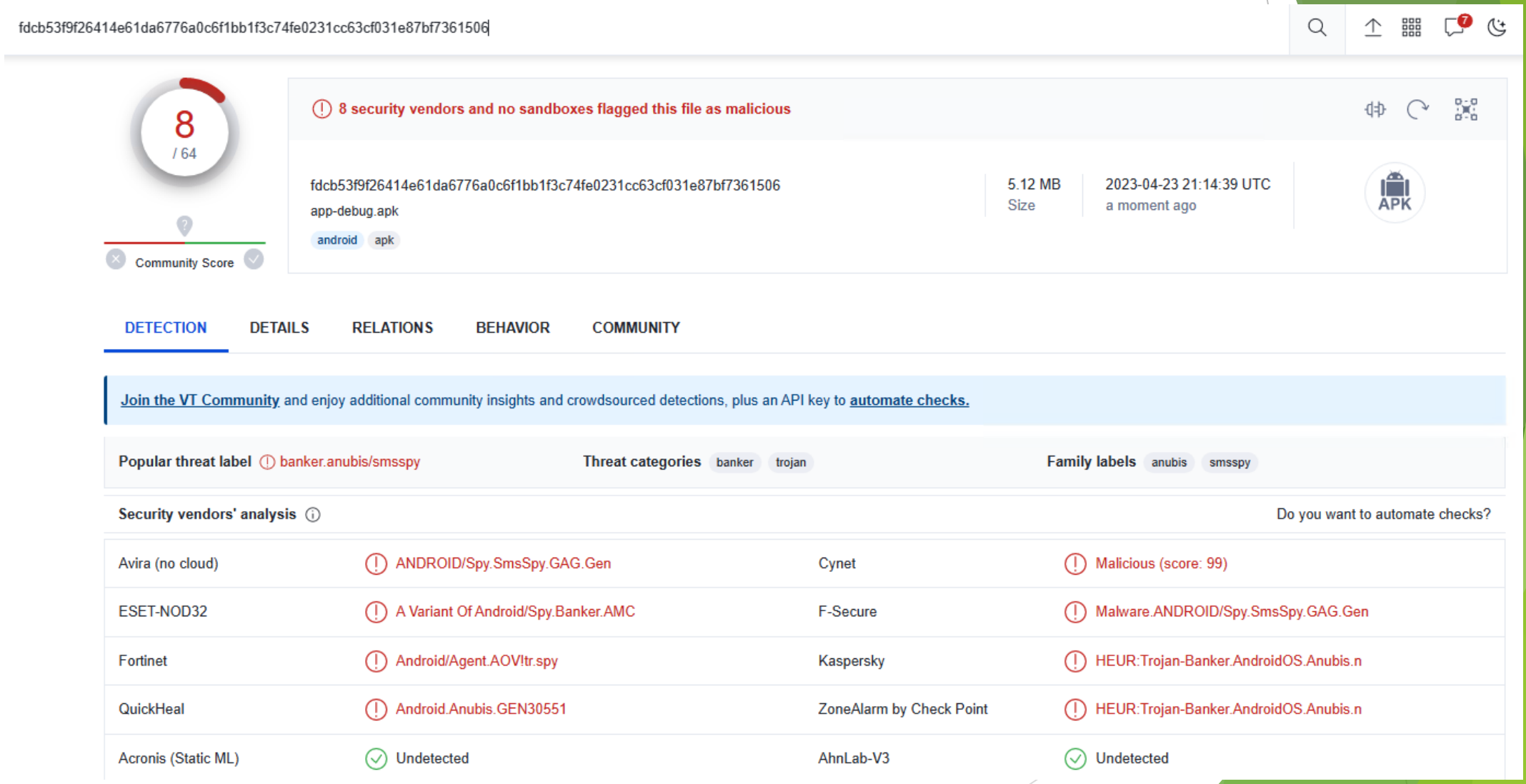
Thank you!

Bonus Section

Original Java Detections Full Sample



First step removing partial codebase



Anubis-specific detections gone when removing target method