ESORICS 2022

A Tale of Four Gates

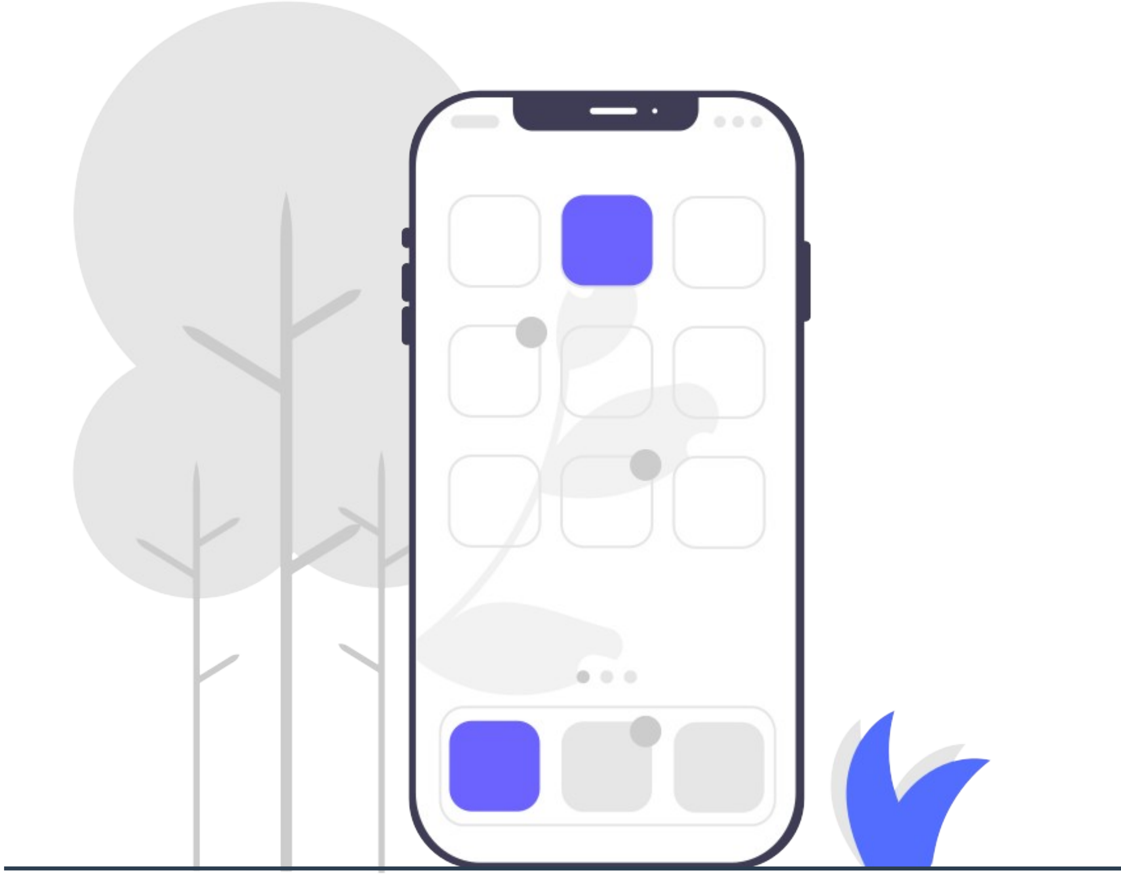
Privilege Escalation and Permission Bypasses on Android through App Components

Presented by

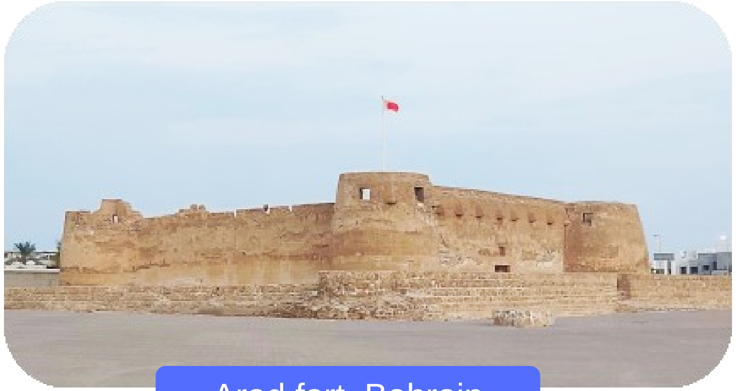
Abdulla Aldoseri
University of Birmingham
axa1170@bham.ac.uk

David Oswald
University of Birmingham
d.f.oswald@bham.ac.uk

Robert Chiper
University of Birmingham
robert.chiper@pm.me

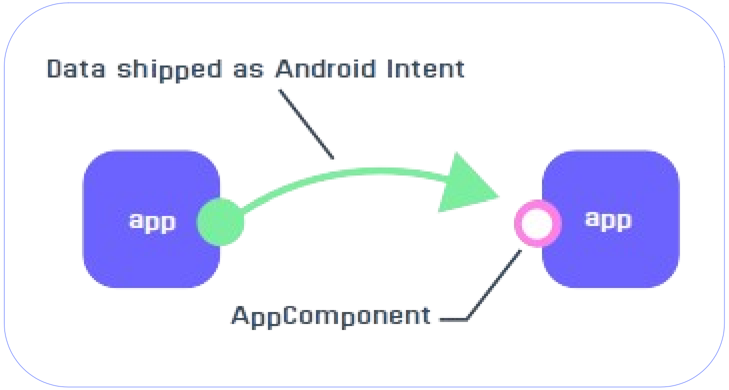


Introduction

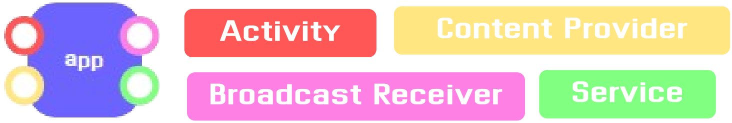


Arad fort, Bahrain

Forts in the past with secure gates

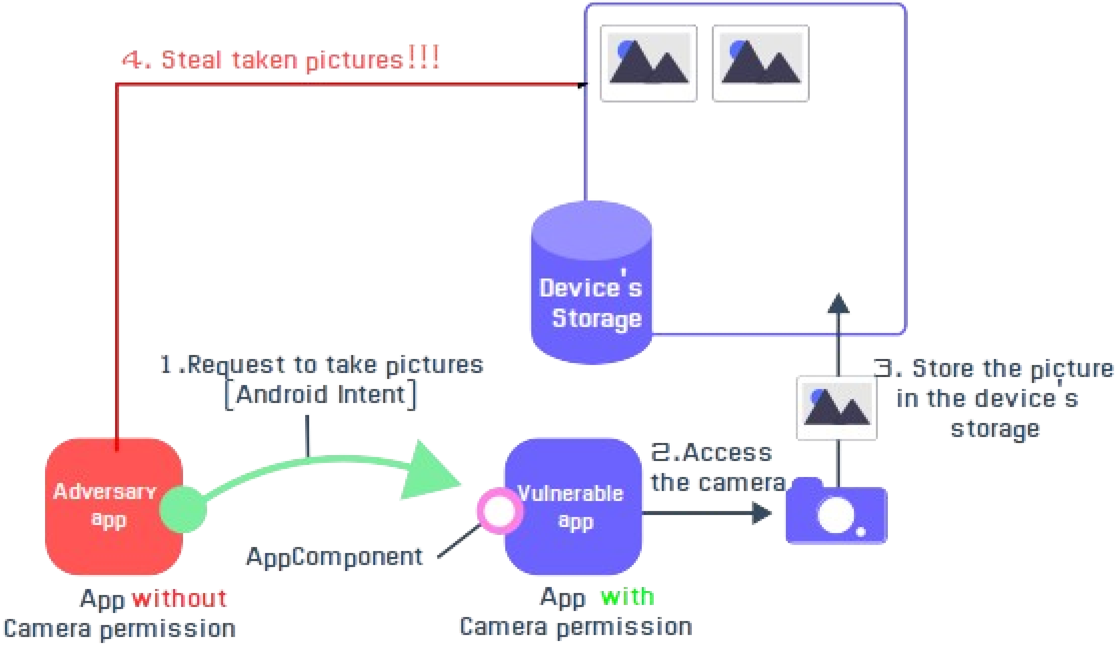


An app sending data as an Android intent to another app via its app component



Introduction

Confused deputy issue in app components


4. Steal taken pictures!!!
Device's Storage
1.Request to take pictures
[Android Intent]
3. Store the picture in the device's storage
Adversary app
2.Access the camera
Vulnerable app
AppComponent
App without Camera permission
App with Camera permission

An app without camera permission, abuse other app via its app components to use the camera

Confused deputy issue: Where an app that does not has a permission relies on other apps to use the permission on their behalf.

Similar scenario could happen for other device's permission like GPS and microphone .. etc.

The issue is well studied in academic at application level but not across system services.

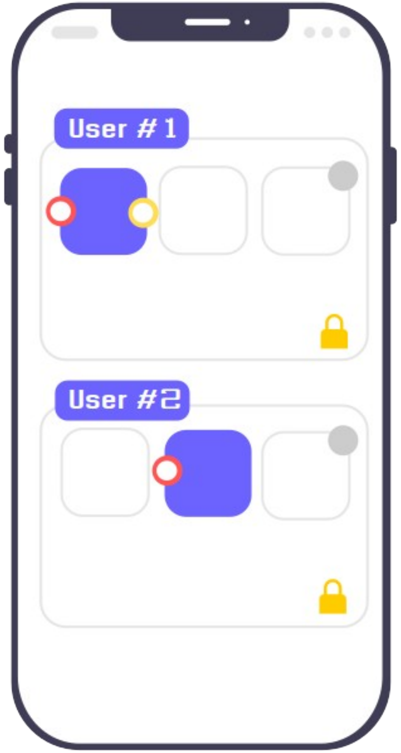
Introduction

Research question

Do confused deputy issues in mobile applications break the integrity of Android OS system level services?

Introduction

Contributions



#1 Analysis of Mutli-user feature

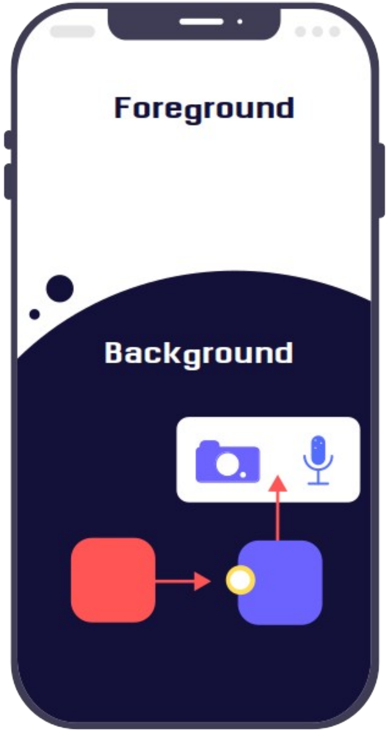
Moderate severity

We analyse the main four implementations of the Android multi-user feature (Samsung secure folder, Huawei private space, Xiaomi second space, Google multi-user) and show how to bypass Android lock screen protection in them, giving full access to an adversary through app components.

CVE-2020-9119 by Samsung

CVE-2020-26606 by Huawei

Unpatched in Google Devices



#2 Analysis of background restriction

High severity fixed (Android 11)

We show how spyware can use app components to stealthily access camera and microphone in the background, breaking the OS countermeasures against such techniques up to Android 10.

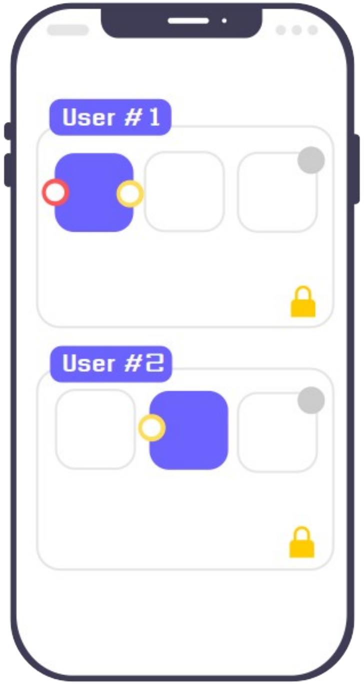
issue-175232797 by Google

#3 Four Gates Inspector

We present Four Gates Inspector, our open-source static analysis tool to detect the use of specific APIs (e.g., sensor access) in app component handlers. Four Gates identified confused deputy issues in 34 (benign) apps (out of a sample of 5,783) downloaded from F-Droid with an average runtime of 4.3 s per app.

Issue # 1

Analysis of app components across user profiles



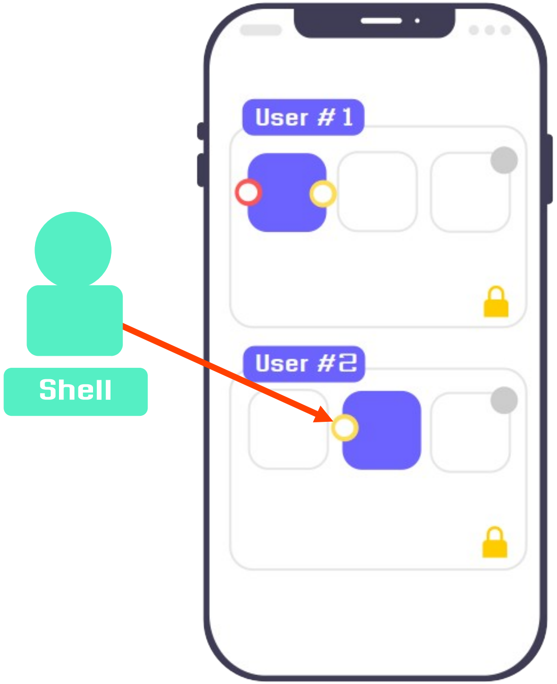
Multi-user is an Android feature that allows to set up several isolated user profiles on a single device. Each profile has a workplace to store data and install apps

Apps can only interact with each other via app components if they are in the same profile.

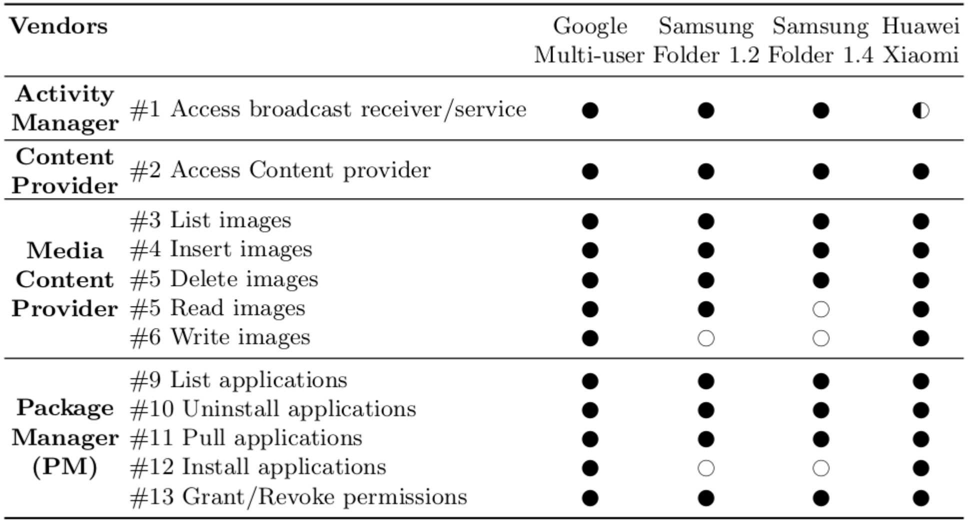
System-level permissions namely: INTERACT ACROSS USERS FULL, INTERACT ACROSS USERS, ACCESS CONTENT PROVIDER EXTERNALLY allow apps to communicate with app components of other apps in other user profiles.

Issue # 1

Analysis of app components across user profiles



Shell user does not have read/write access to the second user profile(Private space) but can access exported app components of that profile.

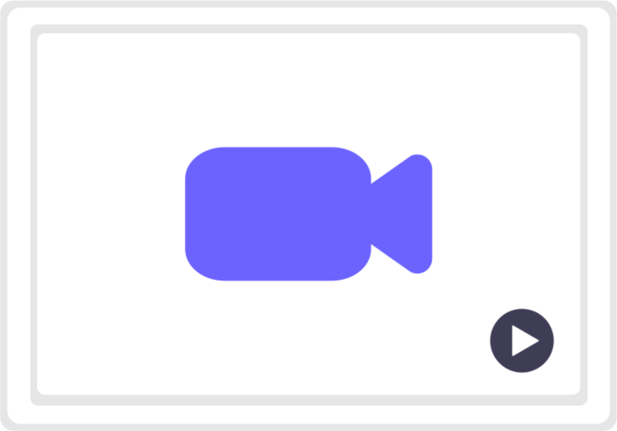


Vendors		Google	Samsung	Samsung	Huawei
		Multi-user	Folder 1.2	Folder 1.4	Xiaomi
Activity Manager	#1 Access broadcast receiver/service	●	●	●	◐
Content Provider	#2 Access Content provider	●	●	●	●
Media Content Provider	#3 List images	●	●	●	●
	#4 Insert images	●	●	●	●
	#5 Delete images	●	●	●	●
	#5 Read images	●	●	○	●
	#6 Write images	●	○	○	●
Package Manager (PM)	#9 List applications	●	●	●	●
	#10 Uninstall applications	●	●	●	●
	#11 Pull applications	●	●	●	●
	#12 Install applications	●	○	○	●
	#13 Grant/Revoke permissions	●	●	●	●

Table 1. MU attacks across vendor implementation. (●) exploited; (○) N/A; (◐) untested

Issue # 1

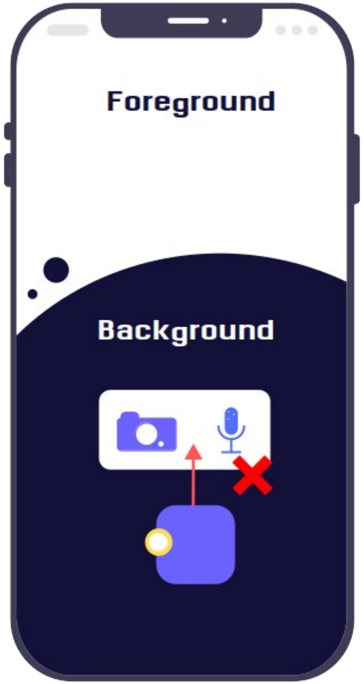
Analysis of app components across user profiles



Watch demo

Issue #2

Analysis of sensor background access



Background restriction policy

Android OS prevent apps to access microphone and camera in the background even if they have the required sensor permissions to ensure user privacy [1] unless the usage was through a foreground services that display a noticeable persistent notification.

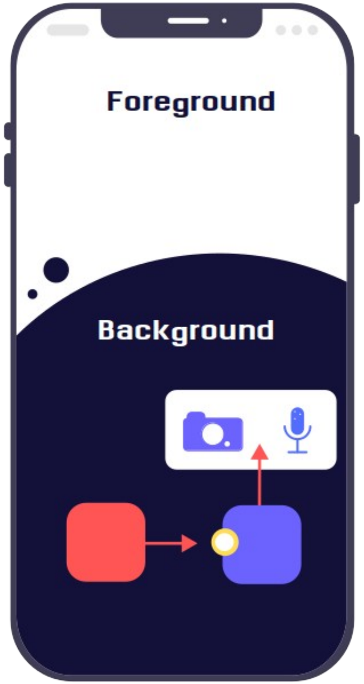
But some components do not have UI,
some app components (Service, provider and Receiver) can be invoked and run but do not have interfaces to display.

So what is background in Android OS ??

[1] Android: Behavior changes: all apps. https://developer.android.com/about/versions/pie/android-9.0-changes-all (2020), accessed on 09/13/2020

Issue #2

Analysis of sensor background access



The definition of background

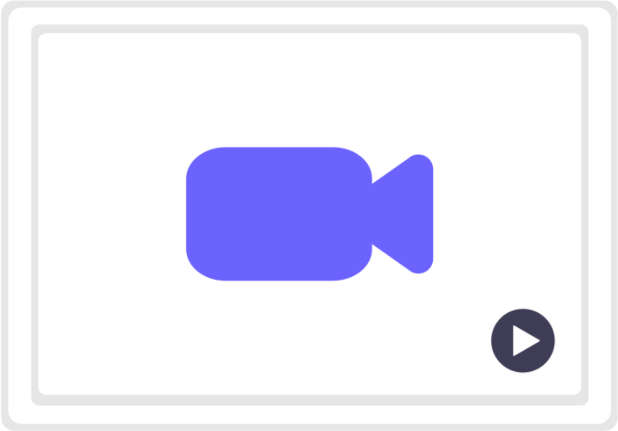
“Background” in Android OS does not refer to a single state, but actually a range of importances and an invocation to an app component does not put its app in the background.

Stealthy spyware

With an exception to Activity, we found that all three app components were capable to access the camera with the through invocation from another app in the background, while the microphone was accessible from the service and content provider. No notification is shown to the user, making exploitation of this issue stealthy.

Issue #2

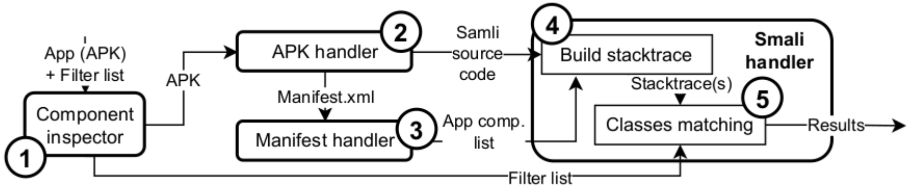
Analysis of sensor background access



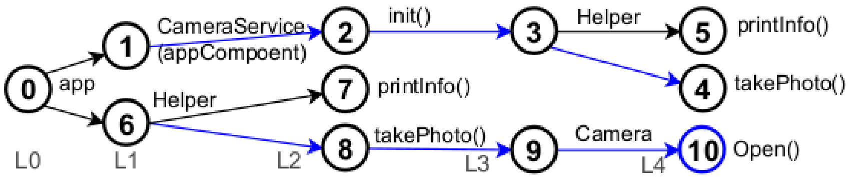
Watch demo

Four Gates Inspector

Evaluation



The architecture of Four gate inspector



A generated stacktrace by Four Gate Inspector to detect camera usage in an app component

What is Four Gates Inspector?

It is a statically analysis tool that detect confused deputy issues based on the usage of given classes and methods. It is based on analysing the execution trace of the decompiled Smali code of app components and provides fine-grained control over the analysis scope on the class level to detect non-pemrission classes.

Performance

As results, our tool managed to identify exposed components issues in 34 apps (out of 5783) with average analysis runtime of only 4.3 s per app on average which considered faster than AppSealer and Firmscope.

Discussion and mitigation

Mitigation for Mutli-user issue (issue #1)

Removing the permission from the shell user or sanitise the user flag of system binaries are both sufficient to solve the issue.

Samsung seems applied the second approach, Huawei' verify the password of private's space and warns user while enabling developer mode. While, Google consider it as interned behavior.

Mitigation for Background issue (issue #2)

Restrict sensors access when an app is not in use like in Android 11. Google did not release a patch for Android 10 (Note as of Jan 2022 (two years after the Android 11 release), the majority of Android users (64.63%) still use Android 10 and below [2].

[2] Stats, S.G.: Mobile & tablet android version market share worldwide — statcounter global stats. https://gs.statcounter.com/android-version-market-share/mobile-tablet/worldwide/#monthly-202006-202009, (Accessed on 08/20/2021)

Conclusion

Confused deputy issues in mobile applications do break the integrity of Android OS system level services