Static Methods for Large-scale Android Supply Chain Analysis

Eduardo Blázquez González

Advisor: Prof. Juan Tapiador

September 25, 2024

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

Key Takeaways

Connected areas of research, each with unique motivations and conclusions:

- ❑ Studied a complex and fragmented ecosystem in the Android updates with first- and third-party actors. We uncover insecure update technology and harmful privacy practices. Used in some extreme cases for malware distribution.
- ❑ Discovered that installation capabilities are not only in first-party code, but in SDKs. In some cases, with capabilities to silently install applications.
- ❑ Current SOTA tools for static analysis of Android apps do not fit the performance demand imposed by the datasets sizes, needing tools that perform faster and with less memory consumption.
- ❑ A large dataset analysis revealed low adoption of software protection solutions, despite an increasing trend. Sensitive data applications and external market apps use these protections more frequently.

Key Takeaways

Connected areas of research, each with unique motivations and conclusions:

- ❑ Studied a complex and fragmented ecosystem in the Android updates with first- and third-party actors. We uncover insecure update technology and harmful privacy practices. Used in some extreme cases for malware distribution.
- ❑ Discovered that installation capabilities are not only in first-party code, but in SDKs. In some cases, with capabilities to silently install applications.
- ❑ Current SOTA tools for static analysis of Android apps do not fit the performance demand imposed by the datasets sizes, needing tools that perform faster and with less memory consumption.
- ❑ A large dataset analysis revealed low adoption of software protection solutions, despite an increasing trend. Sensitive data applications and external market apps use these protections more frequently.

Key Takeaways

Connected areas of research, each with unique motivations and conclusions:

- ❑ Studied a complex and fragmented ecosystem in the Android updates with first- and third-party actors. We uncover insecure update technology and harmful privacy practices. Used in some extreme cases for malware distribution.
- ❑ Discovered that installation capabilities are not only in first-party code, but in SDKs. In some cases, with capabilities to silently install applications.
- ❑ Current SOTA tools for static analysis of Android apps do not fit the performance demand imposed by the datasets sizes, needing tools that perform faster and with less memory consumption.
- ❑ A large dataset analysis revealed low adoption of software protection solutions, despite an increasing trend. Sensitive data applications and external market apps use these protections more frequently.

Key Takeaways

Connected areas of research, each with unique motivations and conclusions:

- ❑ Studied a complex and fragmented ecosystem in the Android updates with first- and third-party actors. We uncover insecure update technology and harmful privacy practices. Used in some extreme cases for malware distribution.
- ❑ Discovered that installation capabilities are not only in first-party code, but in SDKs. In some cases, with capabilities to silently install applications.
- ❑ Current SOTA tools for static analysis of Android apps do not fit the performance demand imposed by the datasets sizes, needing tools that perform faster and with less memory consumption.
- ❑ A large dataset analysis revealed low adoption of software protection solutions, despite an increasing trend. Sensitive data applications and external market apps use these protections more frequently.

Key Takeaways

Connected areas of research, each with unique motivations and conclusions:

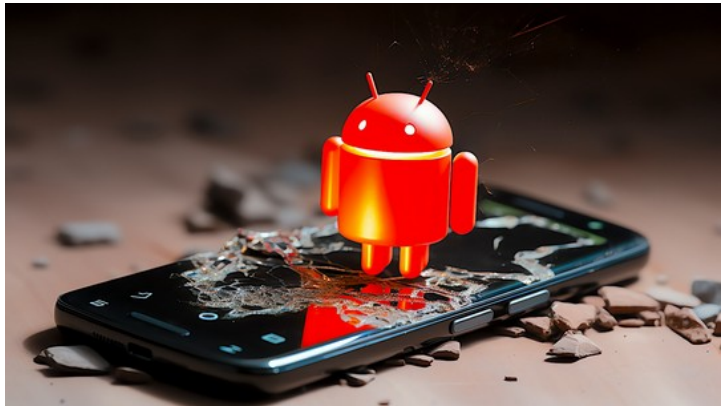
- ❑ Studied a complex and fragmented ecosystem in the Android updates with first- and third-party actors. We uncover insecure update technology and harmful privacy practices. Used in some extreme cases for malware distribution.
- ❑ Discovered that installation capabilities are not only in first-party code, but in SDKs. In some cases, with capabilities to silently install applications.
- ❑ Current SOTA tools for static analysis of Android apps do not fit the performance demand imposed by the datasets sizes, needing tools that perform faster and with less memory consumption.
- ❑ A large dataset analysis revealed low adoption of software protection solutions, despite an increasing trend. Sensitive data applications and external market apps use these protections more frequently.

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

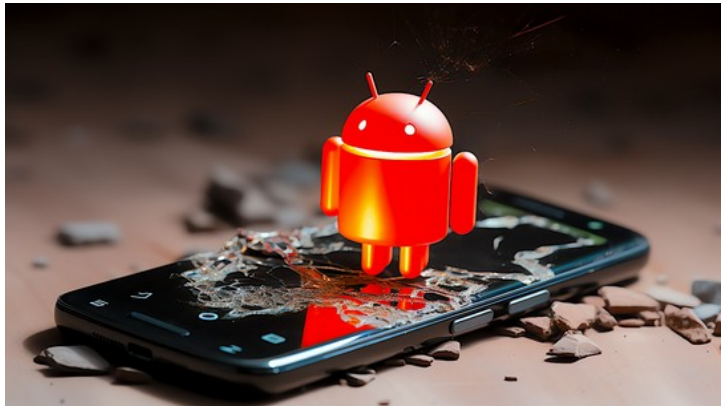
Motivation

Motivation

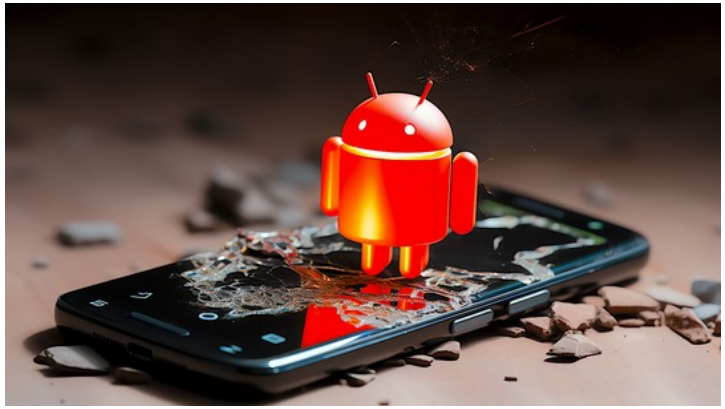
Motivation



Motivation

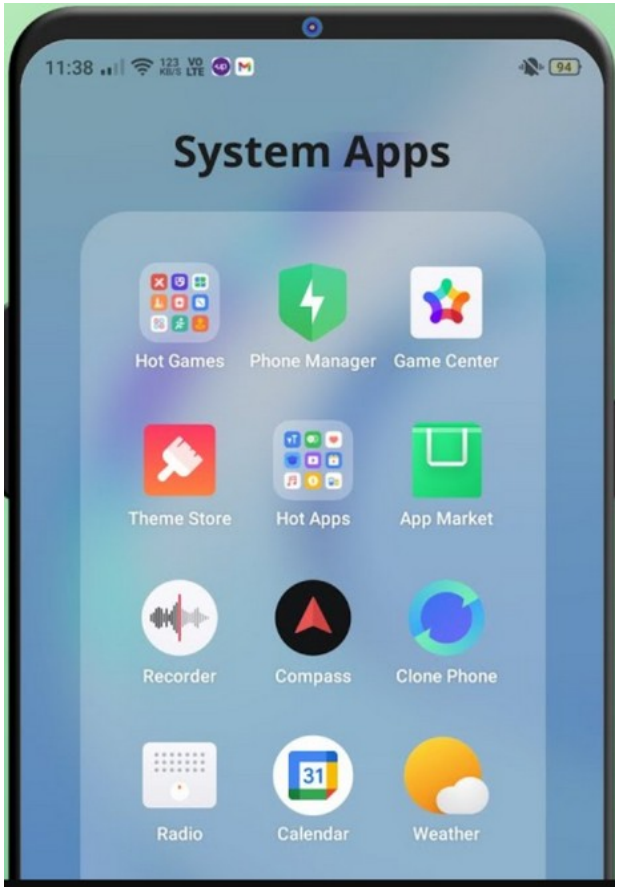


Motivation

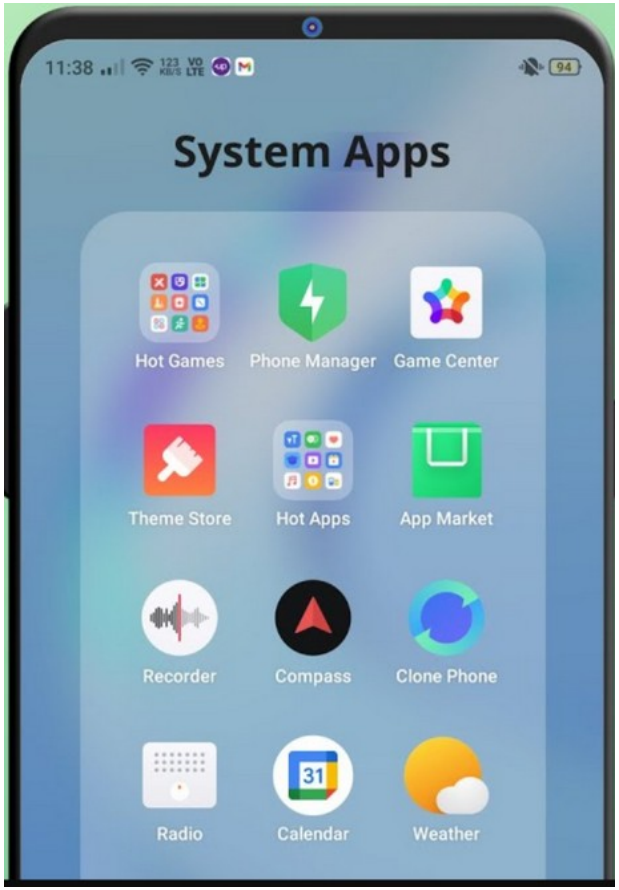


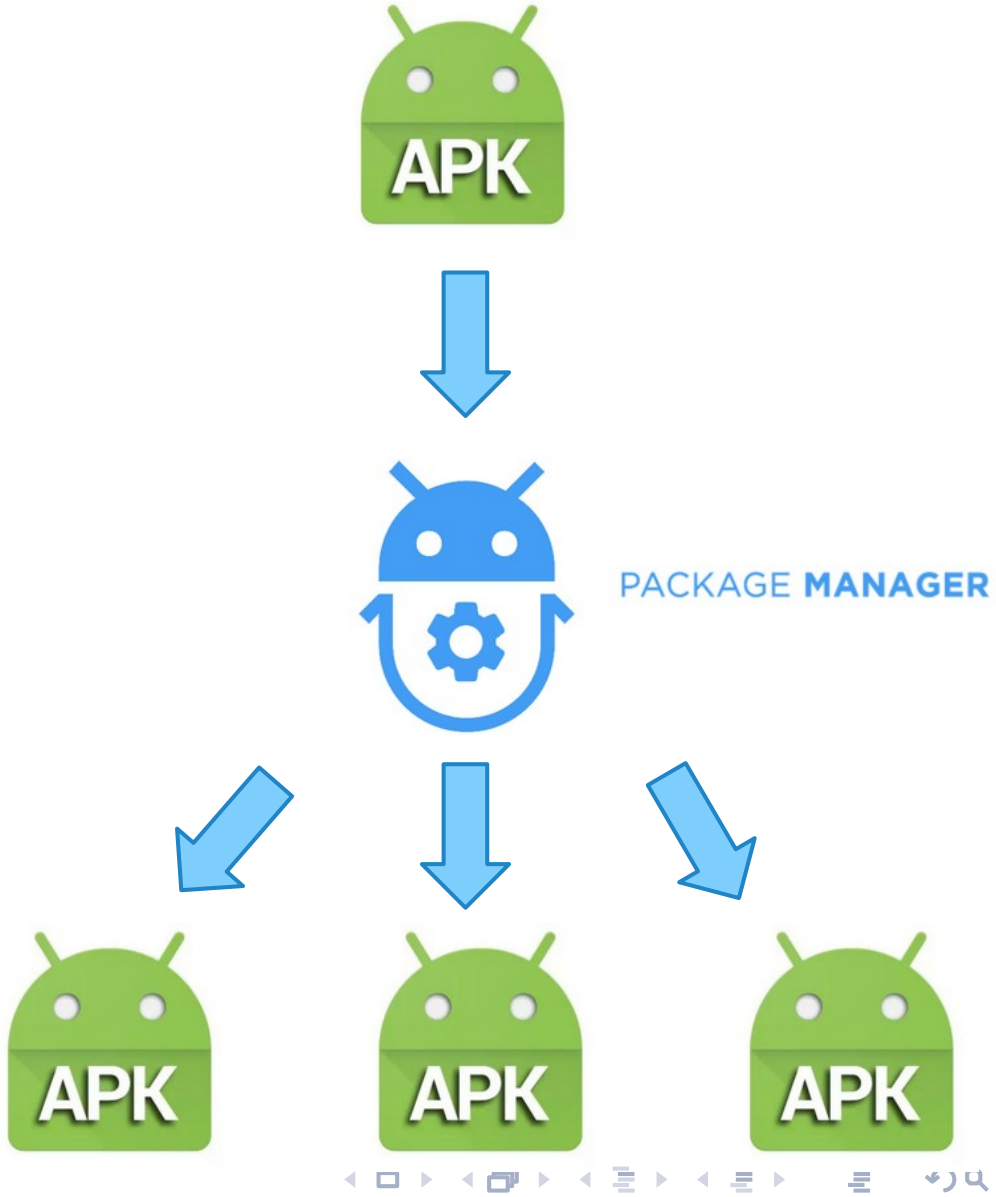
Motivation

Motivation


11:38
94
System Apps
Hot Games
Phone Manager
Game Center
Theme Store
Hot Apps
App Market
Recorder
Compass
Clone Phone
Radio
31
Calendar
Weather

Motivation


11:38
System Apps
Hot Games
Phone Manager
Game Center
Theme Store
Hot Apps
App Market
Recorder
Compass
Clone Phone
Radio
Calendar
Weather


APK
PACKAGE MANAGER
APK
APK
APK

Motivation

- Execution time?
- Memory consumption?
- Performance?

Motivation

Motivation

Motivation

- Who does obfuscate?
- Which Software is Used?

Motivation

- Who does obfuscate?
- Which Software is Used?

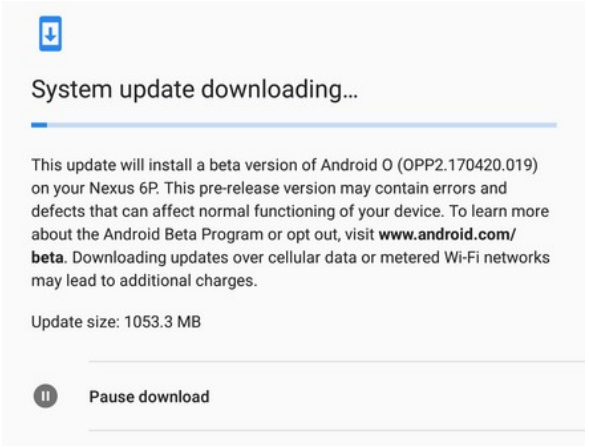
- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

FOTA (Firmware-Over-The-Air)

- Software used to update Android devices

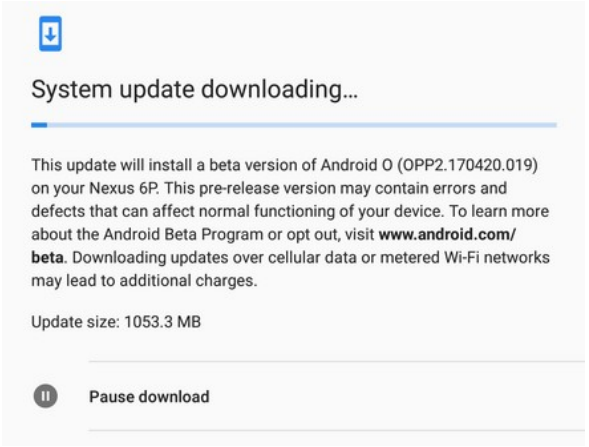
FOTA (Firmware-Over-The-Air)

- Software used to update Android devices



FOTA (Firmware-Over-The-Air)

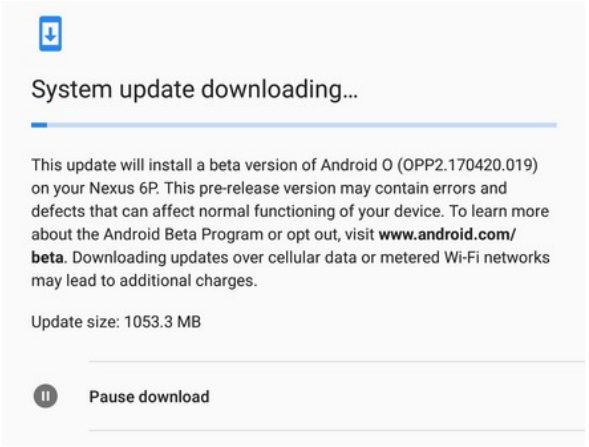
- Software used to update Android devices



- Critical Pre-installed App

FOTA (Firmware-Over-The-Air)

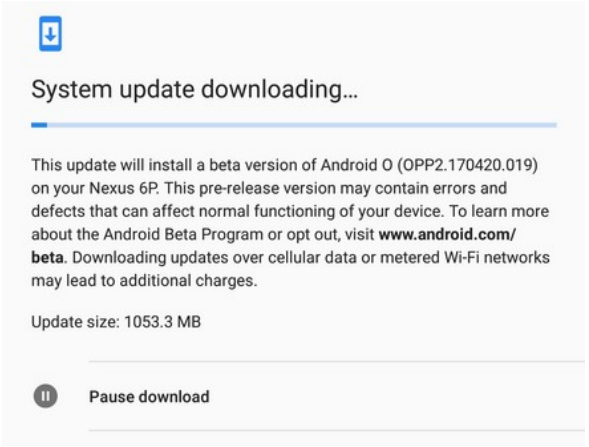
- Software used to update Android devices



- Critical Pre-installed App
- Turns a static supply-chain into a dynamic one

FOTA (Firmware-Over-The-Air)

- Software used to update Android devices



- Critical Pre-installed App
- Turns a static supply-chain into a dynamic one
- No large-scale analysis was done before

Problems Reported in Media

Problems Reported in Media

- Sometimes used with intrusive purposes

Problems Reported in Media

- Sometimes used with intrusive purposes

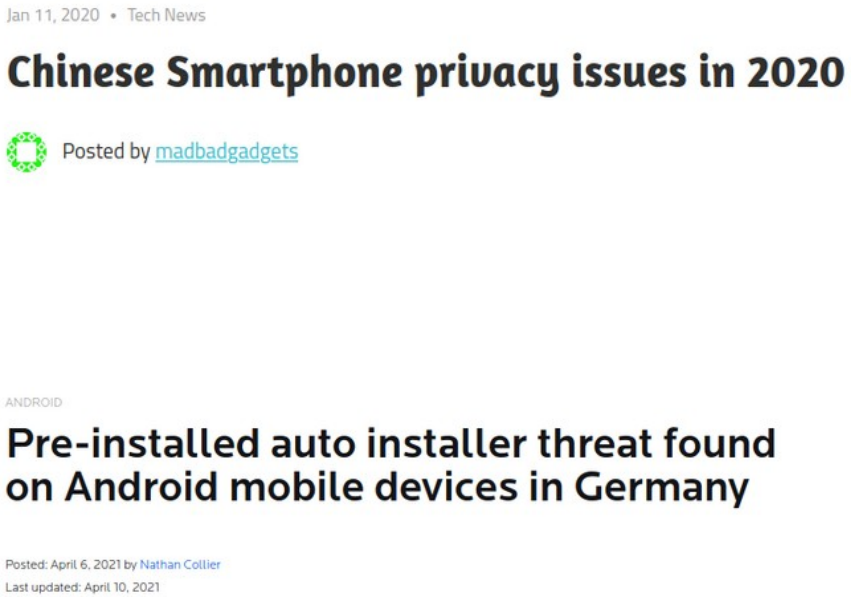

Jan 11, 2020 • Tech News

Chinese Smartphone privacy issues in 2020

Posted by madbadgadgets

Problems Reported in Media

- Sometimes used with intrusive purposes



Jan 11, 2020 • Tech News

Chinese Smartphone privacy issues in 2020

Posted by madbadgadgets

ANDROID

Pre-installed auto installer threat found on Android mobile devices in Germany

Posted: April 6, 2021 by Nathan Collier
Last updated: April 10, 2021

Problems Reported in Media

- Sometimes used with intrusive purposes

Jan 11, 2020 • Tech News

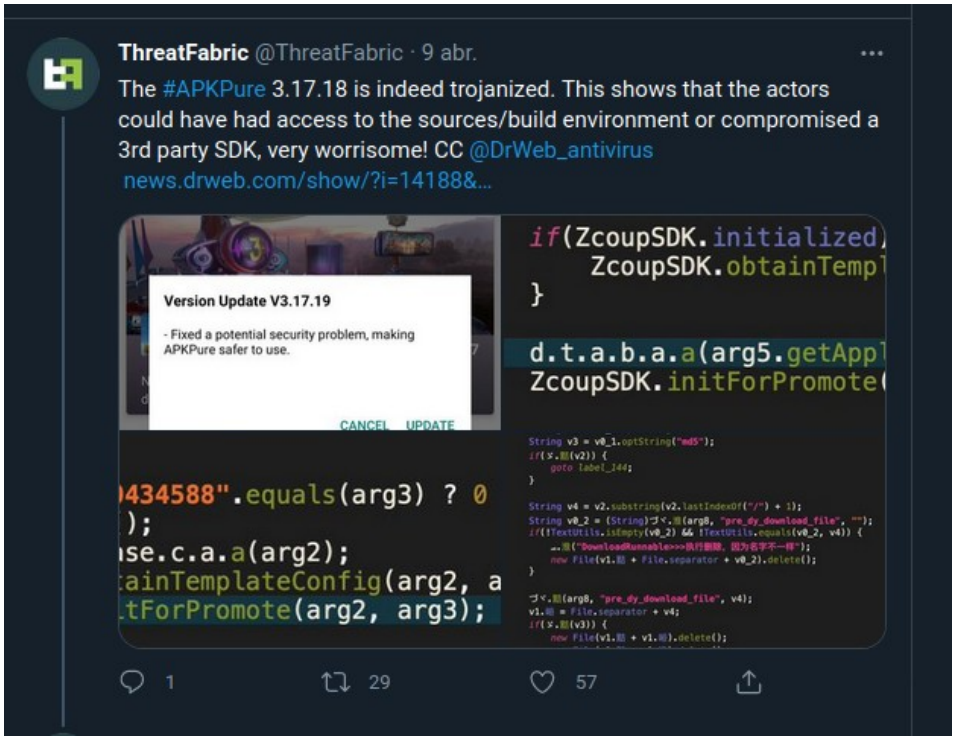
Chinese Smartphone privacy issues in 2020

Posted by madbadgadgets

ANDROID

Pre-installed auto installer threat found on Android mobile devices in Germany

Posted: April 6, 2021 by Nathan Collier
Last updated: April 10, 2021



Research Questions

- How to **detect** a **FOTA** app? (Detection)
- Who is behind these apps? (Attribution)
- What **capabilities** do these apps **have**?
- And... What **behavior** do they **present**?

(Behavioral Analysis)

Datasets

Firmware Scanner[1]

- +400K pre-installed apps
- Device information

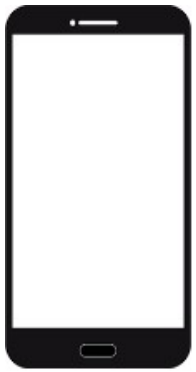
Reputation and Installation Logs

NortonLifeLock™

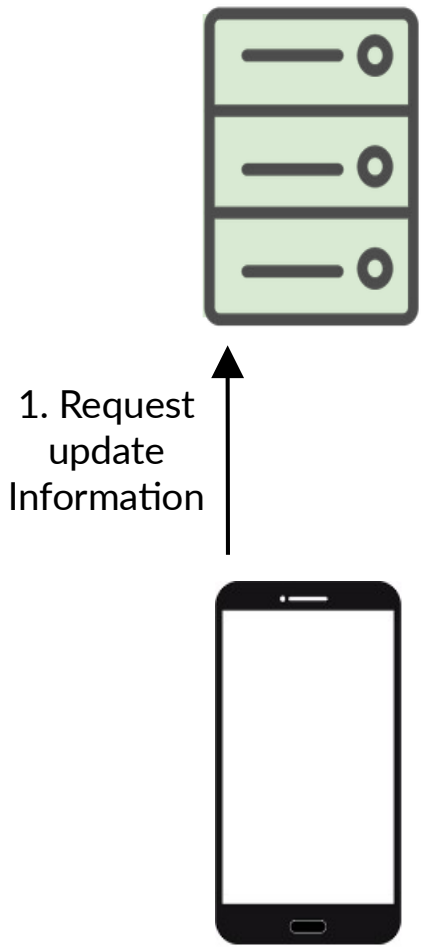
- Reputation logs
- Installed packages information

[1]Gamba, Julien, Rashed, Mohammed, Razaghpanah, Abbas, Tapiador, Juan, and Vallina-Rodriguez, Narseo, "An Analysis of Pre-installed Android Software," in *2020 IEEE Symposium on Security and Privacy (SP)*, 2020.

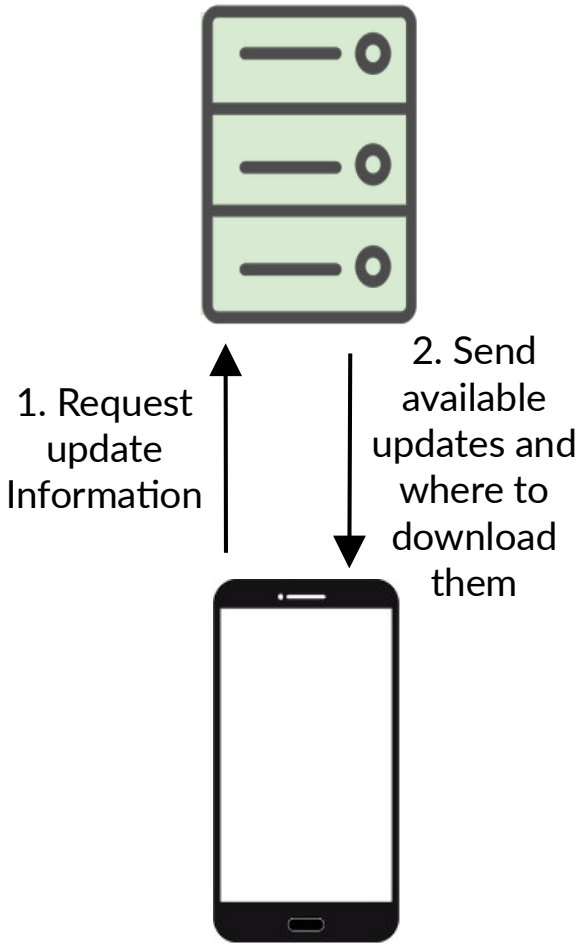
FOTA Lifecycle



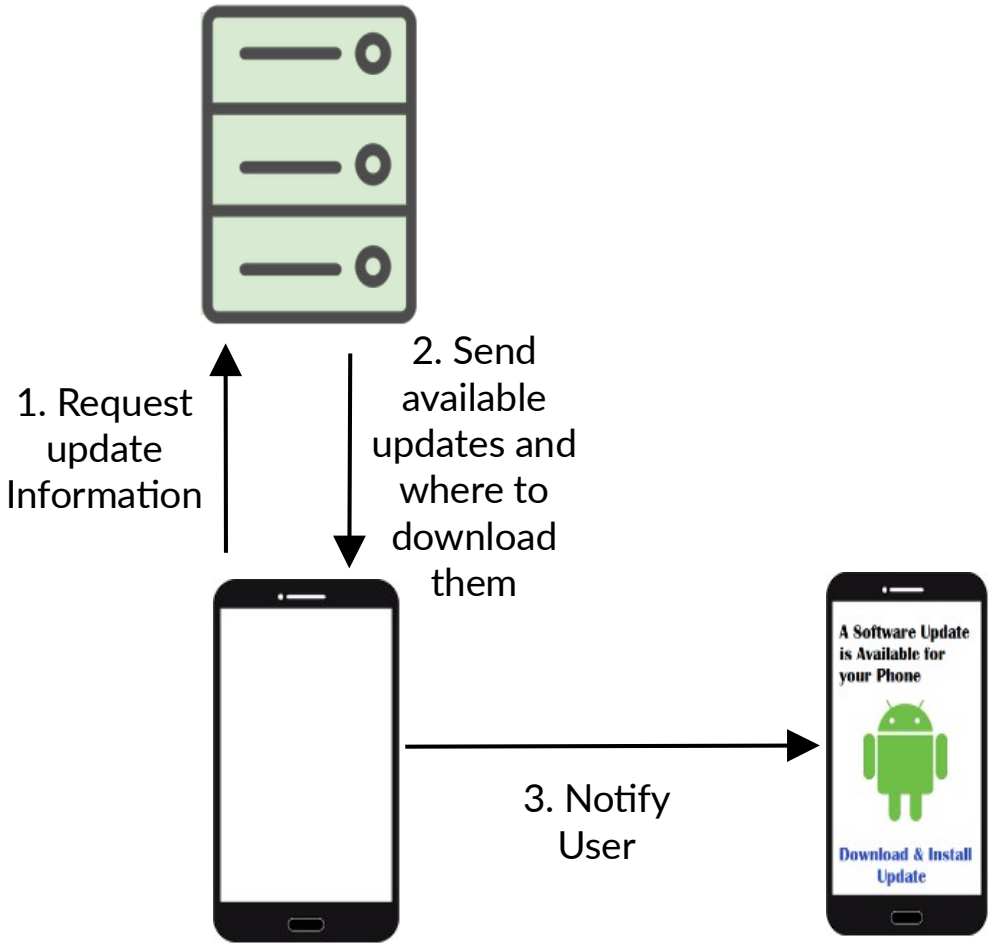
FOTA Lifecycle



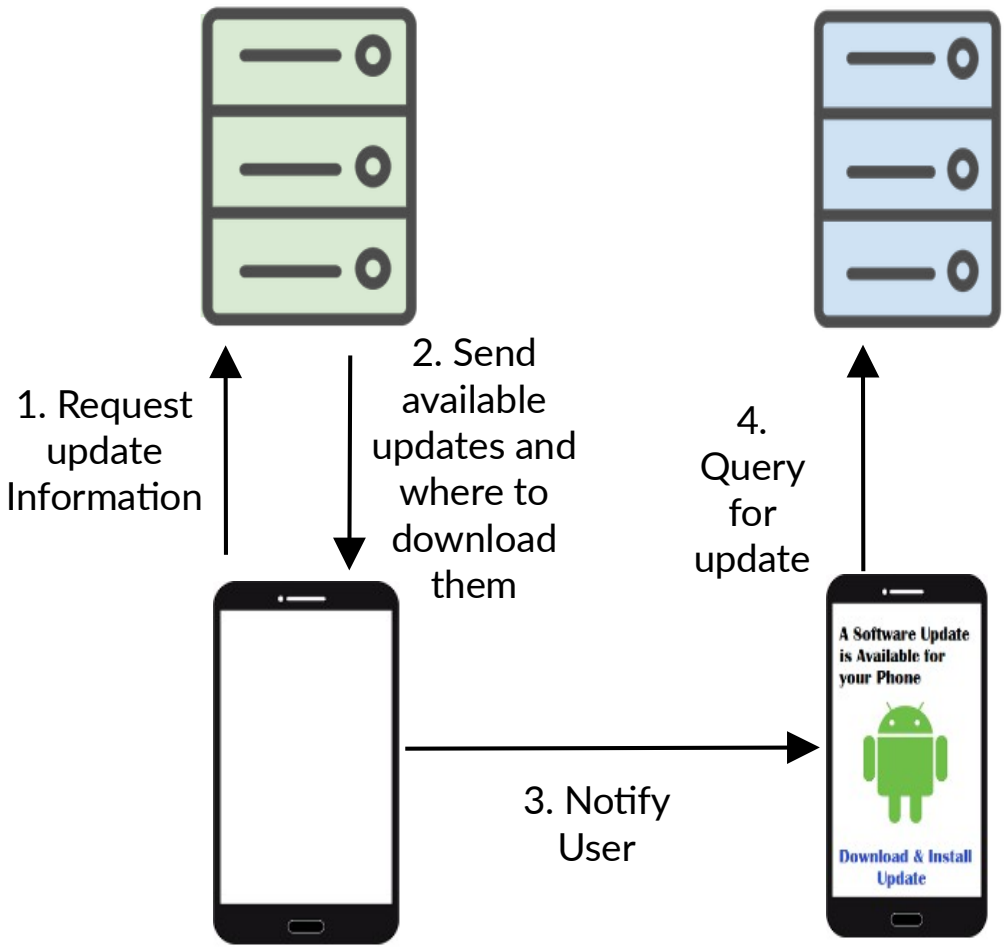
FOTA Lifecycle



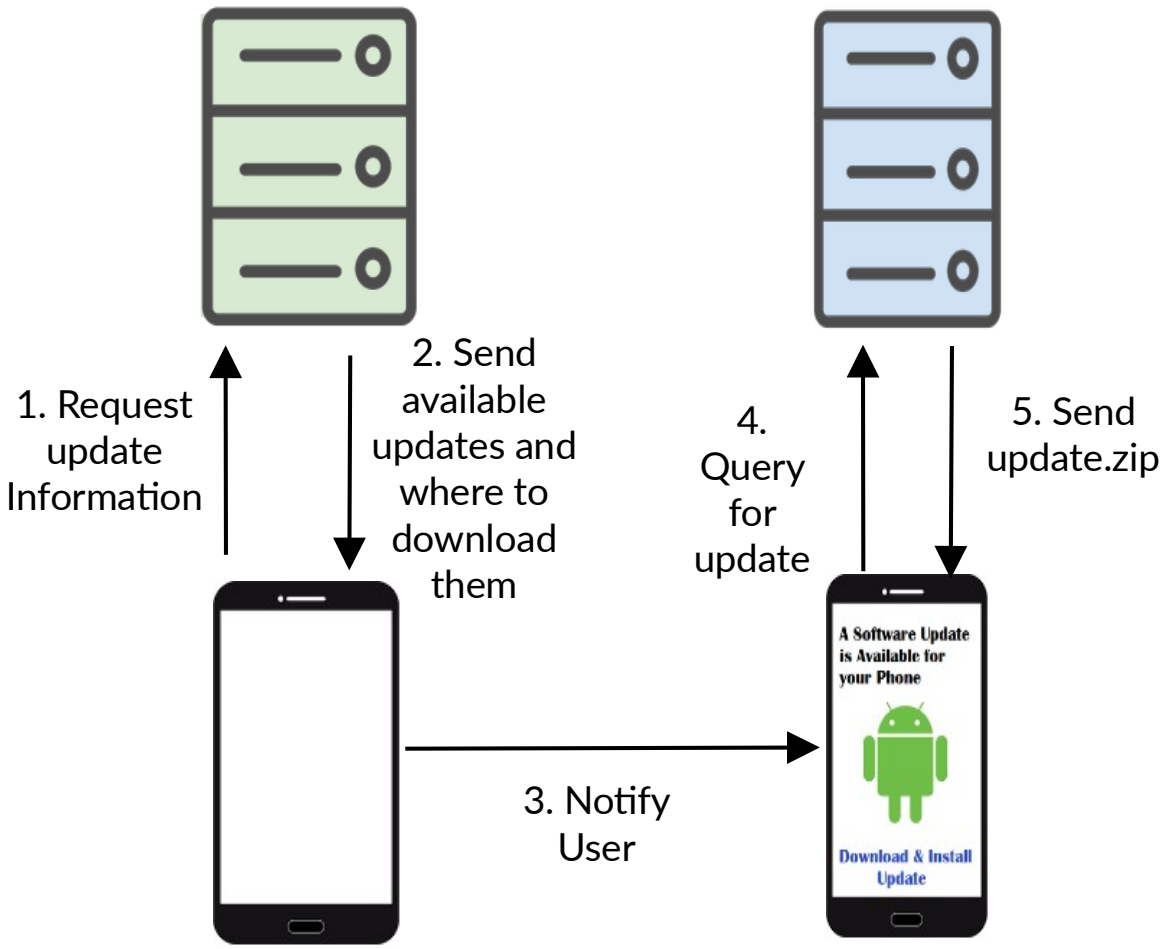
FOTA Lifecycle



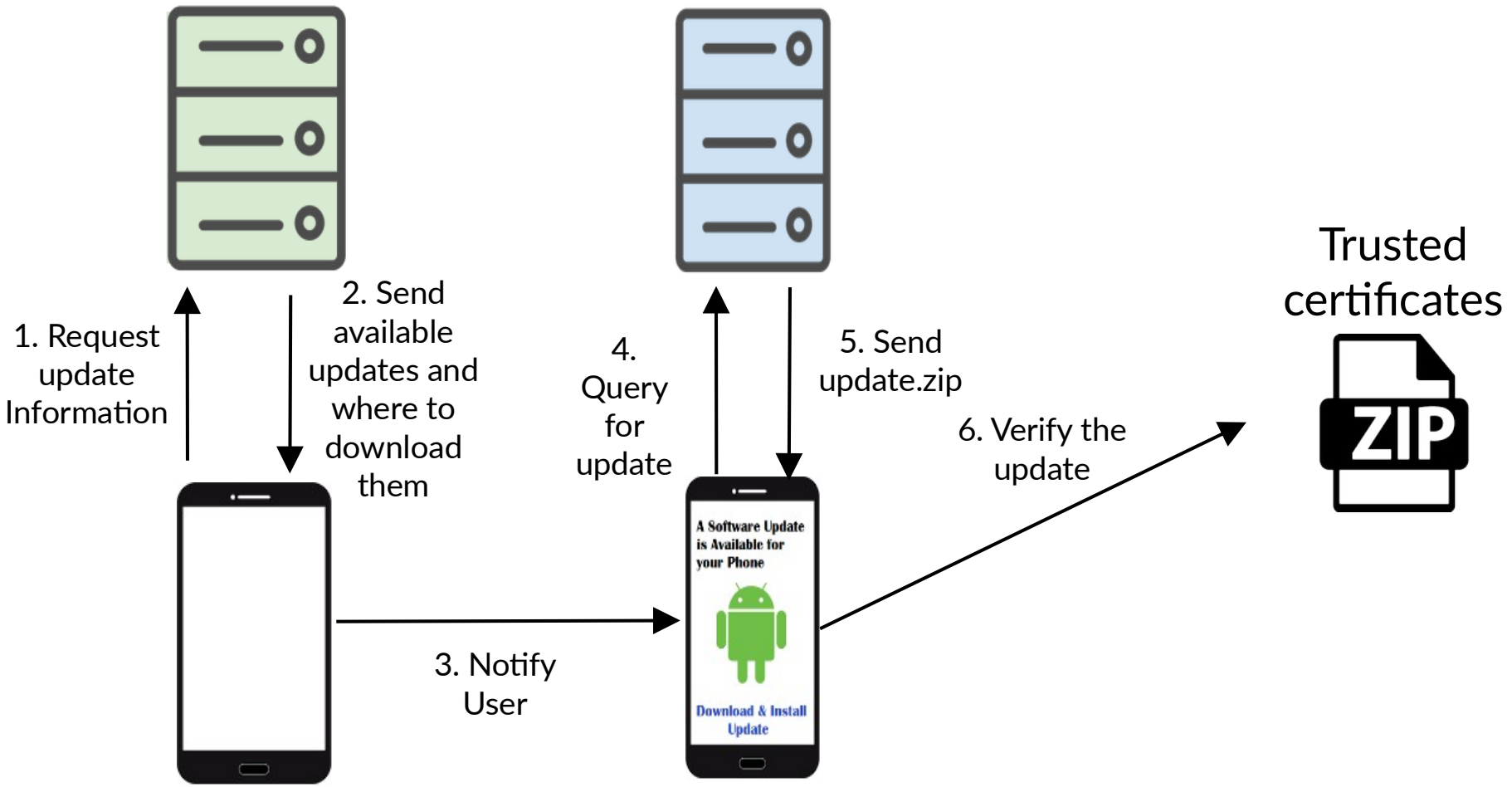
FOTA Lifecycle



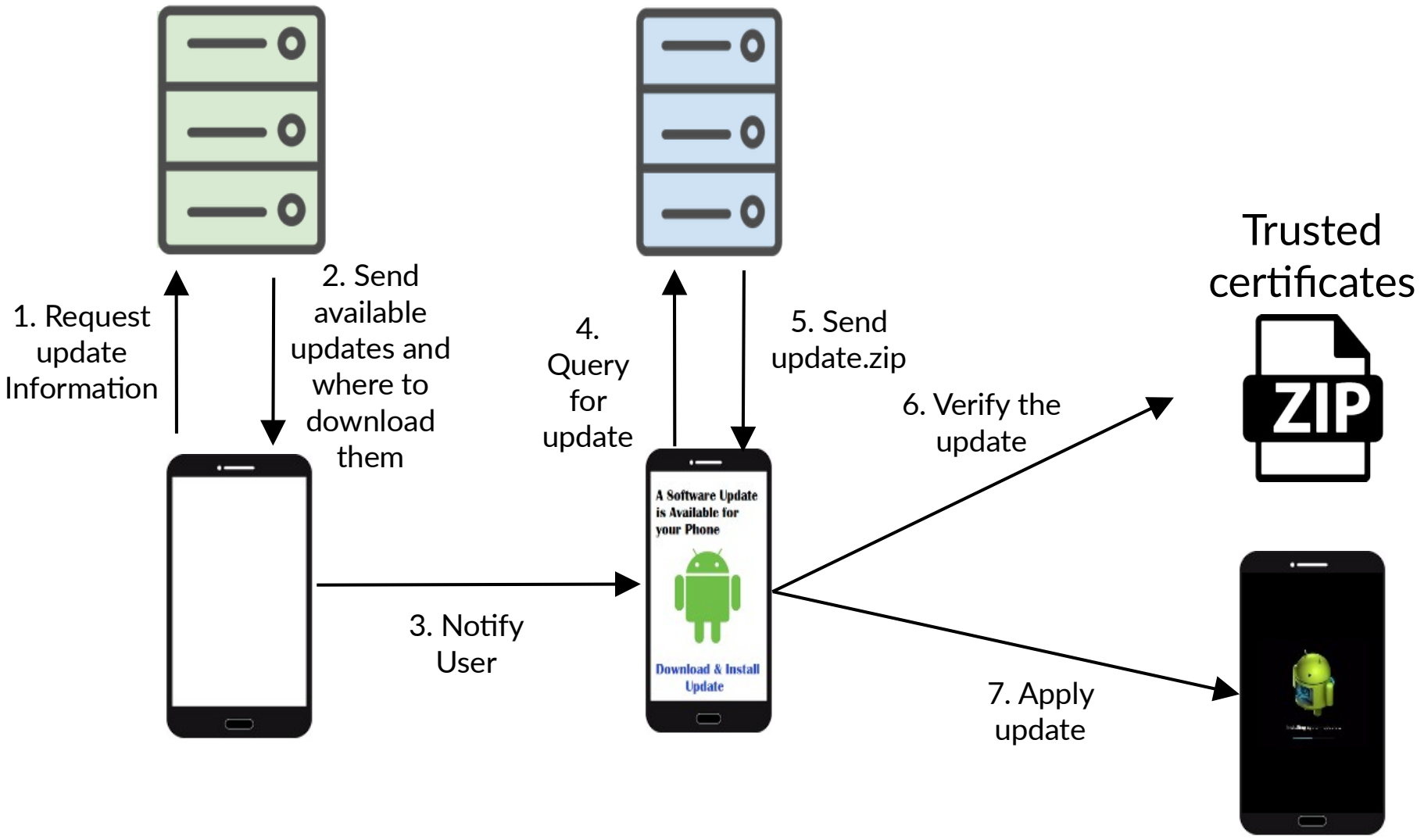
FOTA Lifecycle



FOTA Lifecycle



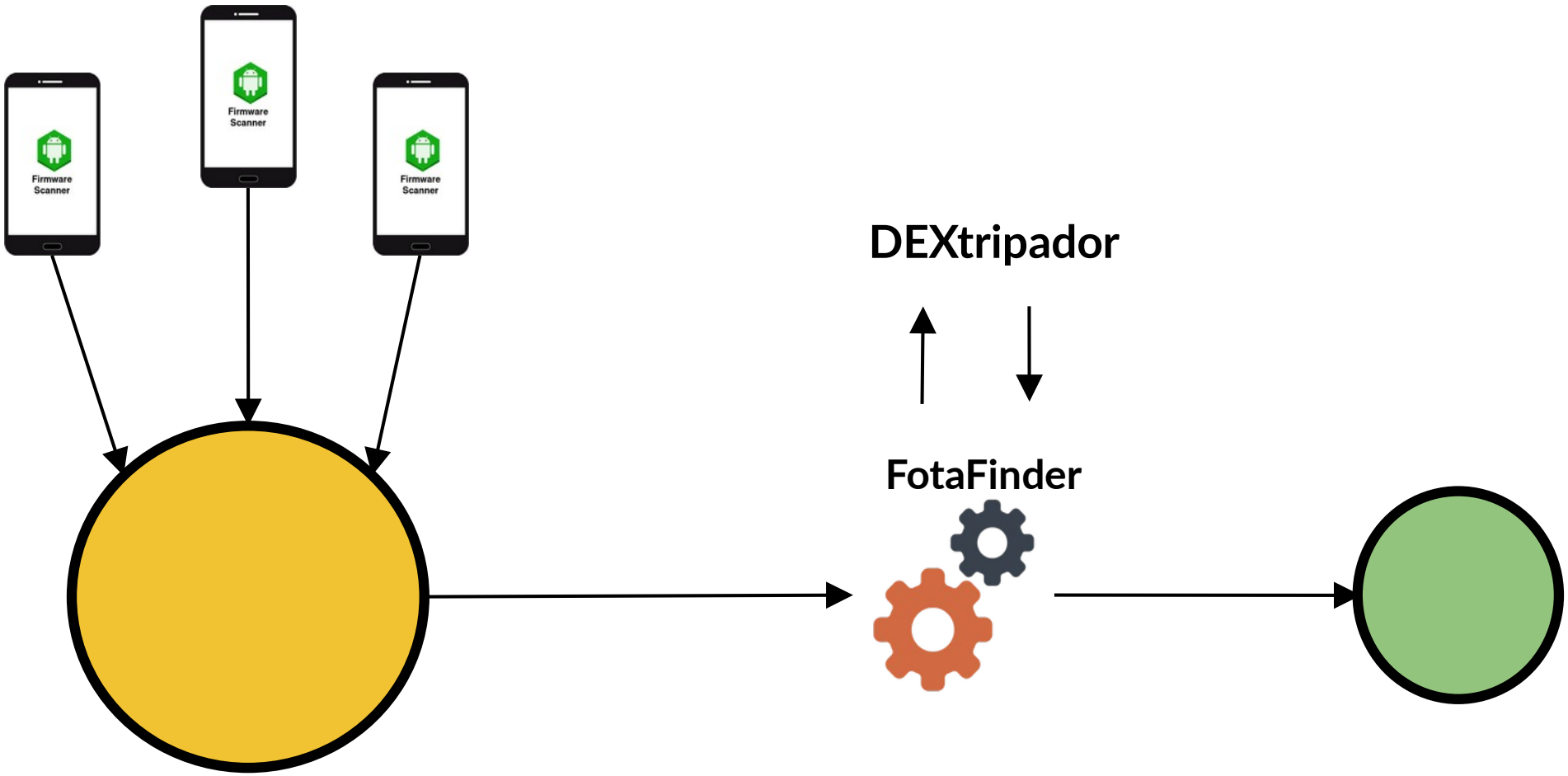
FOTA Lifecycle



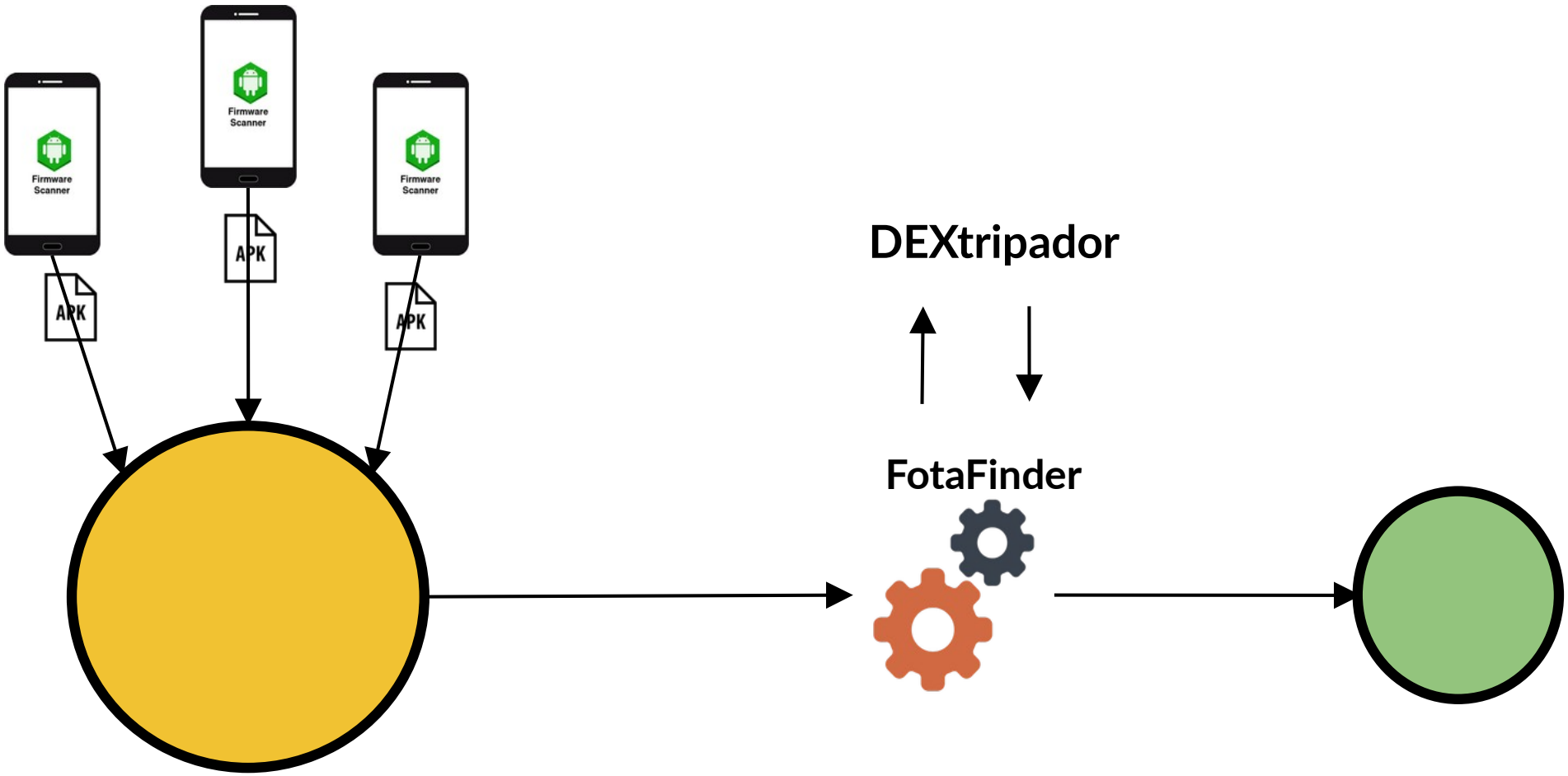
FOTA Finder

- Static analysis tool to automatically detect FOTA applications.
- Search for 4 specific signals related to FOTA:
  - `verifyPackage()`
  - `installPackage()`
  - `applyPayload()`
  - "*/cache/recovery/command*" and "*--update-pacakge*"

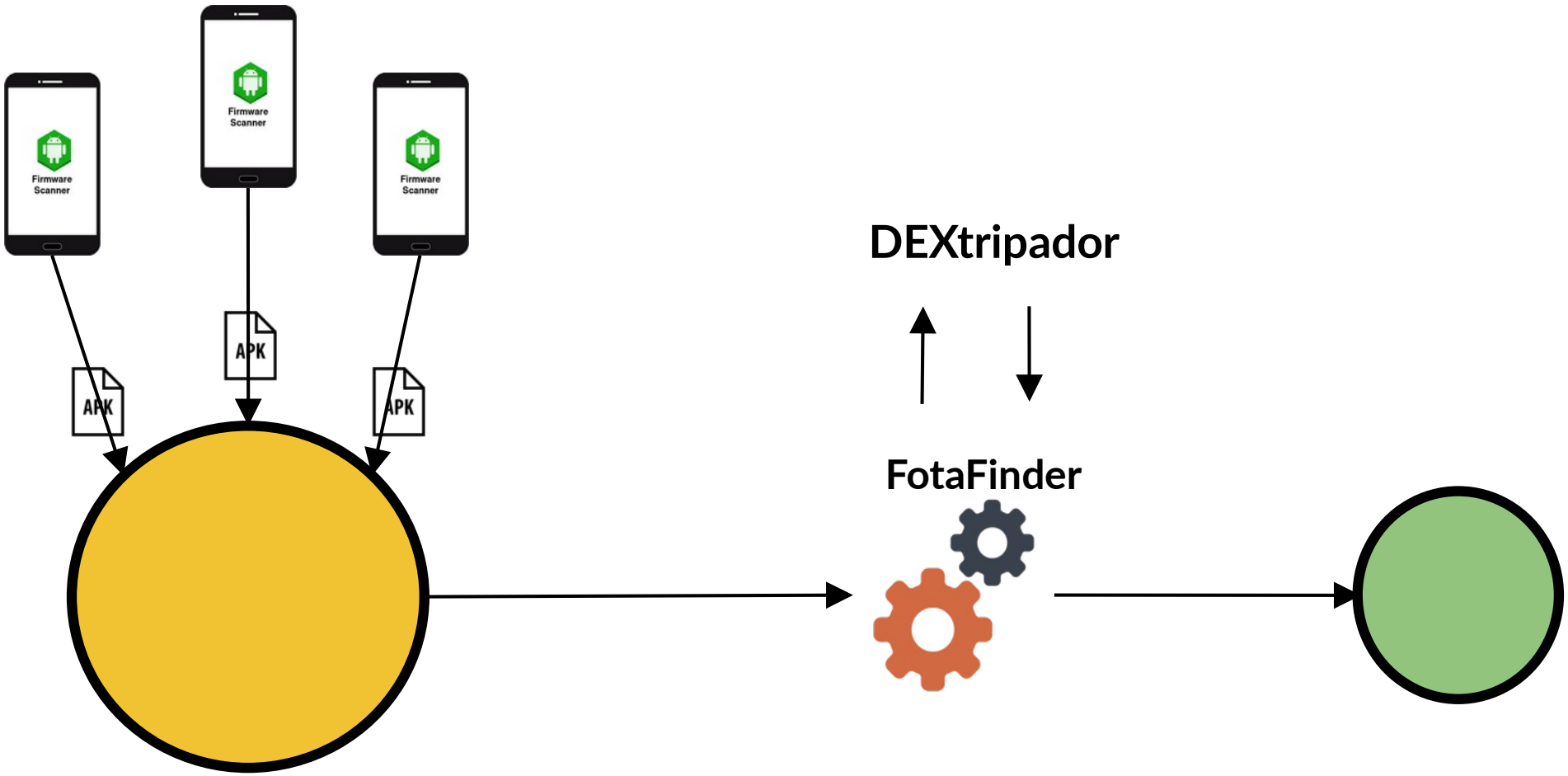
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
DEXtripador
FotaFinder

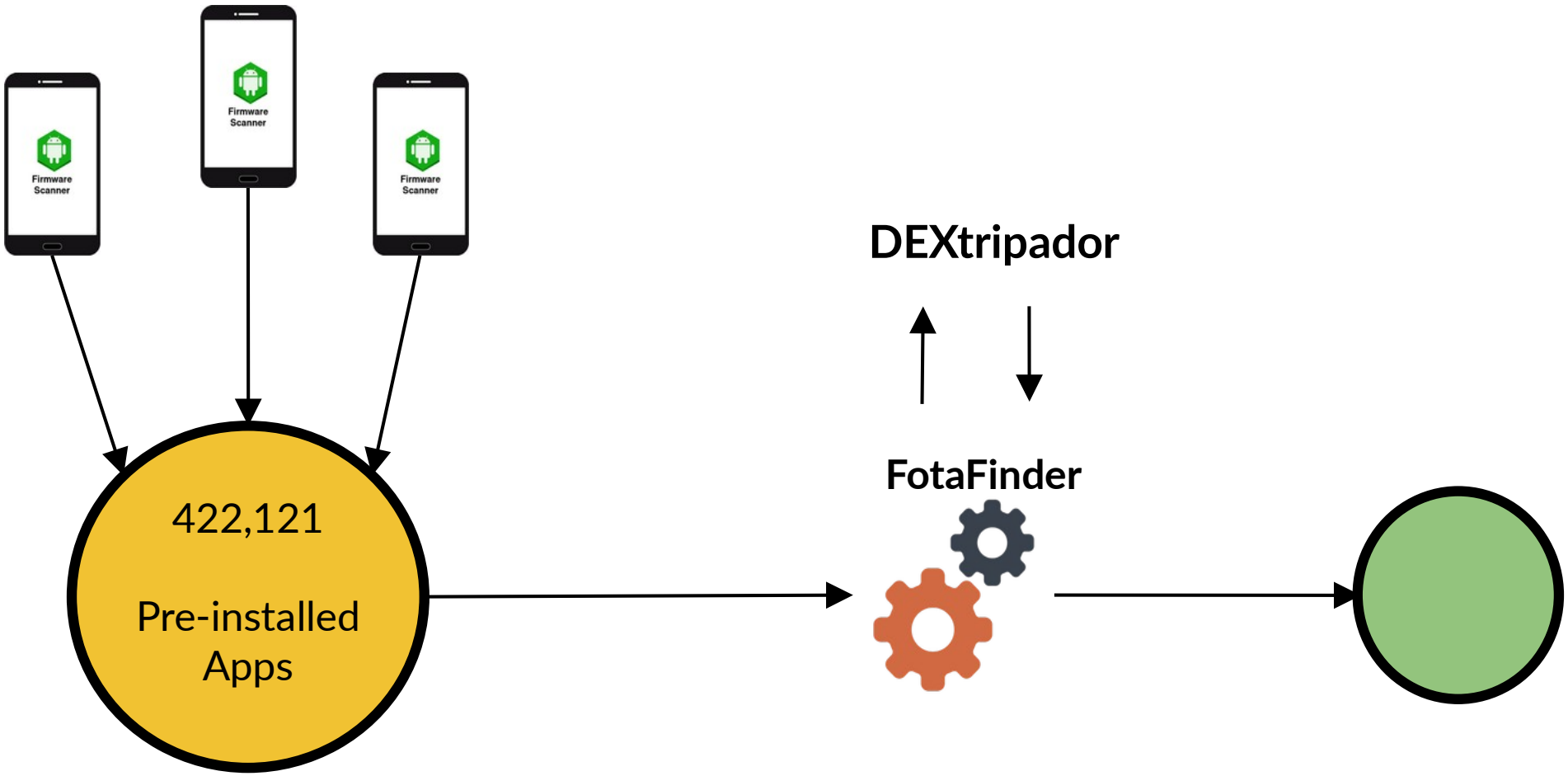
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
APK
APK
APK
DEXtripador
FotaFinder

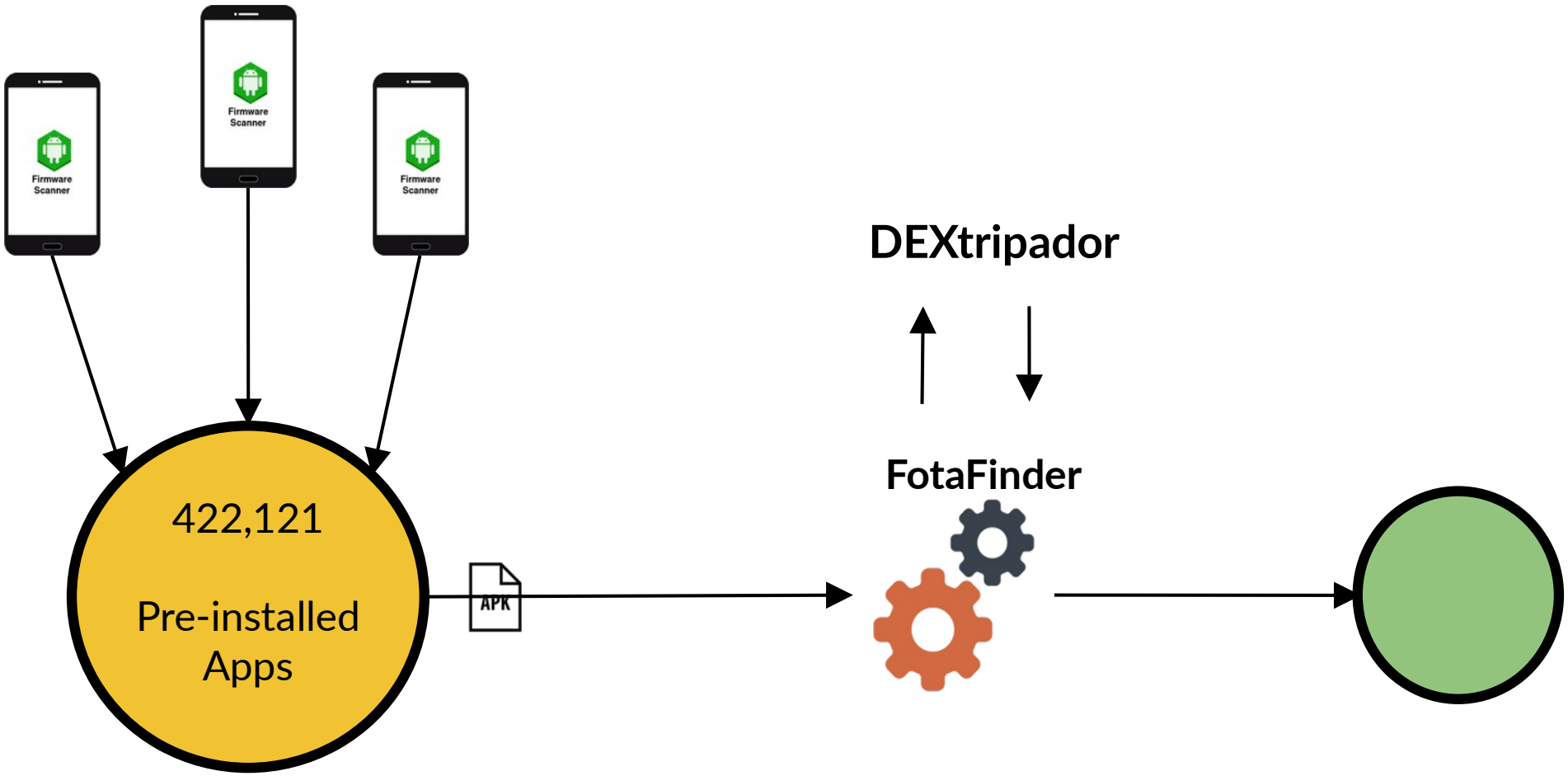
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
APK
APK
APK
DEXtripador
FotaFinder

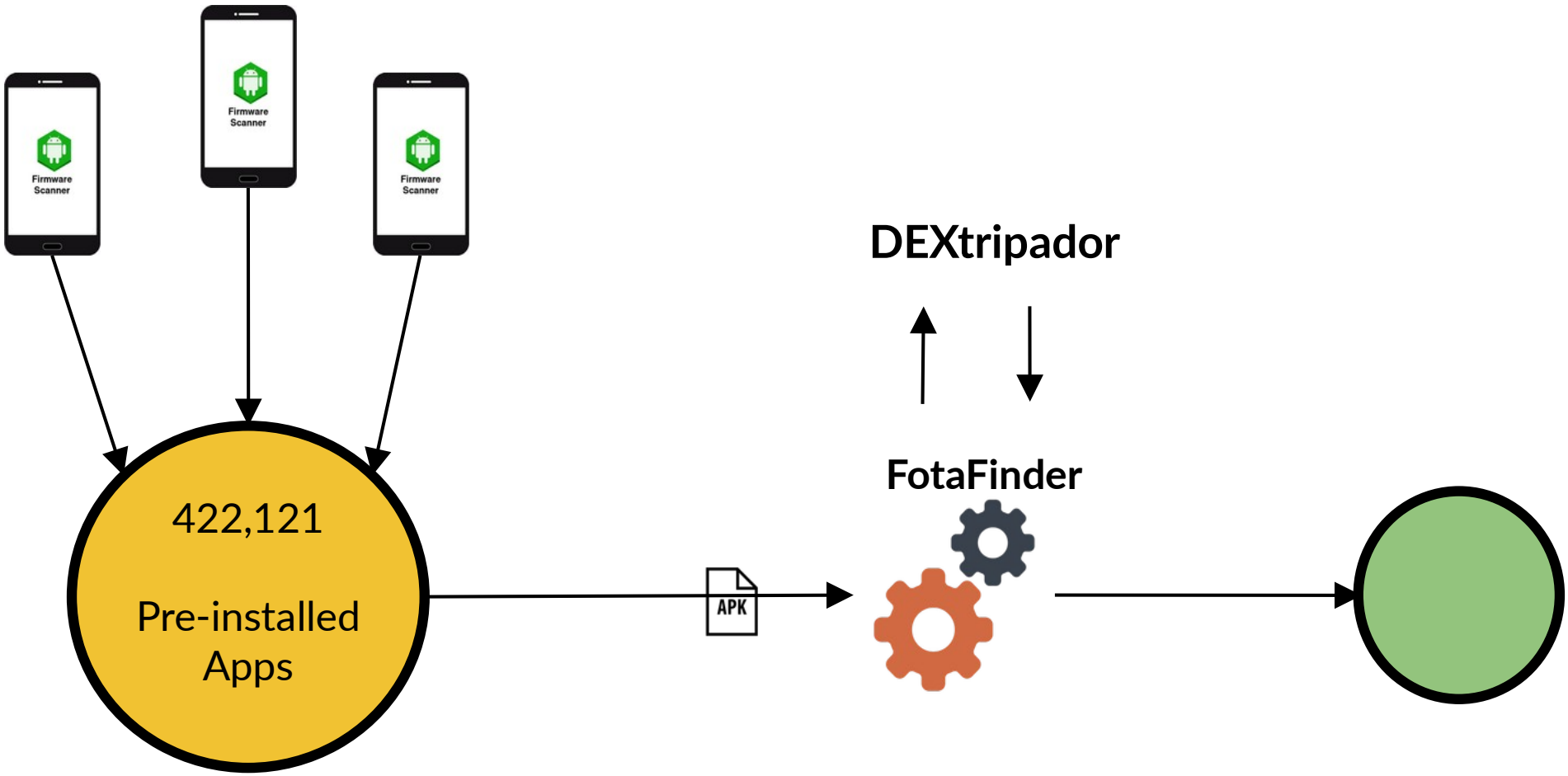
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
DEXtripador
FotaFinder

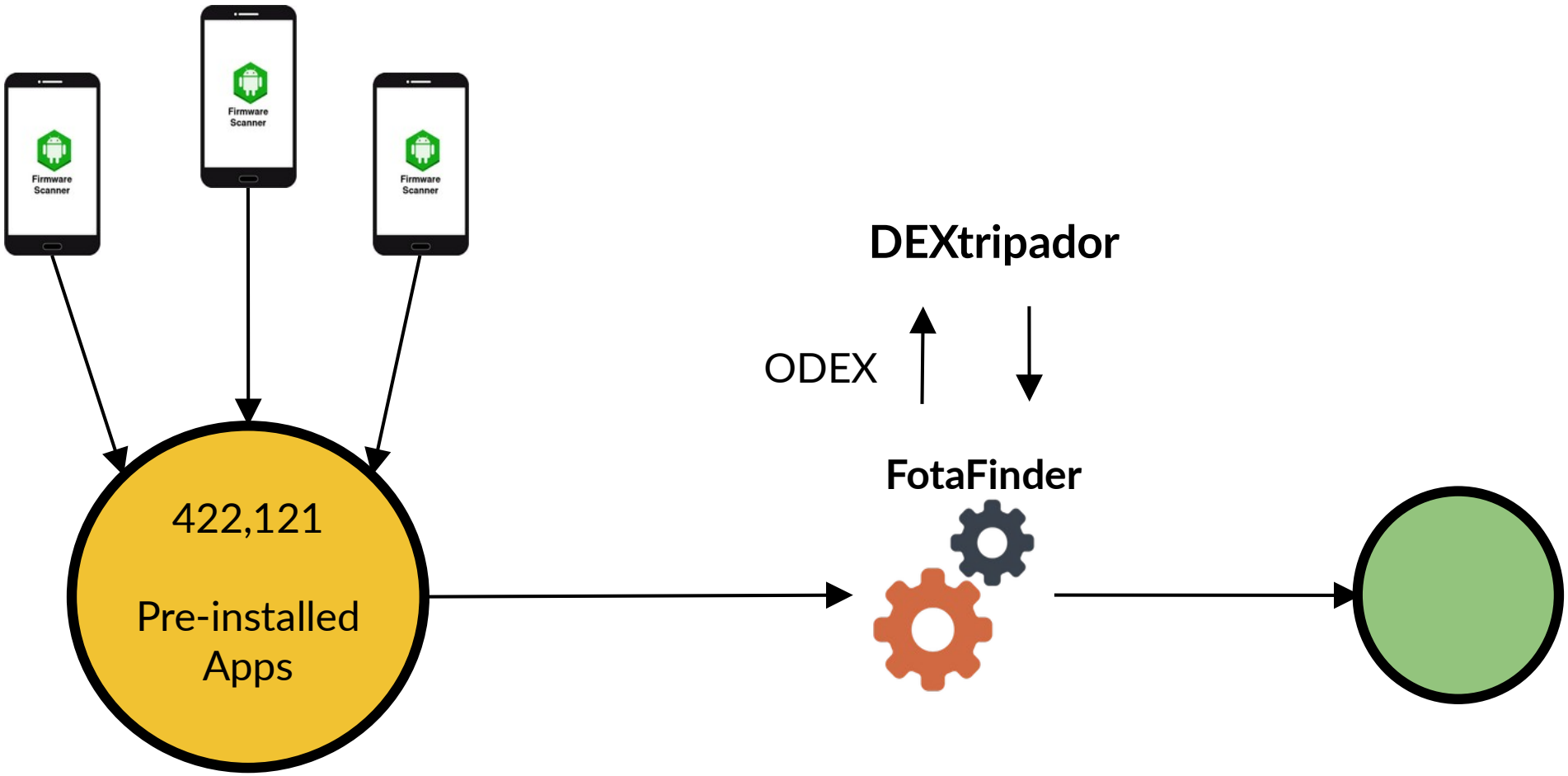
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
APK
DEXtripador
FotaFinder

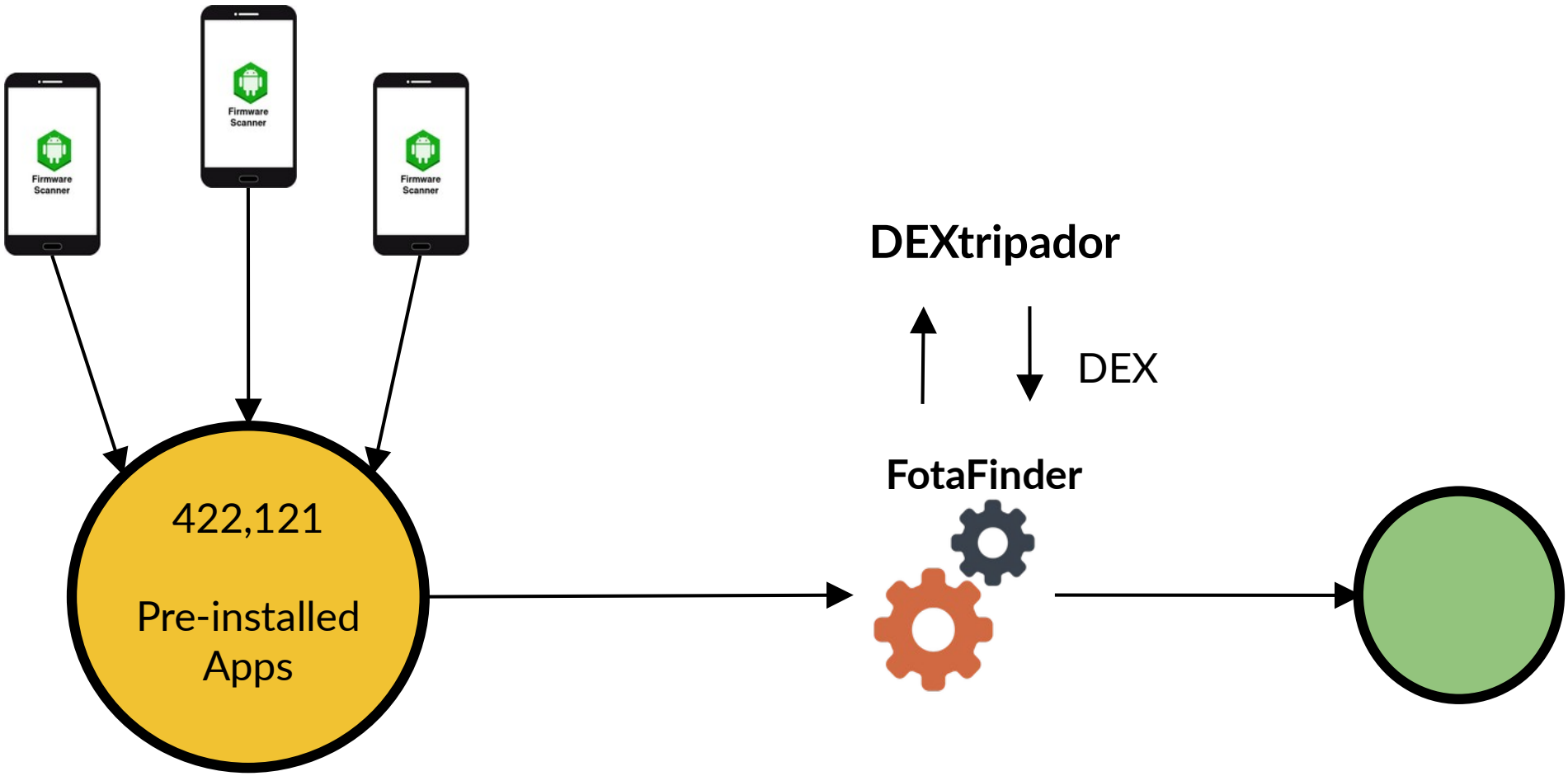
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
DEXtripador
FotaFinder
422,121
Pre-installed Apps
APK

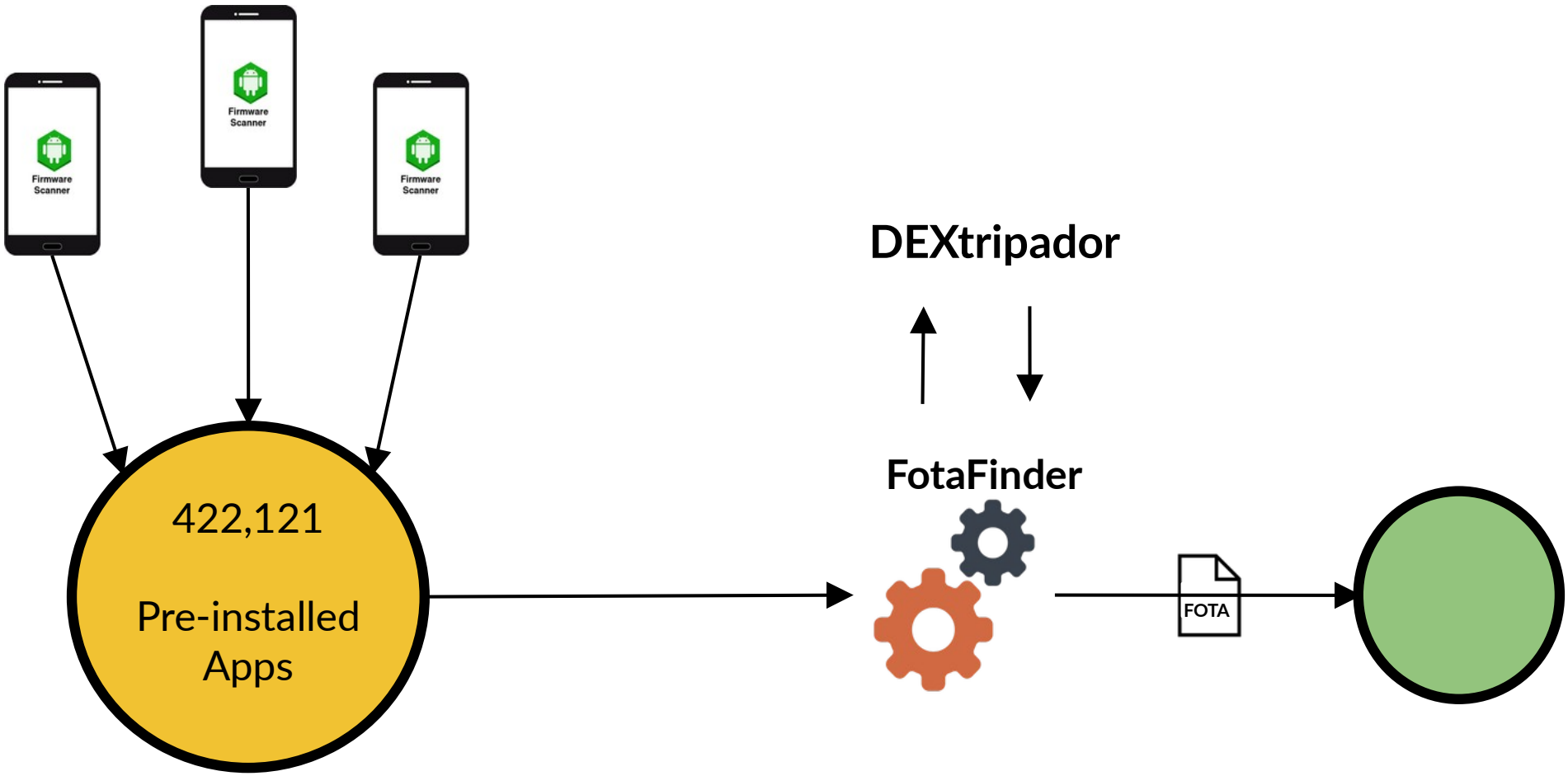
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
DEXtripador
ODEX
FotaFinder

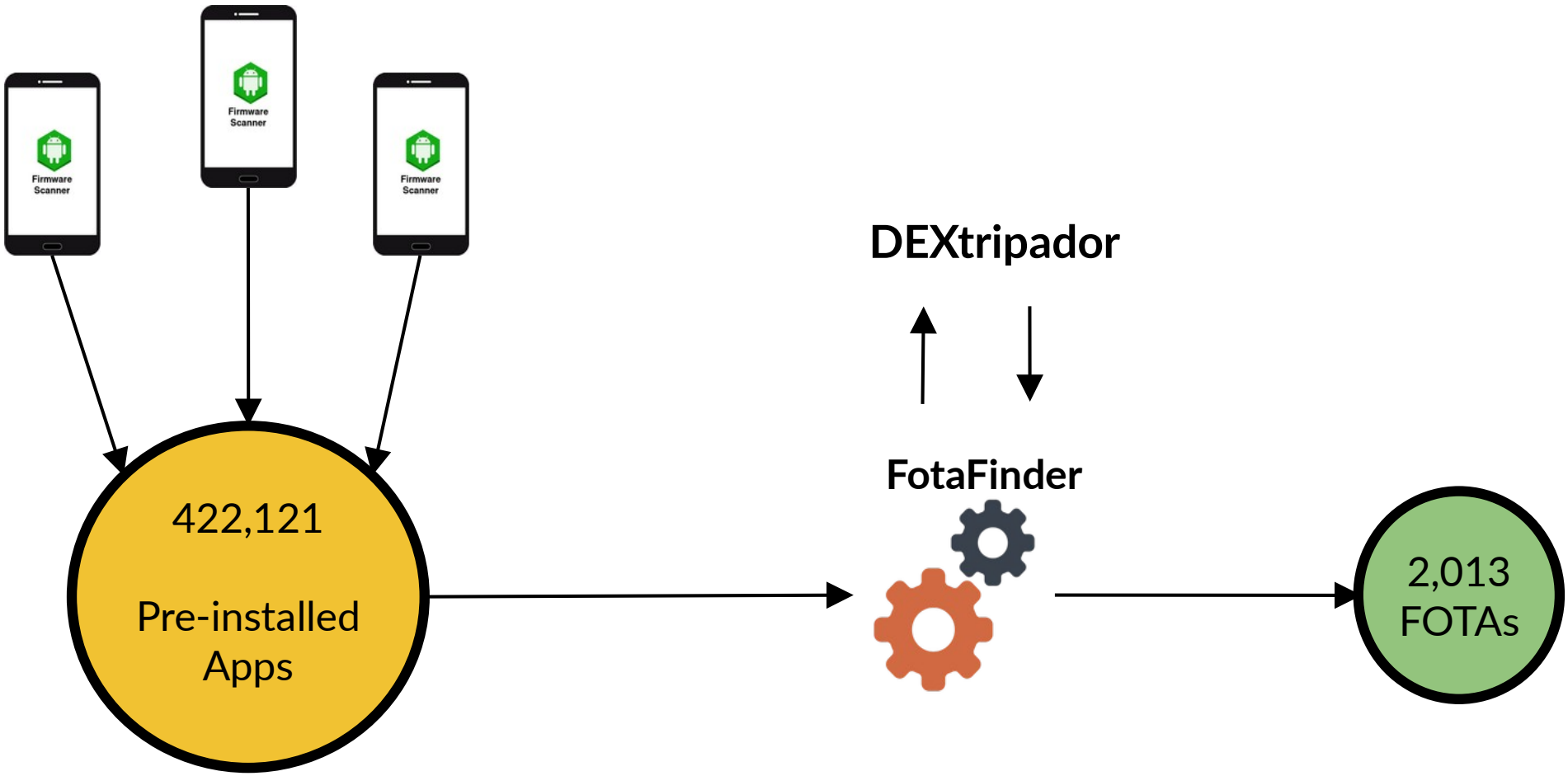
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
DEXtripador
DEX
FotaFinder

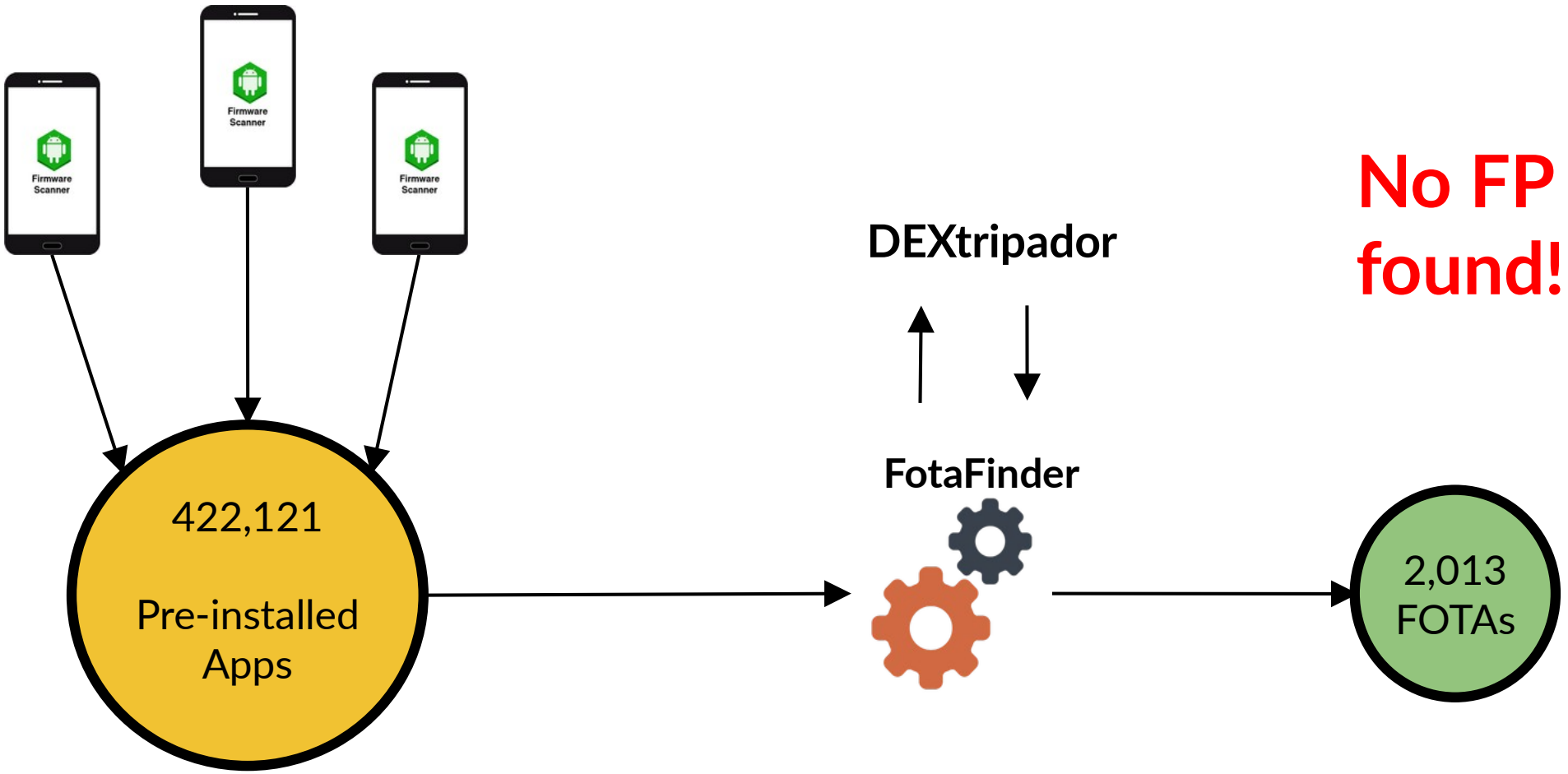
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
DEXtripador
FotaFinder
FOTA

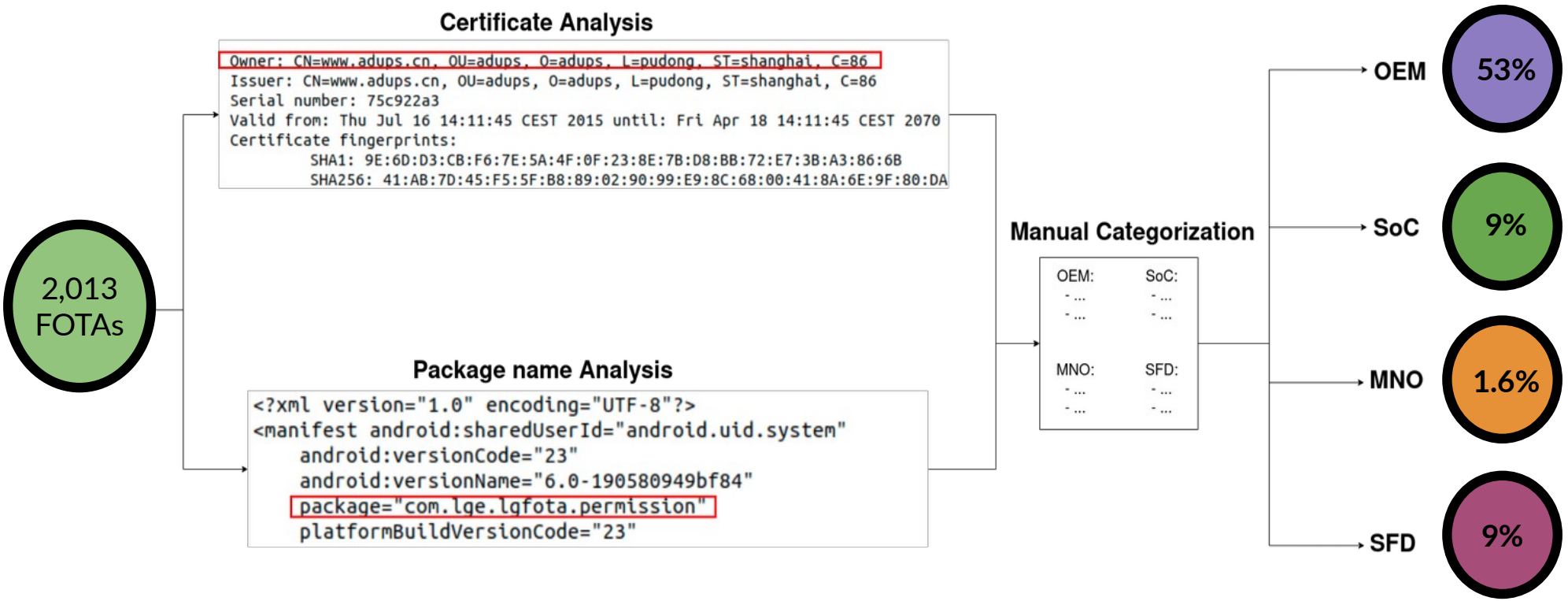
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
DEXtripador
FotaFinder
2,013 FOTAs

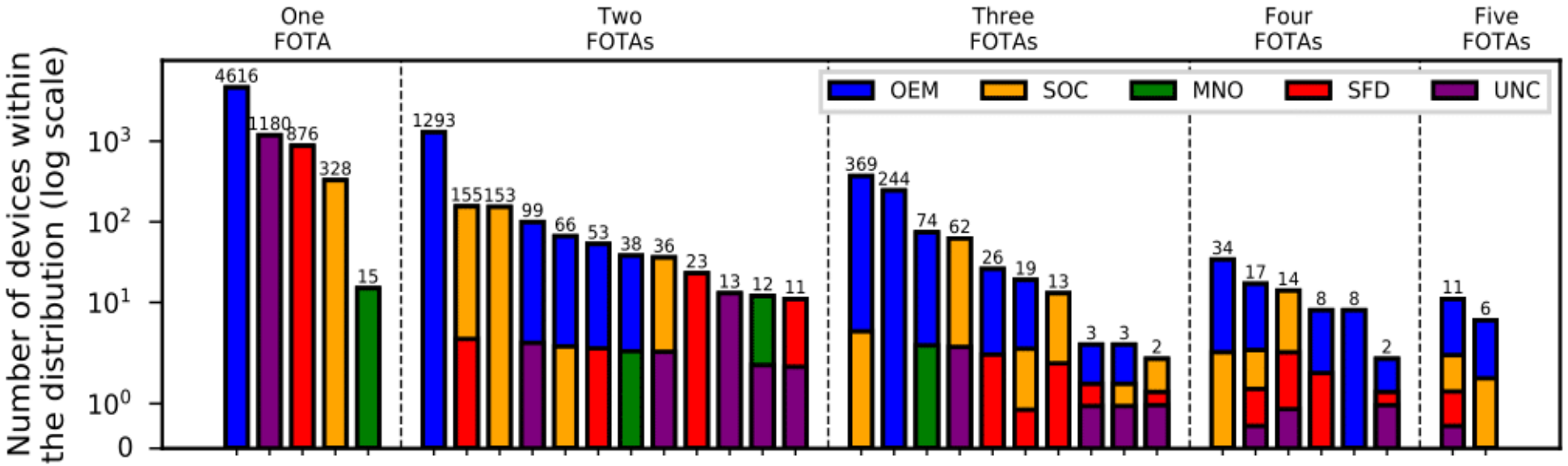
FOTA Finder


Firmware Scanner
Firmware Scanner
Firmware Scanner
422,121
Pre-installed Apps
DEXtripador
FotaFinder
2,013 FOTAs
No FP found!

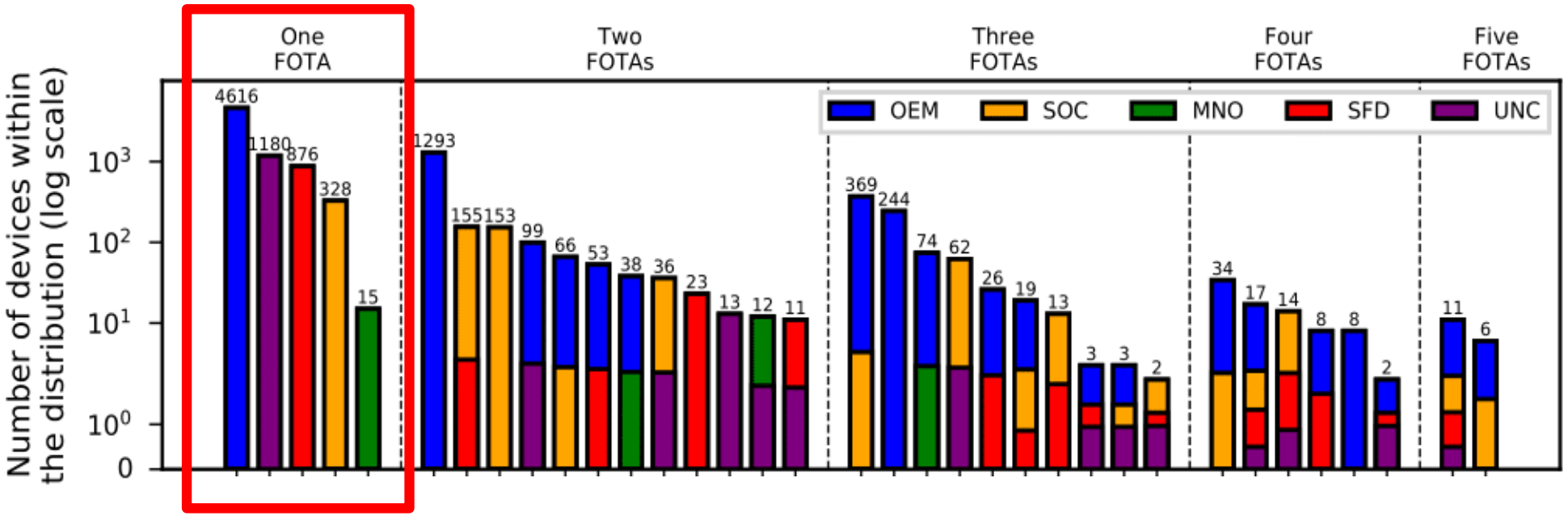
FOTA Stakeholder Analysis (Attribution)



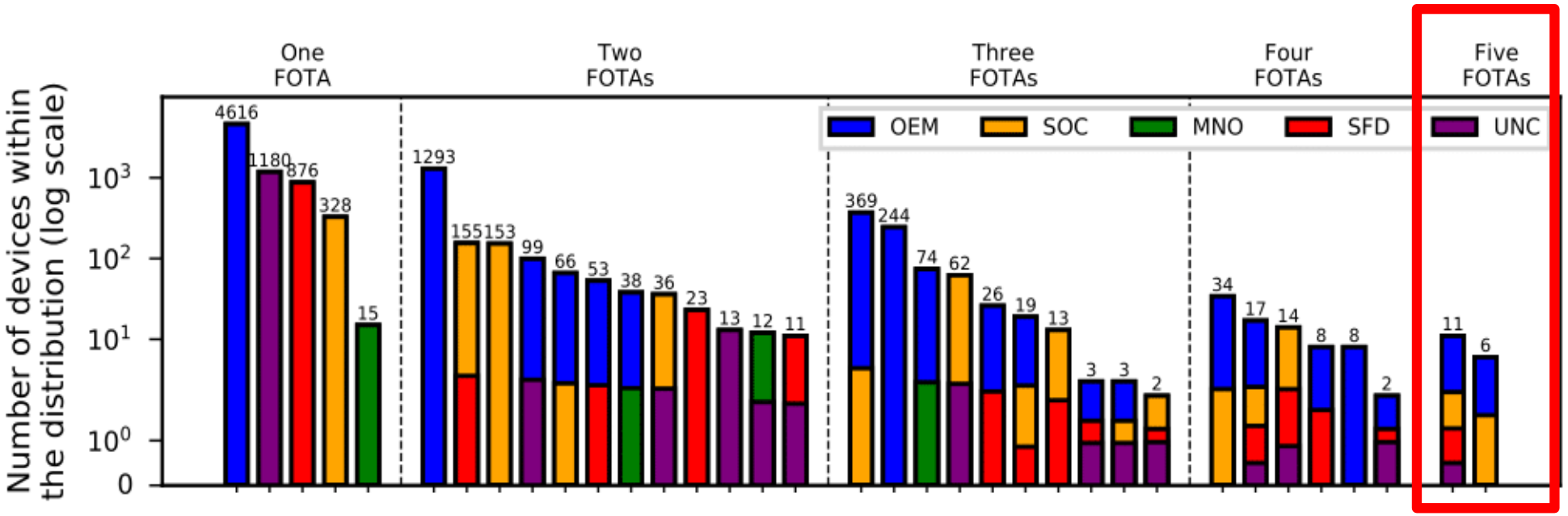
Distribution of FOTA Stakeholders in Devices



Distribution of FOTA Stakeholders in Devices



Distribution of FOTA Stakeholders in Devices



Security Implications

- FOTA Apps signed with AOSP **default** test key

Package	# dev.
com.adups.fota.sysoper	98
com.mediatek.systemupdate.sysoper	16
pl.zdunex25.updater	13
com.abastra.android.goclever.otaupdate	11
com.mediatek.googleota.sysoper	10
com.redstone.ota.ui	8
com.freeme.ota	6
com.fw.upgrade.sysoper	4
com.fota.wirelessupdate	3
org.pixelexperience.ota	3
com.android.settings	2
com.adups.fota	1
com.rock.gota	1

Brand	# dev.
Alps	80
Xiaomi	16
Samsung	12
Goclever	11
Allview	10
Doogee	9
Iku	8
Blackview	6
Bravis	6
Cubot	3
Elite_5	2
BQ	2
Others (9)	11

Security Implications

- FOTA Apps signed with AOSP **default** test key

Package	# dev.
com.adups.fota.sysoper	98
com.mediatek.systemupdate.sysoper	16
pl.zdunex25.updater	13
com.abastra.android.goclever.otaupdate	11
com.mediatek.googleota.sysoper	10
com.redstone.ota.ui	8
com.freeme.ota	6
com.fw.upgrade.sysoper	4
com.fota.wirelessupdate	3
org.pixelexperience.ota	3
com.android.settings	2
com.adups.fota	1
com.rock.gota	1

Brand	# dev.
Alps	80
Xiaomi	16
Samsung	12
Goclever	11
Allview	10
Doogee	9
Iku	8
Blackview	6
Bravis	6
Cubot	3
Elite_5	2
BQ	2
Others (9)	11

Static Analysis of FOTA Behavior

Accessed data type / behaviors		% Apps (#)	% Third-party (#)
Telephony identifiers	IMEI	33.7 (577)	15.2 (260)
	IMSI	31.4 (538)	8.2 (140)
	Phone number	8.8 (151)	4.4 (75)
	MCC & MNC	19.1 (327)	6.3 (108)
	Operator name	5.7 (98)	3.3 (56)
	SIM Serial number	6.5 (111)	2.7 (446)
	SIM State	13.1 (224)	4.5 (77)
	Current country	6.7 (115)	1.3 (22)
	SIM country	7.6 (131)	3.2 (55)
Device settings	Software version	1.0 (17)	1.0 (17)
	Phone state	25.1 (430)	5.5 (95)
	Installed apps	49.2 (843)	17.9 (307)
	Phone type	14.4 (247)	8.3 (143)
	Logs	65.3 (1,119)	24.8 (425)
Location	GPS	0.7 (12)	0.6 (11)
	Cell location	4.3 (73)	2.7 (47)
	CID	4.8 (82)	2.6 (44)
	LAC	3.7 (63)	2.0 (34)

Accessed data type / behaviors		% Apps (#)	% Third-party (#)
Network interfaces	Wi-Fi configuration	2.0 (35)	1.9 (32)
	Current network	50.0 (856)	15.1 (259)
	Data plan	34.9 (598)	8.9 (153)
	Connection state	4.3 (73)	1.7 (29)
	Network type	17.3 (296)	6.2 (106)
Phone service abuse	SMS sending	0.1 (1)	0.0 (0)
	Phone calls	8.5 (146)	3.3 (57)
Audio/video interception	Audio recording	2.6 (44)	2.4 (41)
	Video capture	2.3 (40)	2.3 (40)
Arbitrary code execution	Native code	27.1 (465)	11.4 (196)
	Linux commands	30.9 (530)	10.8 (185)
Socket conn.	Remote connection	6.7 (114)	1.9 (32)

Static Analysis of FOTA Behavior

Accessed data type / behaviors		% Apps (#)	% Third-party (#)
Telephony identifiers	IMEI	33.7 (577)	15.2 (260)
	IMSI	31.4 (538)	8.2 (140)
	Phone number	8.8 (151)	4.4 (75)
	MCC & MNC	19.1 (327)	6.3 (108)
	Operator name	5.7 (98)	3.3 (56)
	SIM Serial number	6.5 (111)	2.7 (446)
	SIM State	13.1 (224)	4.5 (77)
	Current country	6.7 (115)	1.3 (22)
	SIM country	7.6 (131)	3.2 (55)
Device settings	Software version	1.0 (17)	1.0 (17)
	Phone state	25.1 (430)	5.5 (95)
	Installed apps	49.2 (843)	17.9 (307)
	Phone type	14.4 (247)	8.3 (143)
	Logs	65.3 (1,119)	24.8 (425)
Location	GPS	0.7 (12)	0.6 (11)
	Cell location	4.3 (73)	2.7 (47)
	CID	4.8 (82)	2.6 (44)
	LAC	3.7 (63)	2.0 (34)

Accessed data type / behaviors		% Apps (#)	% Third-party (#)
Network interfaces	Wi-Fi configuration	2.0 (35)	1.9 (32)
	Current network	50.0 (856)	15.1 (259)
	Data plan	34.9 (598)	8.9 (153)
	Connection state	4.3 (73)	1.7 (29)
	Network type	17.3 (296)	6.2 (106)
Phone service abuse	SMS sending	0.1 (1)	0.0 (0)
	Phone calls	8.5 (146)	3.3 (57)
Audio/video interception	Audio recording	2.6 (44)	2.4 (41)
	Video capture	2.3 (40)	2.3 (40)
Arbitrary code execution	Native code	27.1 (465)	11.4 (196)
	Linux commands	30.9 (530)	10.8 (185)
Socket conn.	Remote connection	6.7 (114)	1.9 (32)

Static Analysis of FOTA Behavior

Accessed data type / behaviors		% Apps (#)	% Third-party (#)
Telephony identifiers	IMEI	33.7 (577)	15.2 (260)
	IMSI	31.4 (538)	8.2 (140)
	Phone number	8.8 (151)	4.4 (75)
	MCC & MNC	19.1 (327)	6.3 (108)
	Operator name	5.7 (98)	3.3 (56)
	SIM Serial number	6.5 (111)	2.7 (446)
	SIM State	13.1 (224)	4.5 (77)
	Current country	6.7 (115)	1.3 (22)
	SIM country	7.6 (131)	3.2 (55)
Device settings	Software version	1.0 (17)	1.0 (17)
	Phone state	25.1 (430)	5.5 (95)
	Installed apps	49.2 (843)	17.9 (307)
	Phone type	14.4 (247)	8.3 (143)
	Logs	65.3 (1,119)	24.8 (425)
Location	GPS	0.7 (12)	0.6 (11)
	Cell location	4.3 (73)	2.7 (47)
	CID	4.8 (82)	2.6 (44)
	LAC	3.7 (63)	2.0 (34)

Accessed data type / behaviors		% Apps (#)	% Third-party (#)
Network interfaces	Wi-Fi configuration	2.0 (35)	1.9 (32)
	Current network	50.0 (856)	15.1 (259)
	Data plan	34.9 (598)	8.9 (153)
	Connection state	4.3 (73)	1.7 (29)
	Network type	17.3 (296)	6.2 (106)
Phone service abuse	SMS sending	0.1 (1)	0.0 (0)
	Phone calls	8.5 (146)	3.3 (57)
Audio/video interception	Audio recording	2.6 (44)	2.4 (41)
	Video capture	2.3 (40)	2.3 (40)
Arbitrary code execution	Native code	27.1 (465)	11.4 (196)
	Linux commands	30.9 (530)	10.8 (185)
Socket conn.	Remote connection	6.7 (114)	1.9 (32)

FOTA Telemetry Results

- From the 2,013 FOTA apps, we discovered data for 20 of them in 961,424 installation events.
- Interesting results:

Installer		Installations		Children	
Package name	Type	Events	Devices	APKs	Mal. APKs (%)
com.samsung.android.app.omcagent	OEM	3.0M	332K	1.9K	29 (1.5%)
com.coloros.sau	OEM	191K	65K	985	28 (3%)
com.android.settings	Unknown	35K	4.7K	1.4K	494 (35%)
com.qiku.android.ota	OEM	310	77	12	11 (92%)

Malicious Installations

Potentially Unwanted Programs (PUP)

- ❑ adware
- ❑ smsreg
- ❑ hiddad

Malware families

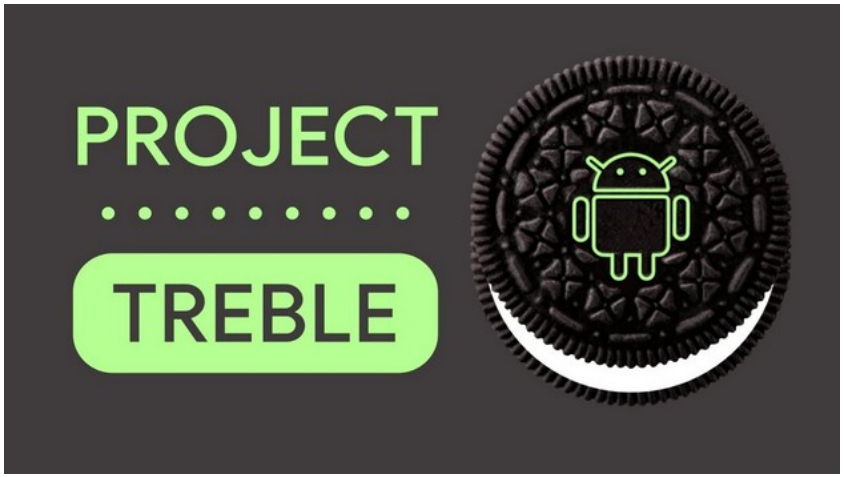
- ❑ triada
- ❑ necro
- ❑ guerilla

Recommendations & Current Improvement Efforts

Although this is not an easy to solve problem, we recommend:

- Following best practices in FOTA development
- Increase transparency through public documentation
- Separate system from non-system installations

Current Google improvement efforts



Recommendations & Current Improvement Efforts

- Separate system from non-system installations

Recommendations & Current Improvement Efforts

Recommendations & Current Improvement Efforts

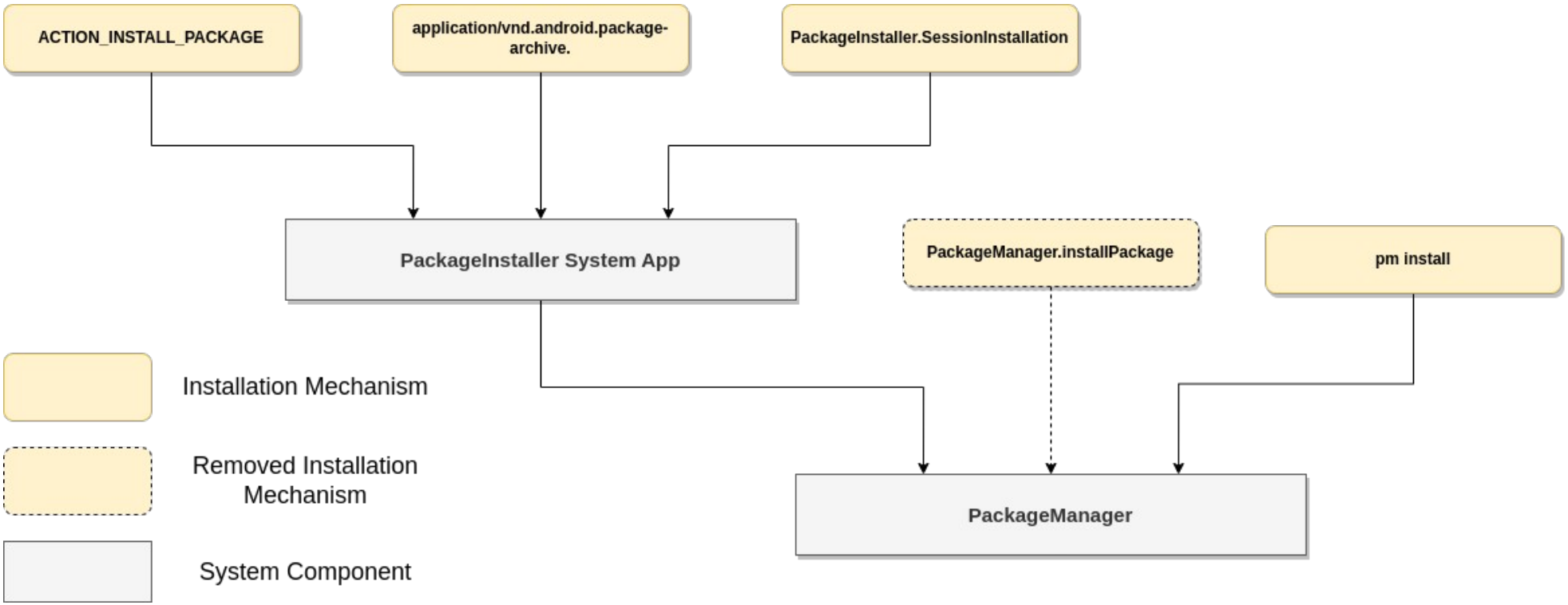
- Updates are not the only artifact that can be distributed over the air
- We found installation capabilities that were interesting to study (we call apps with these capabilities OTA)
- FOTA research didn't focus on these capabilities, this gap led us to our next research

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

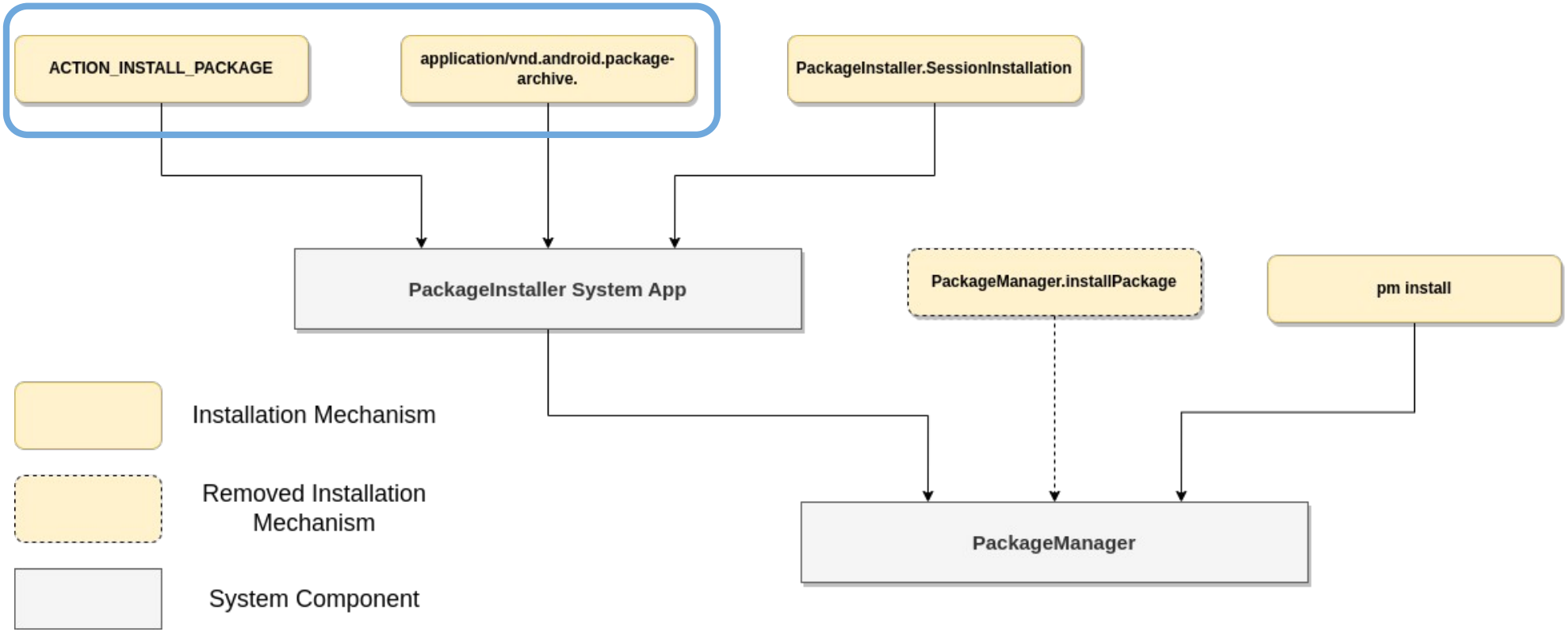
OTA (Over-The-Air) Installations

- Users can download apps from well-known markets .
- Third-party markets exist, and they offer same, similar or even “cracked” versions of legitimate apps.
- Non-market applications (from user devices) can also install other apps, or update themselves.

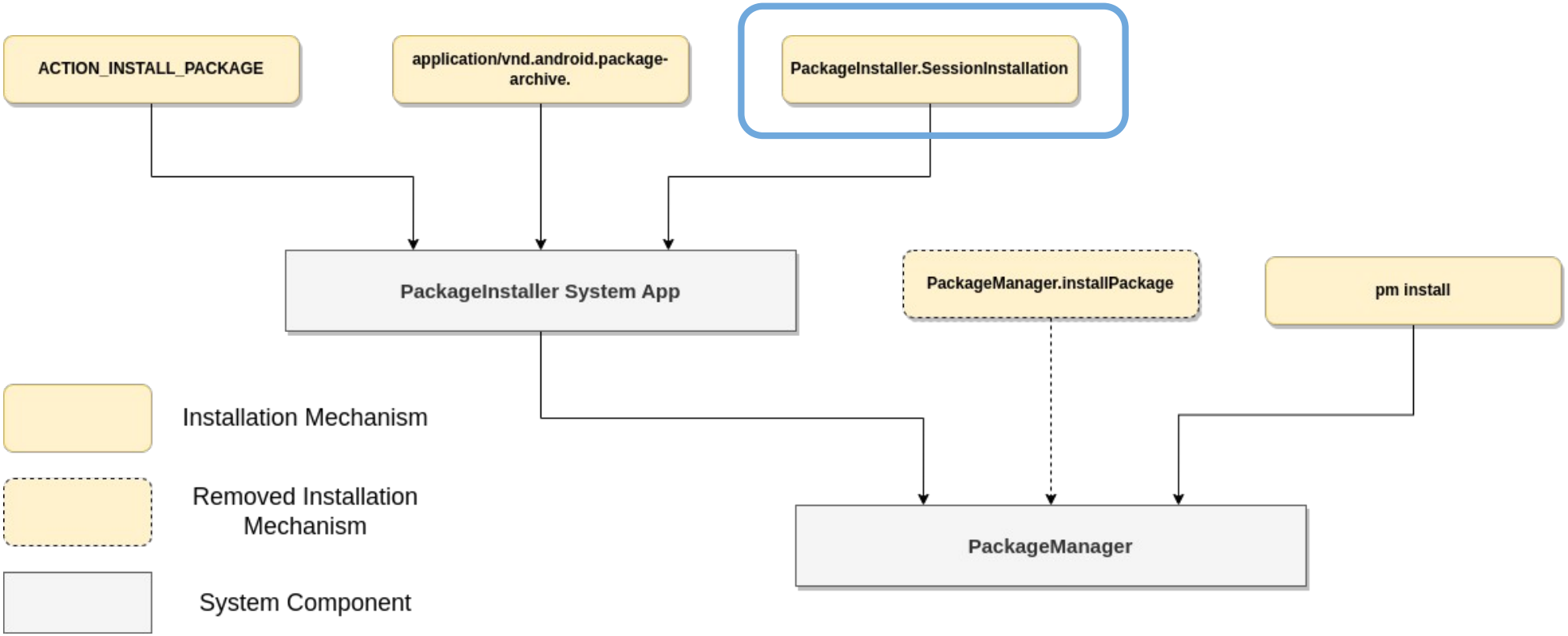
User-level Installation Mechanisms



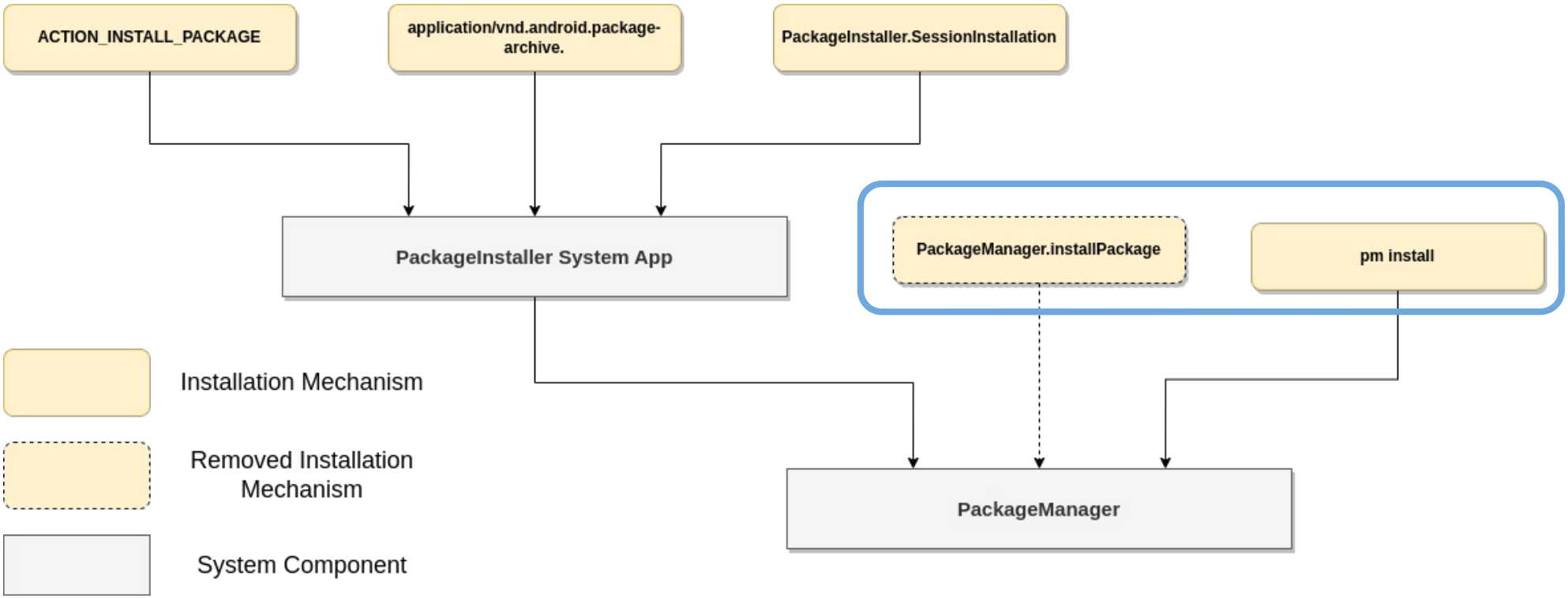
User-level Installation Mechanisms



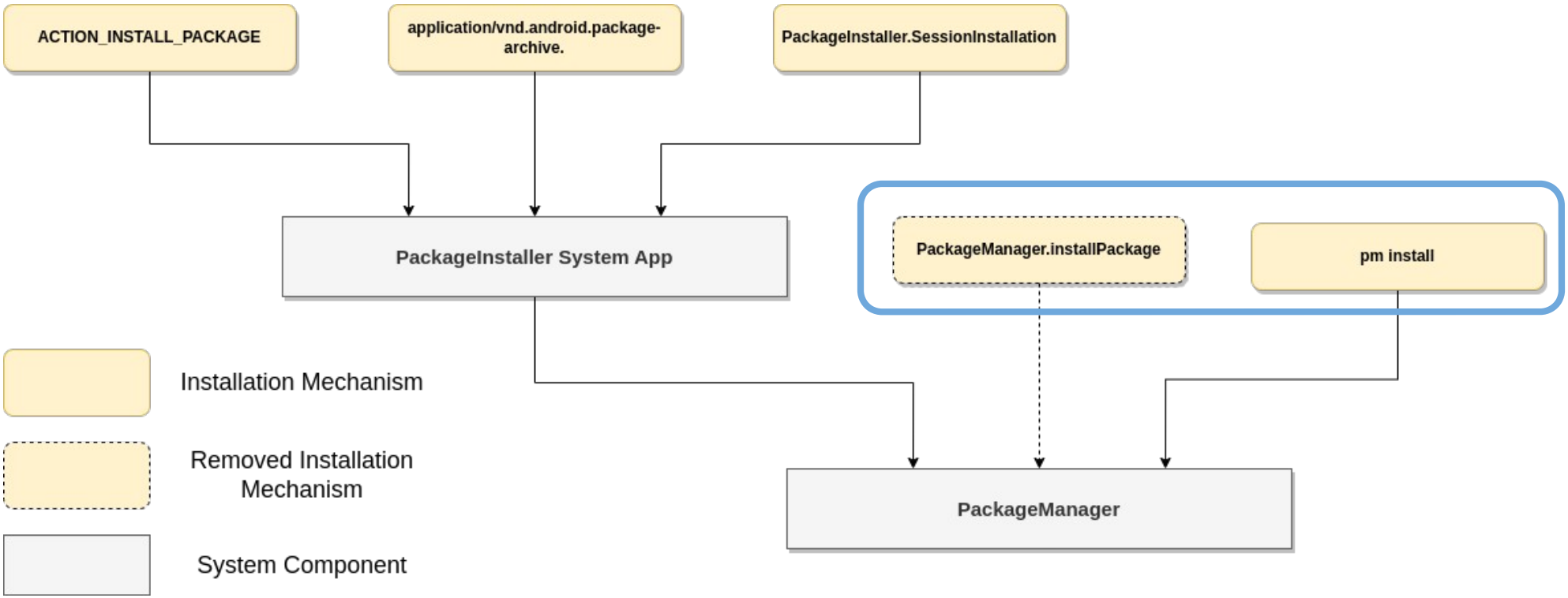
User-level Installation Mechanisms



User-level Installation Mechanisms

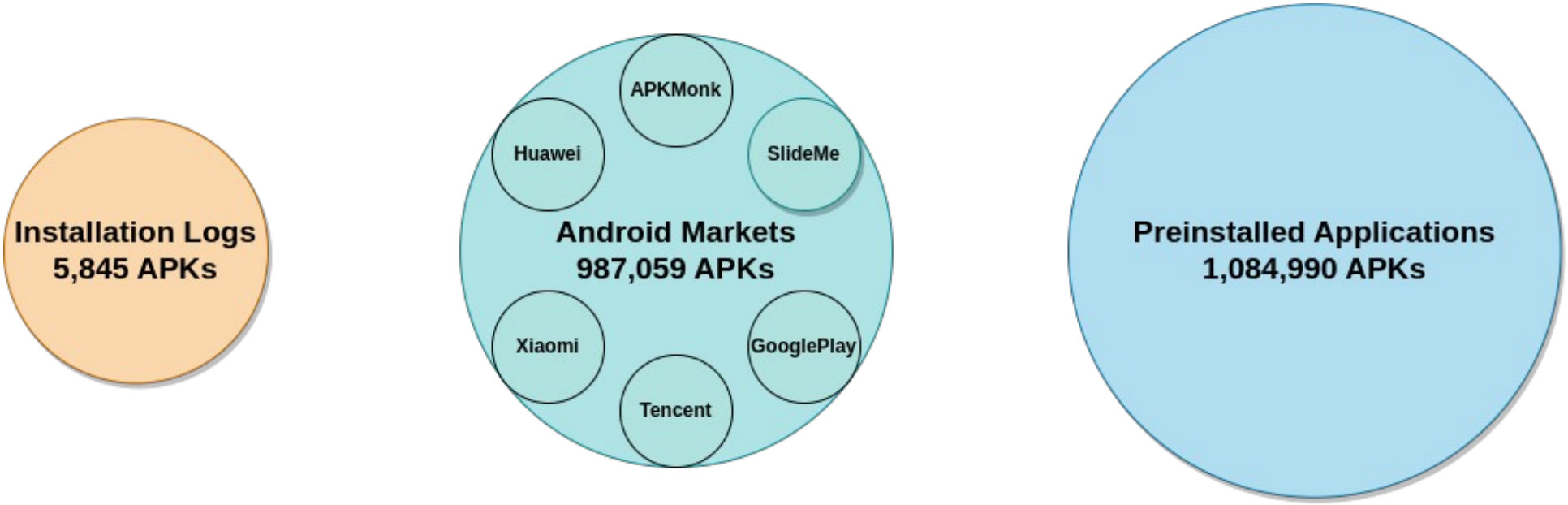


User-level Installation Mechanisms



- Prior to Android API 25
  - global *unknown sources* permission to control installation.
- After Android API 25
  - *REQUEST_INSTALL_PACKAGES* for each package.
- *INSTALL_PACKAGES* previously used for allowing silent installations.

Analysis Dataset



OTA Hunter & New Signals

- Extended version of FOTA Finder with new signals.
- Discovered apps were used for their analysis in installation events from Norton.
- OTA Hunter also retrieves the source from the signals.

Type	Signal	Silent	Description
Installation	VND*	No	Installation intents directed to the PackageManager when requesting it to install a package
	PermI*	N/A	Request of the AOSP permission `android.permission.INSTALL_PACKAGES`
	PM_I*	Yes	Call to the method `installPackage` from the API class `android.content.pm.PackageManager`
	PMI*	Yes	Use of the string `"pm install"` in the code
	PI_SI	Yes	Use of session installation method through the class `SessionParams` from the package `android.content.pm.PackageInstaller`
	IAIP	No	Use of the Intent Action *ACTION_INSTALL_PACKAGE* to install an application.
	permRI	N/A	Request of the AOSP permission `android.permission.REQUEST_INSTALL_PACKAGES`
Uninstallation	PermD*	N/A	Request of the AOSP permission `android.permission.DELETE_PACKAGES`
	PM_D*	Yes	Call to the method `deletePackage` from the API class `anroid.content.pm.PackageManager`
	PMU	Yes	Use of the string `"pm uninstall"` in the code
	PI_U	Yes	Call to the method `uninstall` from the API class `android.content.pm.PackageInstaller`
	IAUP	No	Use of Intent action *ACTION_UNINSTALL_PACKAGE* to uninstall an application.
	permRD	N/A	Request of the AOSP permission `android.permission.REQUEST_DELETE_PACKAGES`

* Signals already present in FOTAFinder

OTA Signals Prevalence

	Signal	#OTAs (#FOTAs)			
		Pre-Installed	Play Store	Other Markets	Installation Logs
Install	*VND*	83,689 (1,119)	31,339 (6)	66,756 (15)	1,095 (6)
	PermI	43,014 (1,133)	337 (0)	20,762 (5)	261 (13)
	permRI	12,258 (54)	3,332 (0)	42,778 (4)	1,174 (5)
	IAIP	7,793 (141)	3,710 (0)	32,672 (2)	720 (4)
	PI_SI	23,844 (2,315)	197 (1)	230 (3)	420 (13)
	PM_I	12,952 (509)	16 (0)	37 (0)	31 (5)
	PMI	2,181 (304)	461 (8)	5,153 (1)	134 (1)
Uninstall	*PermD*	40,189 (623)	211 (0)	685 (5)	156 (10)
	IAUP	11,573 (555)	2,119 (0)	1,497 (2)	195 (2)
	PM_D	11,605 (211)	17 (0)	177 (0)	39 (5)
	permRD	5,953 (306)	126 (0)	516 (2)	321 (2)
	PI_U	5,264 (154)	102 (0)	140 (0)	69 (2)
	PMU	1,507 (165)	540 (0)	419 (1)	91 (1)
	Total	132,916 (8,275)	32,827 (19)	85,817 (28)	1,519 (20)

OTA Signals Prevalence

	Signal	#OTAs (#FOTAs)			
		Pre-Installed	Play Store	Other Markets	Installation Logs
Install	*VND*	83,689 (1,119)	31,339 (6)	66,756 (15)	1,095 (6)
	PermI	43,014 (1,133)	337 (0)	20,762 (5)	261 (13)
	permRI	12,258 (54)	3,332 (0)	42,778 (4)	1,174 (5)
	IAIP	7,793 (141)	3,710 (0)	32,672 (2)	720 (4)
	PI_SI	23,844 (2,315)	197 (1)	230 (3)	420 (13)
	PM_I	12,952 (509)	16 (0)	37 (0)	31 (5)
	PMI	2,181 (304)	461 (8)	5,153 (1)	134 (1)
Uninstall	*PermD*	40,189 (623)	211 (0)	685 (5)	156 (10)
	IAUP	11,573 (555)	2,119 (0)	1,497 (2)	195 (2)
	PM_D	11,605 (211)	17 (0)	177 (0)	39 (5)
	permRD	5,953 (306)	126 (0)	516 (2)	321 (2)
	PI_U	5,264 (154)	102 (0)	140 (0)	69 (2)
	PMU	1,507 (165)	540 (0)	419 (1)	91 (1)
	Total	132,916 (8,275)	32,827 (19)	85,817 (28)	1,519 (20)

OTA Signals Prevalence

	Signal	#OTAs (#FOTAs)			
		Pre-Installed	Play Store	Other Markets	Installation Logs
Install	*VND*	83,689 (1,119)	31,339 (6)	66,756 (15)	1,095 (6)
	PermI	43,014 (1,133)	337 (0)	20,762 (5)	261 (13)
	permRI	12,258 (54)	3,332 (0)	42,778 (4)	1,174 (5)
	IAIP	7,793 (141)	3,710 (0)	32,672 (2)	720 (4)
	PI_SI	23,844 (2,315)	197 (1)	230 (3)	420 (13)
	PM_I	12,952 (509)	16 (0)	37 (0)	31 (5)
	PMI	2,181 (304)	461 (8)	5,153 (1)	134 (1)
Uninstall	*PermD*	40,189 (623)	211 (0)	685 (5)	156 (10)
	IAUP	11,573 (555)	2,119 (0)	1,497 (2)	195 (2)
	PM_D	11,605 (211)	17 (0)	177 (0)	39 (5)
	permRD	5,953 (306)	126 (0)	516 (2)	321 (2)
	PI_U	5,264 (154)	102 (0)	140 (0)	69 (2)
	PMU	1,507 (165)	540 (0)	419 (1)	91 (1)
	Total	132,916 (8,275)	32,827 (19)	85,817 (28)	1,519 (20)

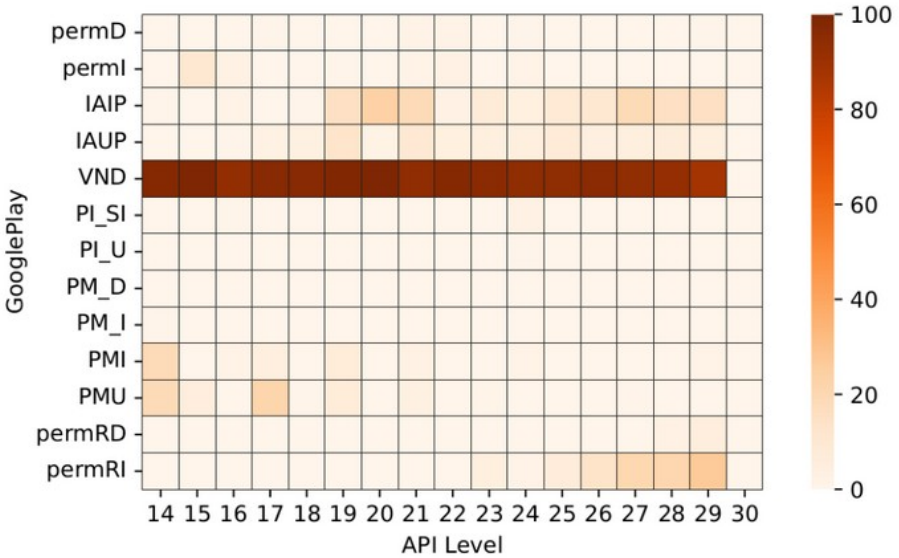
OTA Signals Prevalence

	Signal	#OTAs (#FOTAs)			
		Pre-Installed	Play Store	Other Markets	Installation Logs
Install	*VND*	83,689 (1,119)	31,339 (6)	66,756 (15)	1,095 (6)
	PermI	43,014 (1,133)	337 (0)	20,762 (5)	261 (13)
	permRI	12,258 (54)	3,332 (0)	42,778 (4)	1,174 (5)
	IAIP	7,793 (141)	3,710 (0)	32,672 (2)	720 (4)
	PI_SI	23,844 (2,315)	197 (1)	230 (3)	420 (13)
	PM_I	12,952 (509)	16 (0)	37 (0)	31 (5)
	PMI	2,181 (304)	461 (8)	5,153 (1)	134 (1)
Uninstall	*PermD*	40,189 (623)	211 (0)	685 (5)	156 (10)
	IAUP	11,573 (555)	2,119 (0)	1,497 (2)	195 (2)
	PM_D	11,605 (211)	17 (0)	177 (0)	39 (5)
	permRD	5,953 (306)	126 (0)	516 (2)	321 (2)
	PI_U	5,264 (154)	102 (0)	140 (0)	69 (2)
	PMU	1,507 (165)	540 (0)	419 (1)	91 (1)
	Total	132,916 (8,275)	32,827 (19)	85,817 (28)	1,519 (20)

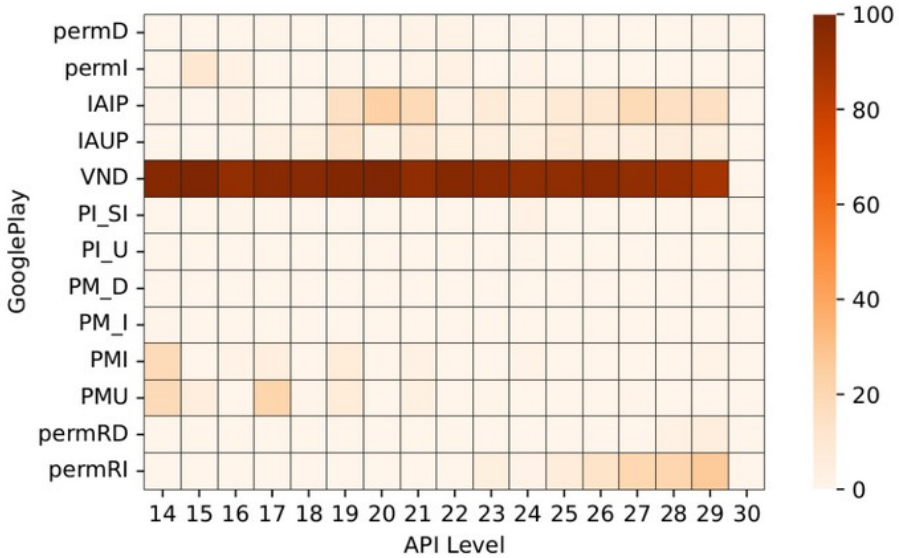
OTA Signals Prevalence

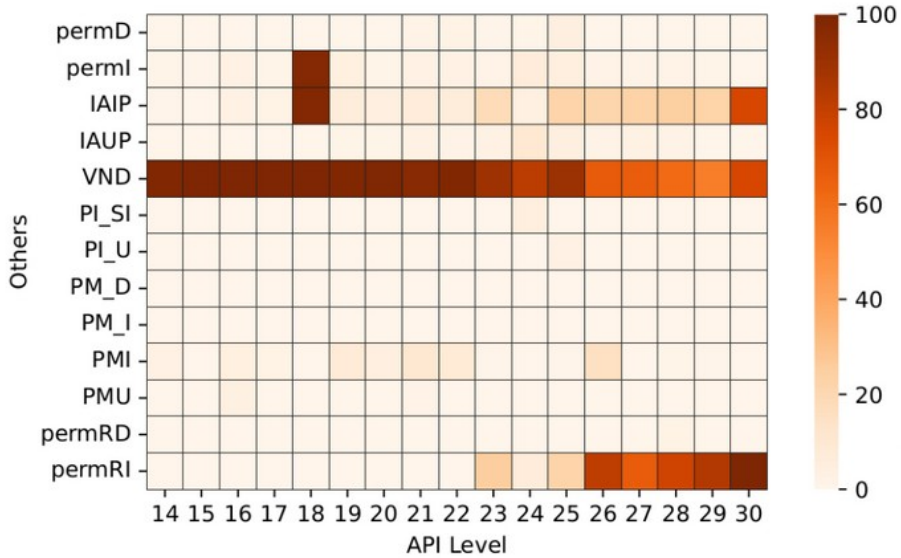
	Signal	#OTAs (#FOTAs)			
		Pre-Installed	Play Store	Other Markets	Installation Logs
Install	*VND*	83,689 (1,119)	31,339 (6)	66,756 (15)	1,095 (6)
	PermI	43,014 (1,133)	337 (0)	20,762 (5)	261 (13)
	permRI	12,258 (54)	3,332 (0)	42,778 (4)	1,174 (5)
	IAIP	7,793 (141)	3,710 (0)	32,672 (2)	720 (4)
	PI_SI	23,844 (2,315)	197 (1)	230 (3)	420 (13)
	PM_I	12,952 (509)	16 (0)	37 (0)	31 (5)
	PMI	2,181 (304)	461 (8)	5,153 (1)	134 (1)
Uninstall	*PermD*	40,189 (623)	211 (0)	685 (5)	156 (10)
	IAUP	11,573 (555)	2,119 (0)	1,497 (2)	195 (2)
	PM_D	11,605 (211)	17 (0)	177 (0)	39 (5)
	permRD	5,953 (306)	126 (0)	516 (2)	321 (2)
	PI_U	5,264 (154)	102 (0)	140 (0)	69 (2)
	PMU	1,507 (165)	540 (0)	419 (1)	91 (1)
	Total	132,916 (8,275)	32,827 (19)	85,817 (28)	1,519 (20)

Evolution of Signals Over Time

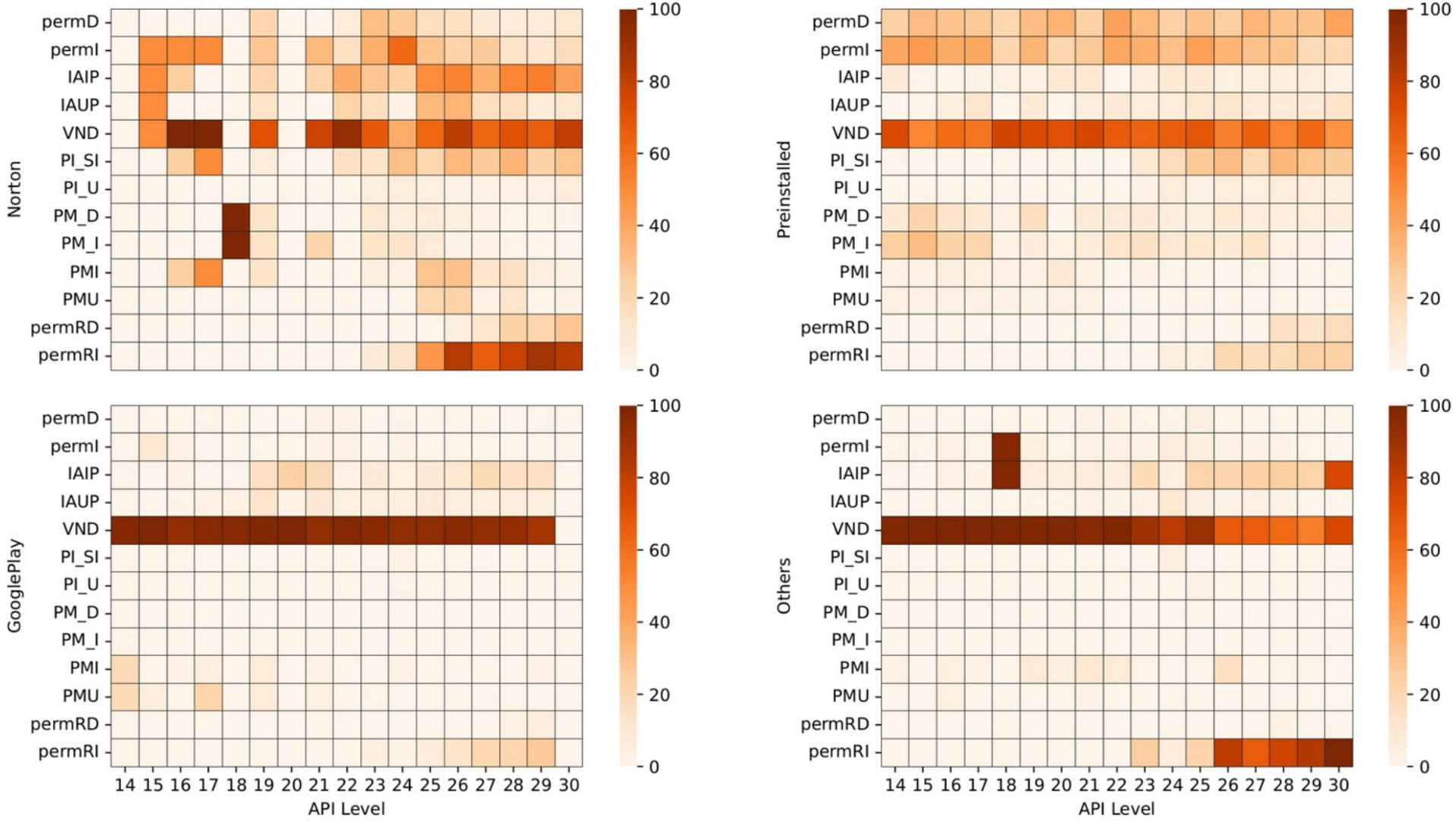


Evolution of Signals Over Time





Evolution of Signals Over Time



Interesting Findings: Use of Malware Certificate

CoolReaper certificate:
5D:F8:F0:82:12:61:A2:34:D1:11:02:8E:FD:DF:FA:3C:88:89:76:49

- com.qiku.bbs
- com.qiku.gamecenter
- com.yulong.android.coolmart
- com.icoolme.android.weather

Interesting Findings: Silent Installation in SDKs

SDKs:

- Baidu
- TCL Technology
- Xuan Yi Xia
- Tencent

```
public static void installApk(String filePath)
        throws IOException, InterruptedException {
    if (TextUtils.isEmpty(filePath) || !new File(filePath).exists()) return;
    execRootCmdSilent("pm install -r " + filePath);
}
```

```
protected static int execRootCmdSilent(String paramString) {
 try {
  Process localProcess = Runtime.getRuntime().exec("su");
  Object localObject = localProcess.getOutputStream();
  DataOutputStream localDataOutputStream = new DataOutputStream(
    (OutputStream) localObject);
  String str = String.valueOf(paramString);
  localObject = str + "\n";
  localDataOutputStream.writeBytes((String) localObject);
  localDataOutputStream.flush();
  localDataOutputStream.writeBytes("exit\n");
  localDataOutputStream.flush();
  localProcess.waitFor();
  localObject = localProcess.exitValue();
  return (Integer) localObject;
 } catch (Exception localException) {
  localException.printStackTrace();
 }
 return 0;
}
```

Installers Observed in the Telemetry

Installer			Installations		Children				
Installer	Cat.	SDK	Events	Devices	Pkgs	Signers	App Cat.	APKs	Mal. APKs
com.dti.att	-	-	1096865	152514	900	791	30	9115	5
com.dti.tracfone	-	-	85344	17545	417	378	24	3059	4
com.telcel.contenedor	Entert.	-	70099	12462	359	324	26	2595	2
com.claroColombia.contenedor	Tools	-	59285	11307	375	335	26	2880	3
com.dti.lenovo.tablet	-	-	16906	7099	62	59	18	585	0
com.dti.blu	-	-	6667	1395	155	146	19	767	0
com.taptap.global	-	-	6271	2621	1783	1572	8	2483	16
com.jio.mobileservices	-	-	5788	1399	135	123	18	562	6
com.orange.aura.oobe	-	IronSource	5704	2052	243	217	26	1499	3
com.aura.oobe.ml	-	IronSource	1785	1284	153	141	25	461	5
com.dti.gionee	-	-	1723	347	61	58	13	248	0
com.miui.huanji	-	-	1482	49	960	812	30	1181	56
com.dti.medion	-	-	1447	603	28	28	9	201	0
xyz.jmir.tachiyomi.mi	-	-	1439	149	195	2	1	390	0
com.dti.lava	-	-	971	230	61	59	13	195	0

Installers Observed in the Telemetry

Installer			Installations		Children				
Installer	Cat.	SDK	Events	Devices	Pkgs	Signers	App Cat.	APKs	Mal. APKs
com.dti.att	-	-	1096865	152514	900	791	30	9115	5
com.dti.tracfone	-	-	85344	17545	417	378	24	3059	4
com.telcel.contenedor	Entert.	-	70099	12462	359	324	26	2595	2
com.claroColombia.contenedor	Tools	-	59285	11307	375	335	26	2880	3
com.dti.lenovo.tablet	-	-	16906	7099	62	59	18	585	0
com.dti.blu	-	-	6667	1395	155	146	19	767	0
com.taptap.global	-	-	6271	2621	1783	1572	8	2483	16
com.jio.mobileservices	-	-	5788	1399	135	123	18	562	6
com.orange.aura.oobe	-	IronSource	5704	2052	243	217	26	1499	3
com.aura.oobe.ml	-	IronSource	1785	1284	153	141	25	461	5
com.dti.gionee	-	-	1723	347	61	58	13	248	0
com.miui.huanji	-	-	1482	49	960	812	30	1181	56
com.dti.medion	-	-	1447	603	28	28	9	201	0
xyz.jmir.tachiyomi.mi	-	-	1439	149	195	2	1	390	0
com.dti.lava	-	-	971	230	61	59	13	195	0

Installers Observed in the Telemetry

Installer			Installations		Children				
Installer	Cat.	SDK	Events	Devices	Pkgs	Signers	App Cat.	APKs	Mal. APKs
com.dti.att	-	-	1096865	152514	900	791	30	9115	5
com.dti.tracfone	-	-	85344	17545	417	378	24	3059	4
com.telcel.contenedor	Entert.	-	70099	12462	359	324	26	2595	2
com.claroColombia.contenedor	Tools	-	59285	11307	375	335	26	2880	3
com.dti.lenovo.tablet	-	-	16906	7099	62	59	18	585	0
com.dti.blu	-	-	6667	1395	155	146	19	767	0
com.taptap.global	-	-	6271	2621	1783	1572	8	2483	16
com.jio.mobileservices	-	-	5788	1399	135	123	18	562	6
com.orange.aura.oobe	-	IronSource	5704	2052	243	217	26	1499	3
com.aura.oobe.ml	-	IronSource	1785	1284	153	141	25	461	5
com.dti.gionee	-	-	1723	347	61	58	13	248	0
com.miui.huanji	-	-	1482	49	960	812	30	1181	56
com.dti.medion	-	-	1447	603	28	28	9	201	0
xyz.jmir.tachiyomi.mi	-	-	1439	149	195	2	1	390	0
com.dti.lava	-	-	971	230	61	59	13	195	0

Interesting SDK

- Digital Turbine
  - Advertising company focused on mobile devices.
  - Pre-installed on different devices through partnerships with OEMs and network carriers.
  - Allow the installation of apps through *"single-tap"* technology.

Remarks on Installation

- We have found that different mechanisms are available for applications to install apk files.
- These mechanisms are not only used in first-party code, but also in third-party code (like SDKs).
- We have observed that this code has been already used with malicious purposes (installation of unwanted software)

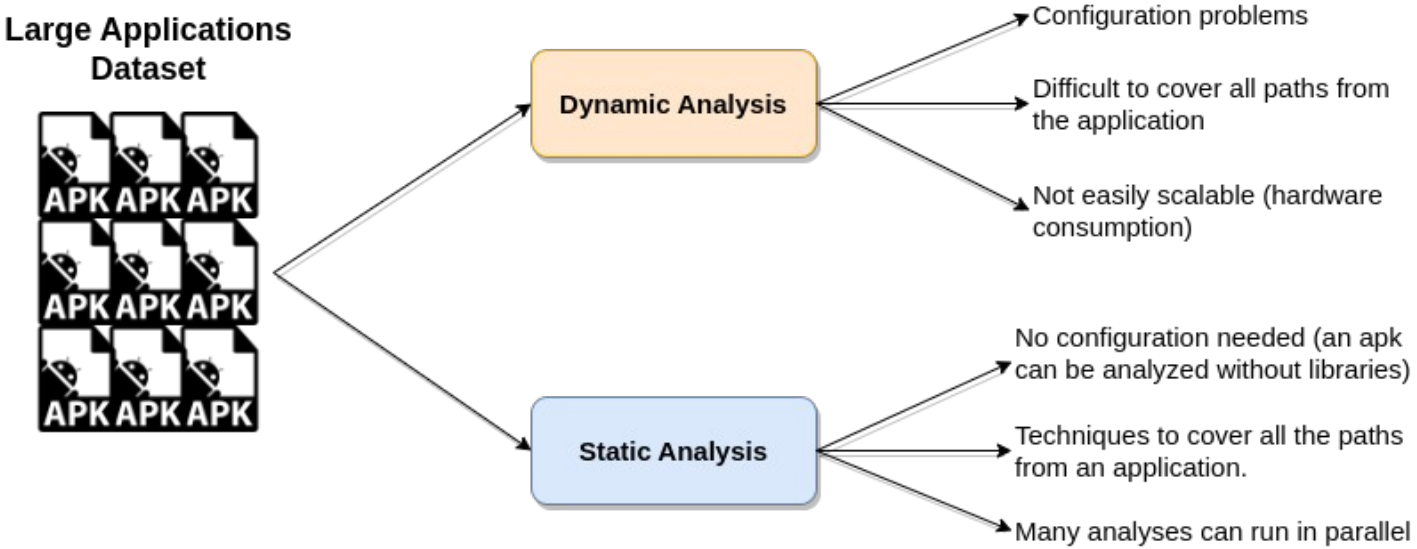
Problems Found During the Analysis

- More than 2 million apps analyzed.

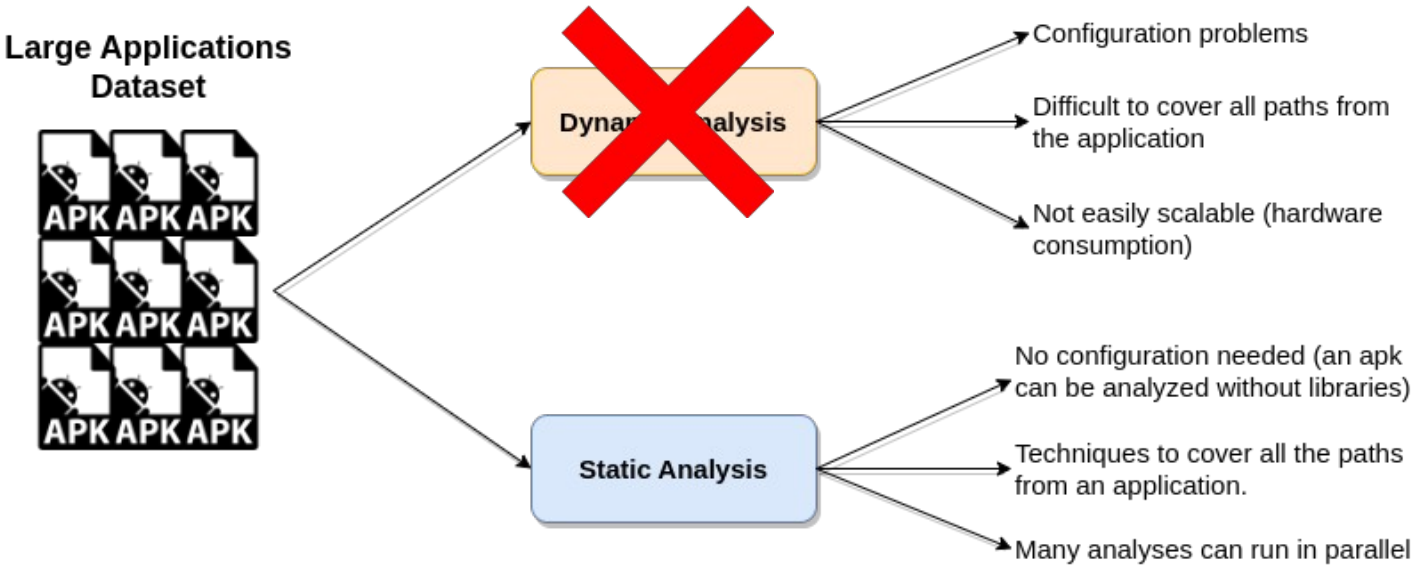
- More than one week to finish the analysis running in parallel.
- We needed a tool with a better performance for future analysis

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

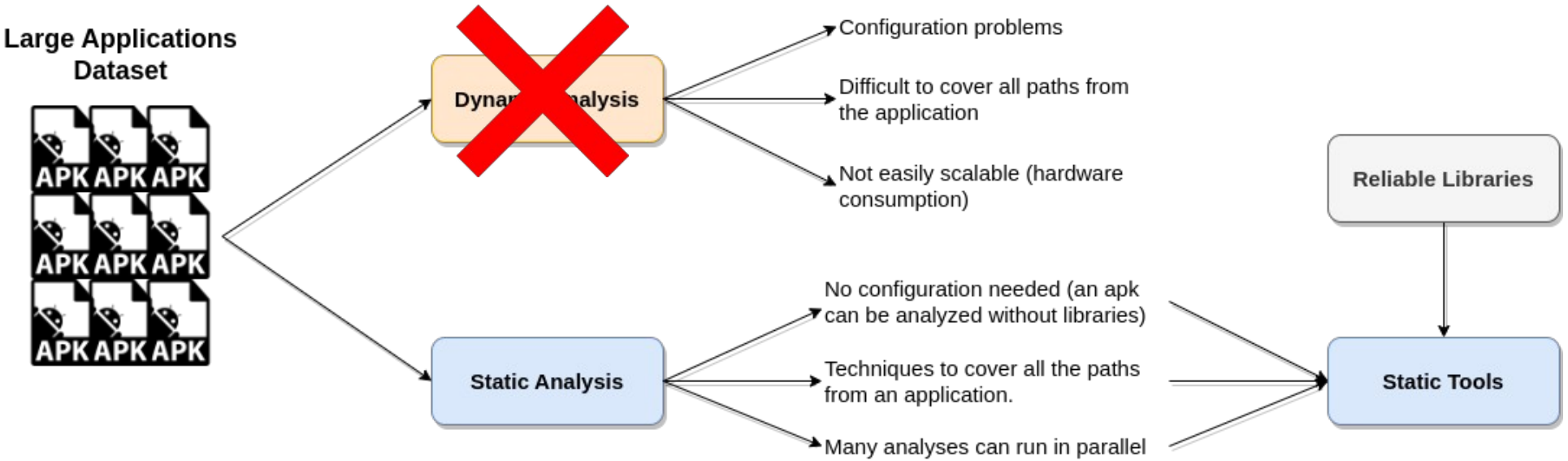
Dynamic Analysis vs. Static Analysis



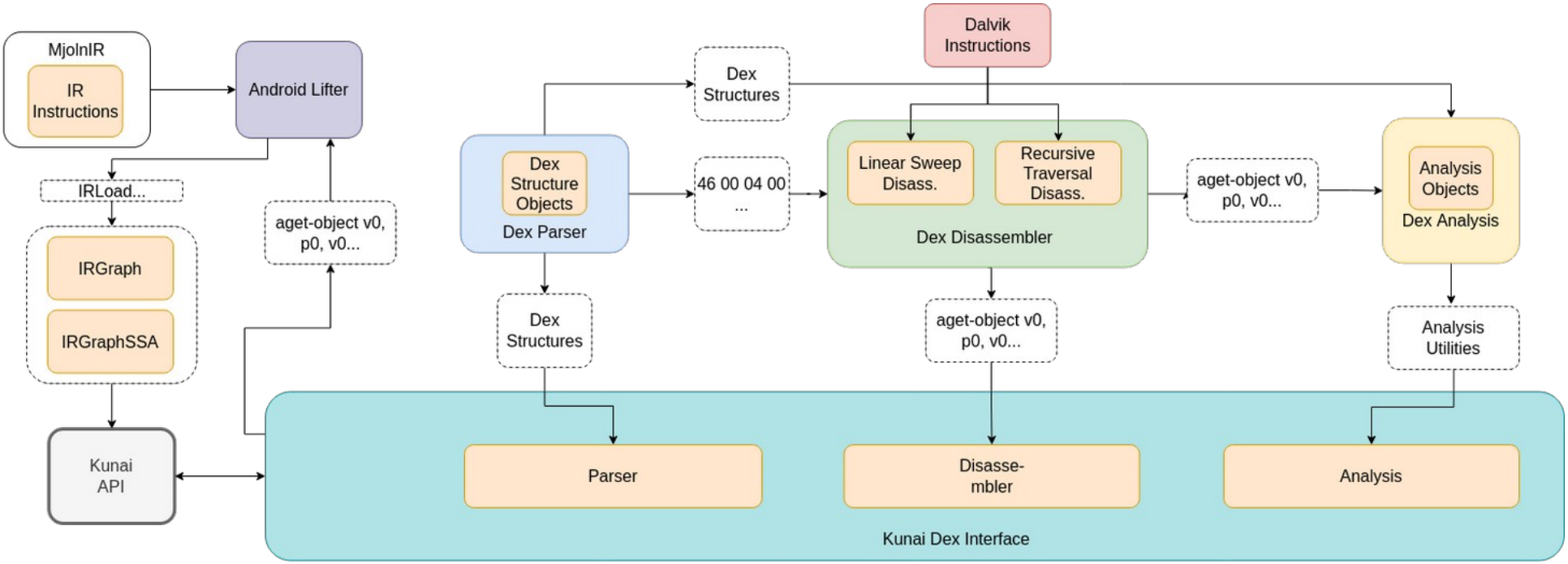
Dynamic Analysis vs. Static Analysis



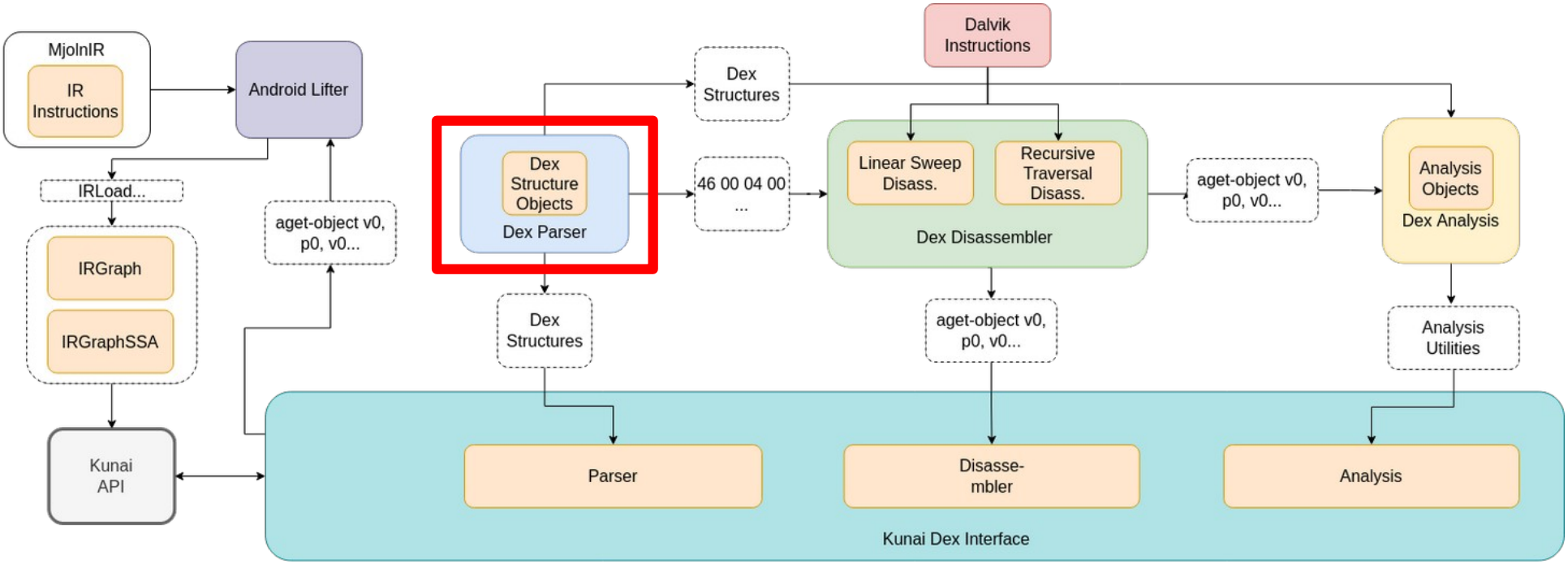
Dynamic Analysis vs. Static Analysis



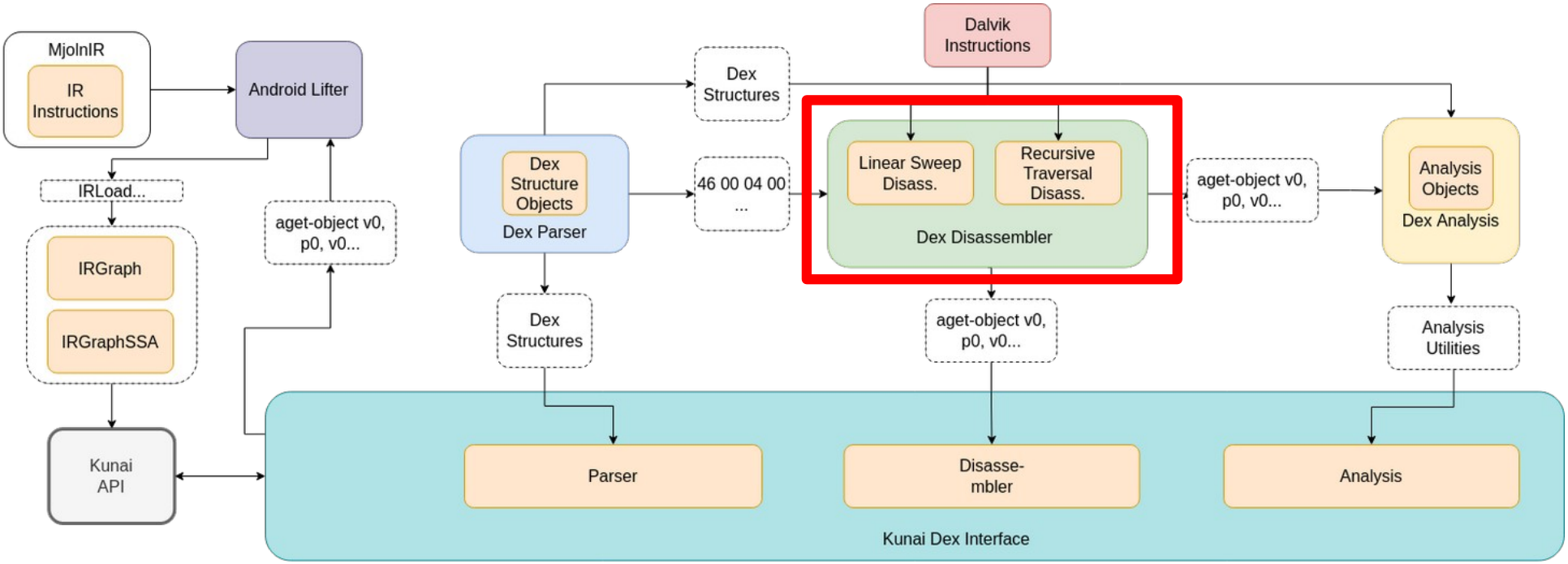
Kunai Static Analysis



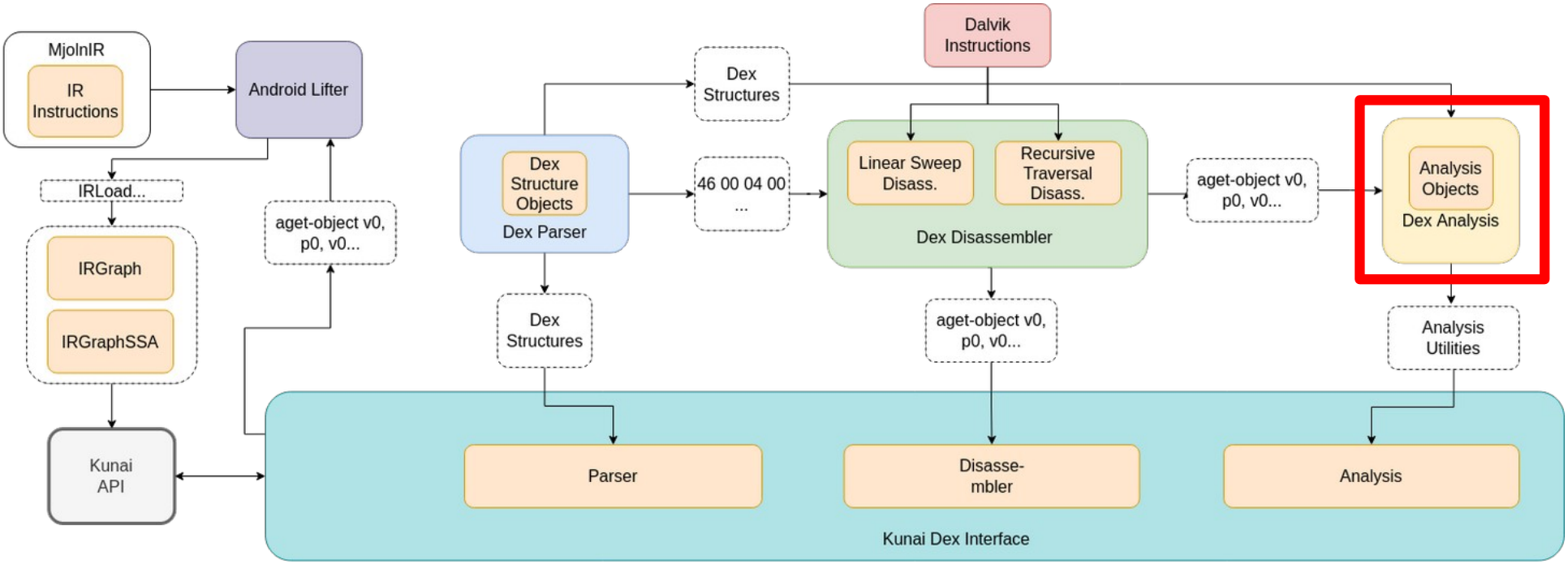
Kunai Static Analysis



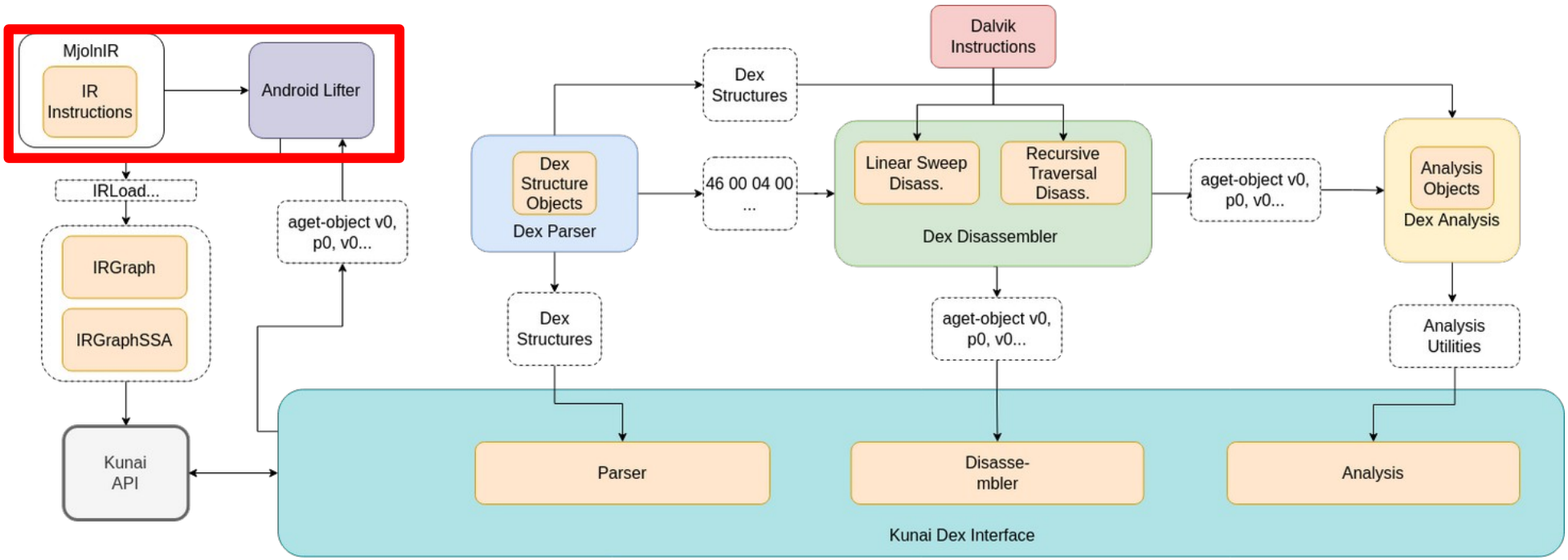
Kunai Static Analysis



Kunai Static Analysis



Kunai Static Analysis



Androguard vs. Kunai

Androguard

- ❑ Framework of analysis written in Python.
- ❑ Parsing, disassembly and analysis are always run.
- ❑ Rich API to access all the structures.
- ❑ AST only used for a basic decompilation, not accessible for analysis.

Kunai

- ❑ Shared library written in C++
- ❑ Parsing, disassembly and analysis are separated modules.
- ❑ Similar API to Androguard.
- ❑ Intermediate Representation accessible for analysis.

Analysis Dataset

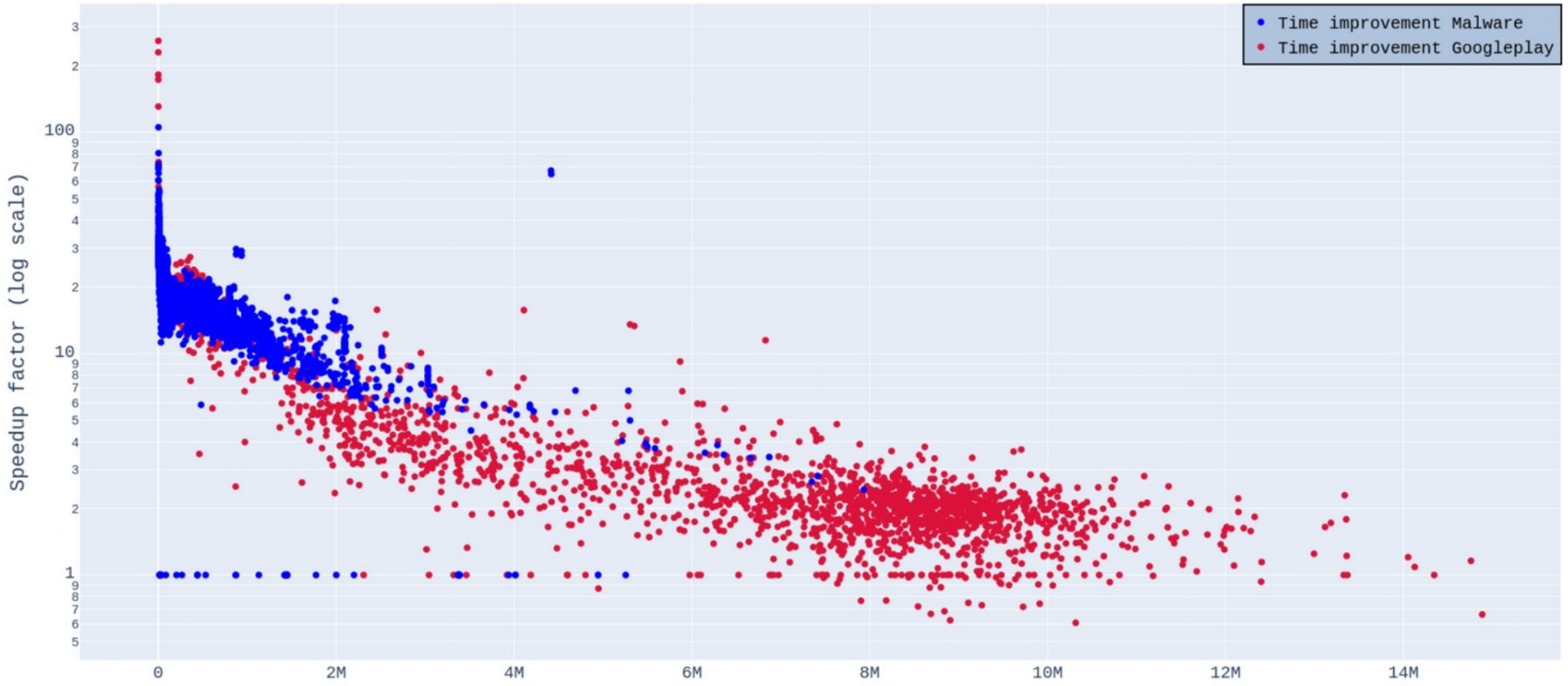
- Packages from Google Play Top-500 list.
- Crawled using an Android emulator and Pure Python ADB library.
- 396 apks, 2094 dex files (2092 dex analyzed)

- Argus malware collection with 62 different malware families.
- Downloaded from Vx-Underground.
- 8,247 apks, 8,246 dex files (8,244)

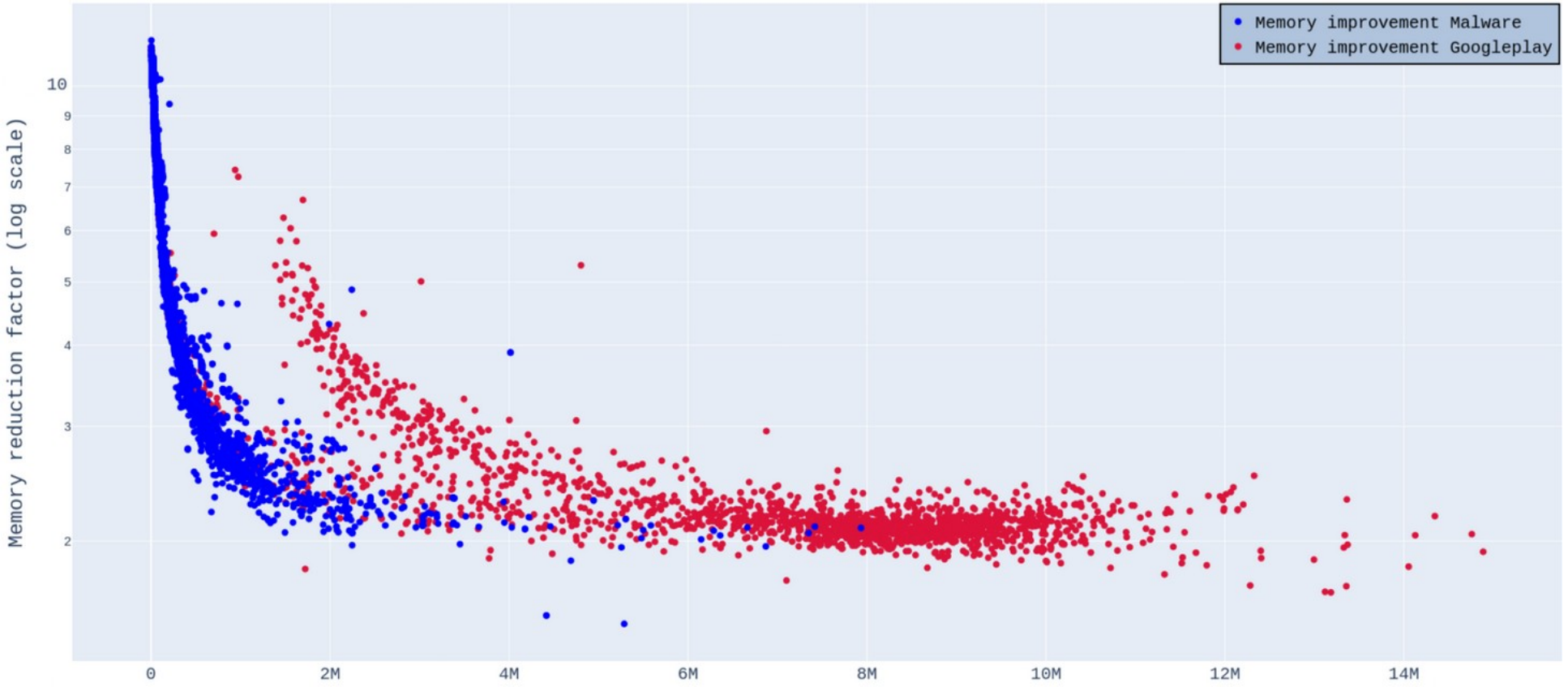
Analysis

- Our Analysis compared both frameworks using similar codes (applying: parsing, disassembly and cross-ref analysis).
- We run a sequential analysis of each DEX file with each one of the frameworks.
- We measure and compare three aspects we consider important in large-scale analysis:
  - Running time (we compare the speed-up of one tool over the other).
  - Memory footprint (we compare the usage of memory from each one)
  - Retrieved data (we compare the differences in the data obtained with each tool)

Analysis Time (Androguard vs Kunai)

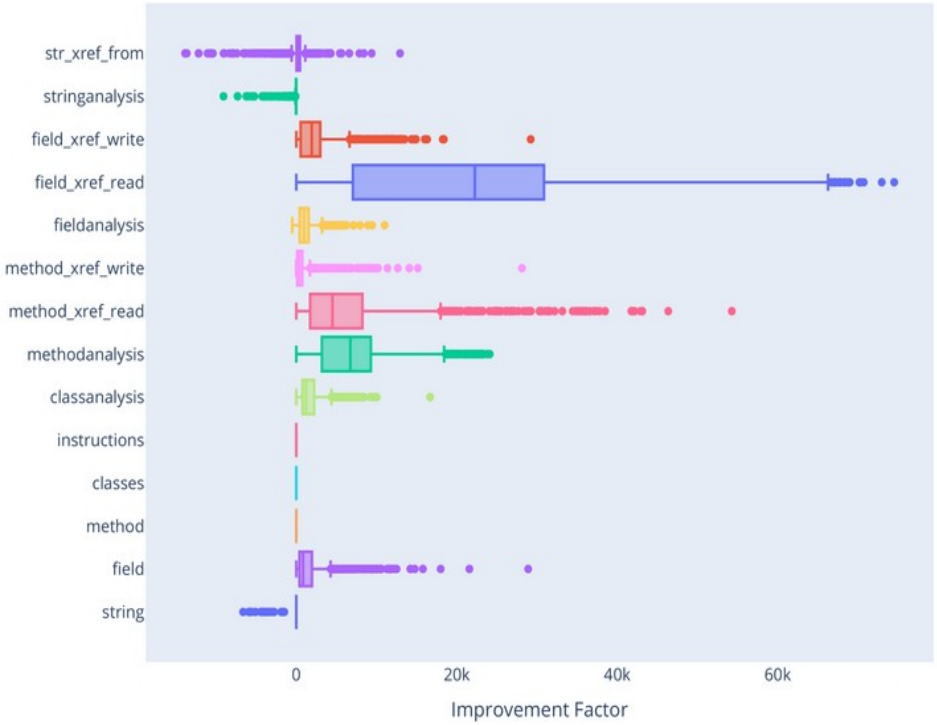


Memory Footprint (Androguard vs Kunai)

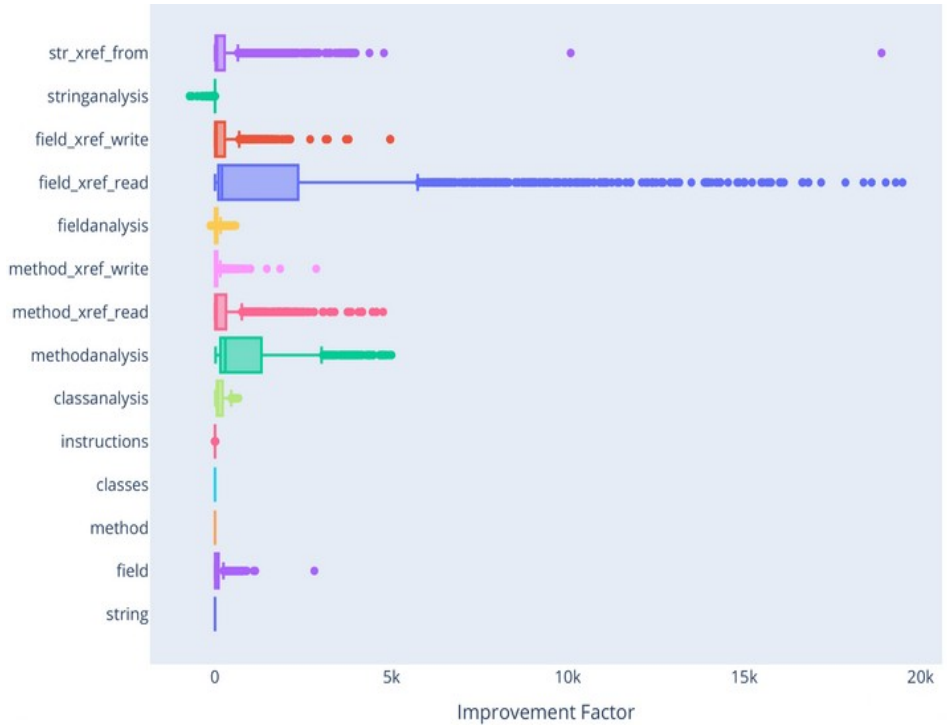


Functional Validation (Androguard vs Kunai)

Top-500 Apps

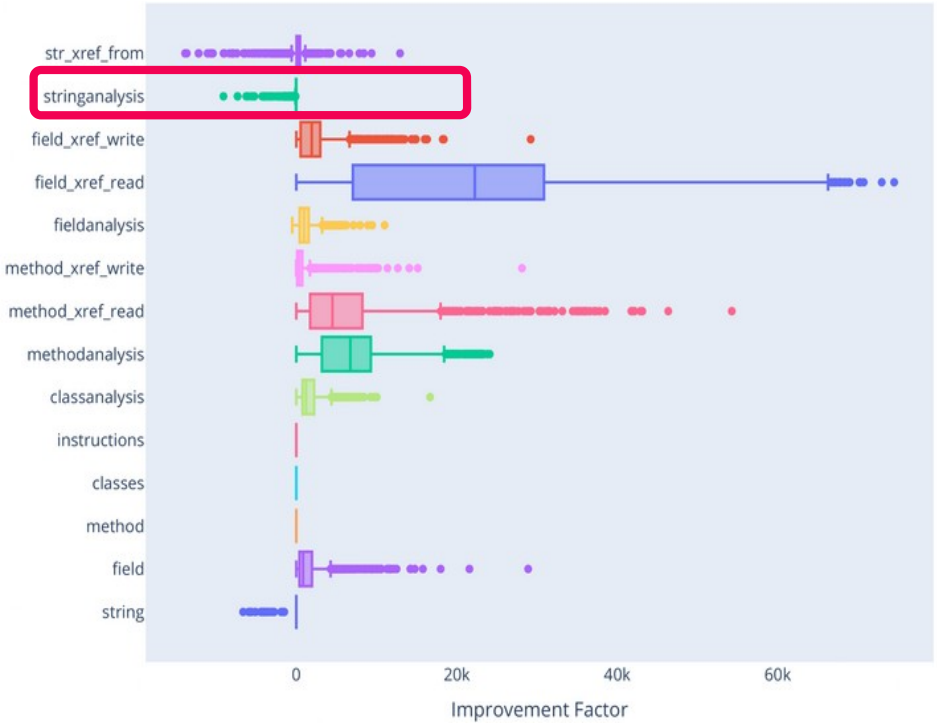


Malware Apps

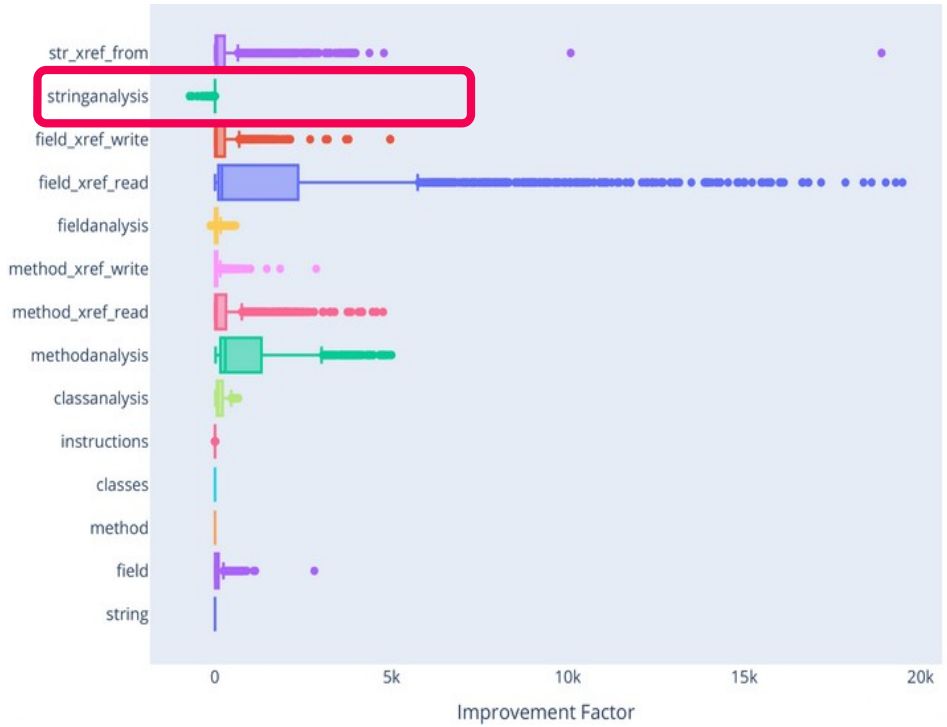


Functional Validation (Androguard vs Kunai)

Top-500 Apps

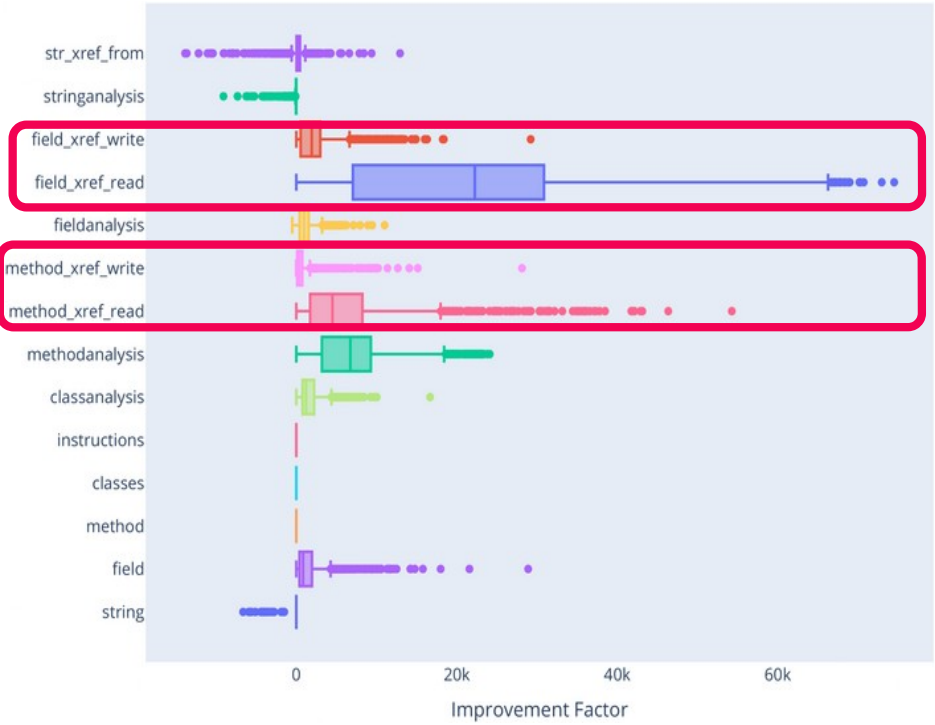


Malware Apps

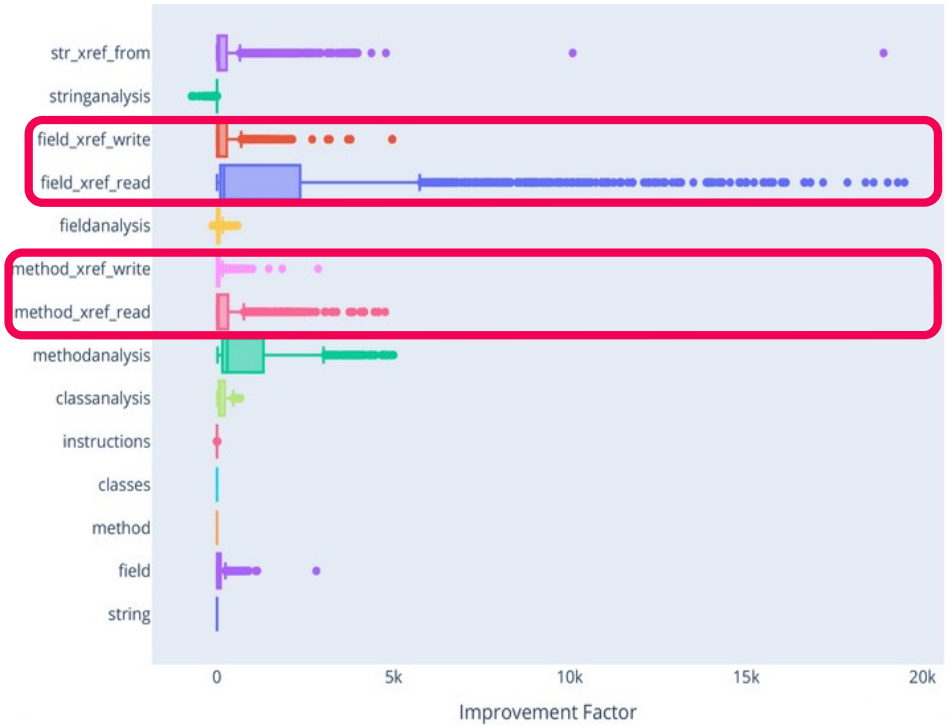


Functional Validation (Androguard vs Kunai)

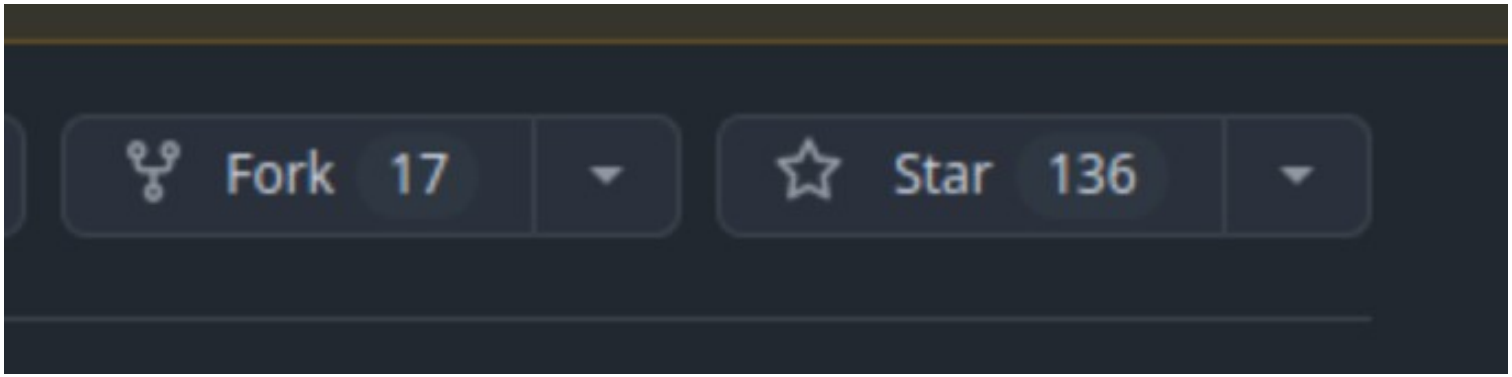
Top-500 Apps

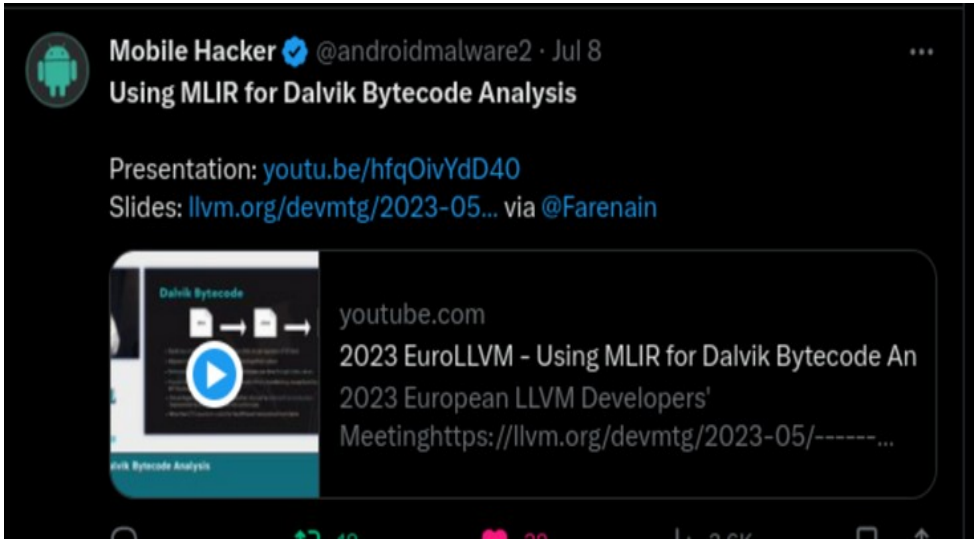


Malware Apps



Kunai Impact



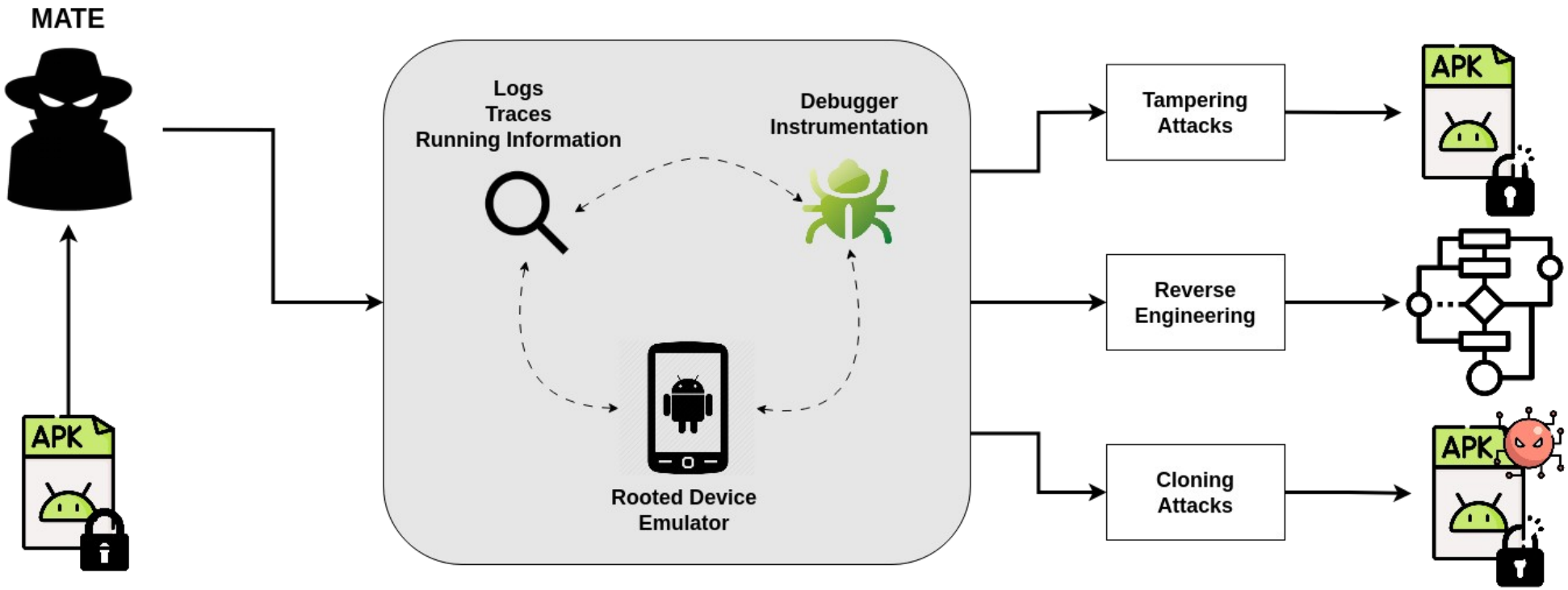


- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

Android Software Protection

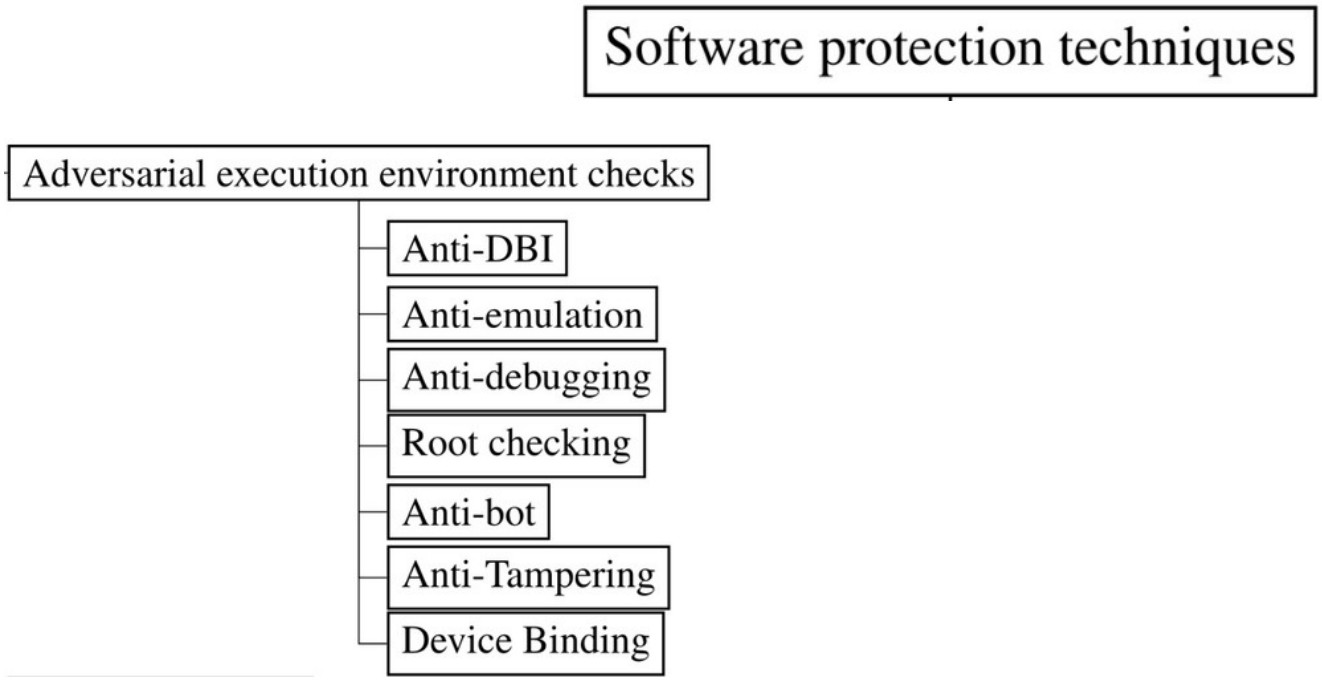
- Increasing trend on mobile phones usage as an *"all-in-one"* device.
- More applications contain sensitive information (e.g. banking, medical apps), other apps contain algorithms and intellectual property (e.g. video games).
- Nowadays existing solutions implement anti-analysis techniques, and sell these protections as SDKs, or compilers to protection applications.
- This research analyze the use of these protections in the wild, and in which applications we have detected the use of these protections.

Man-At-The-End (MATE) Attacks

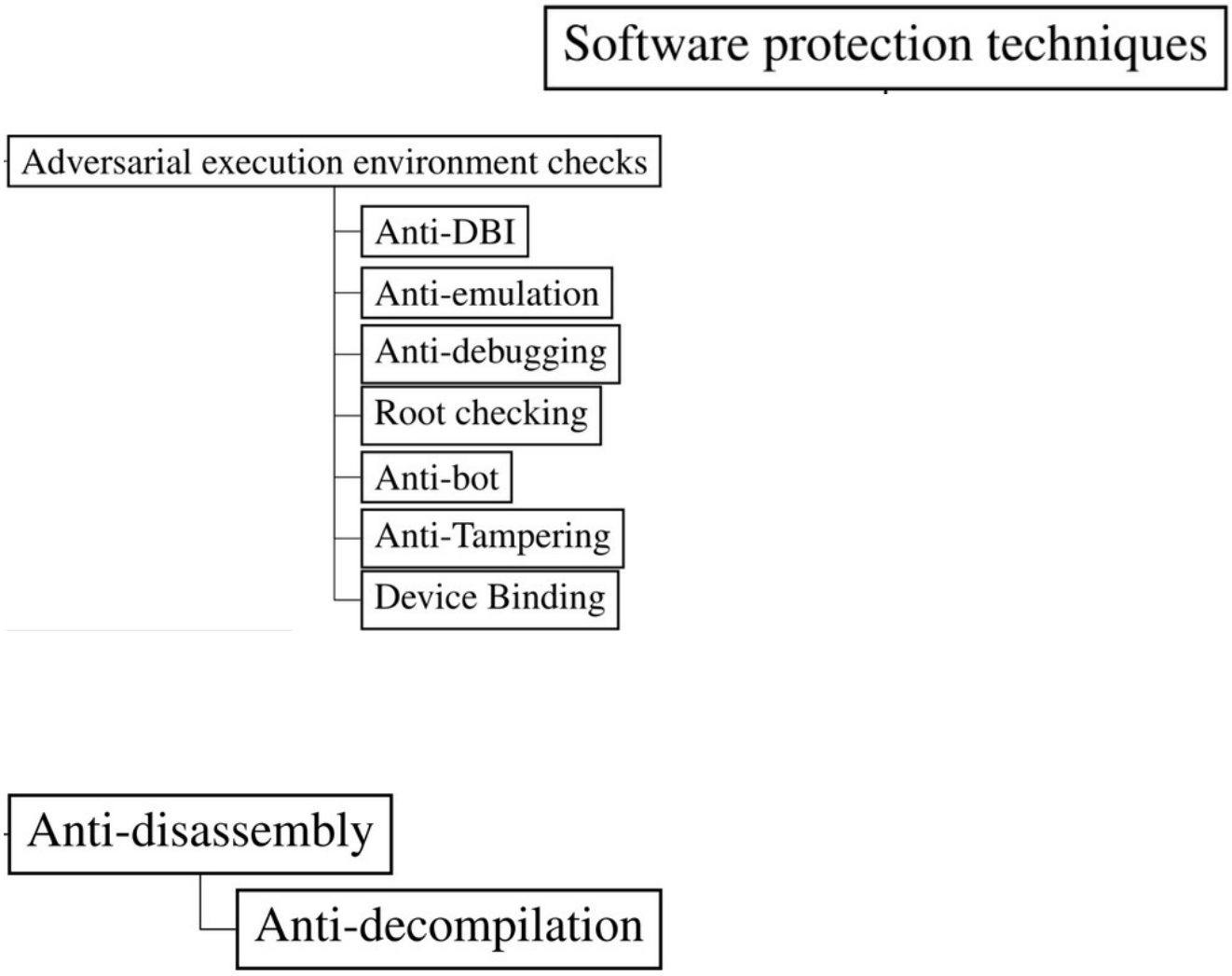


Protection Techniques

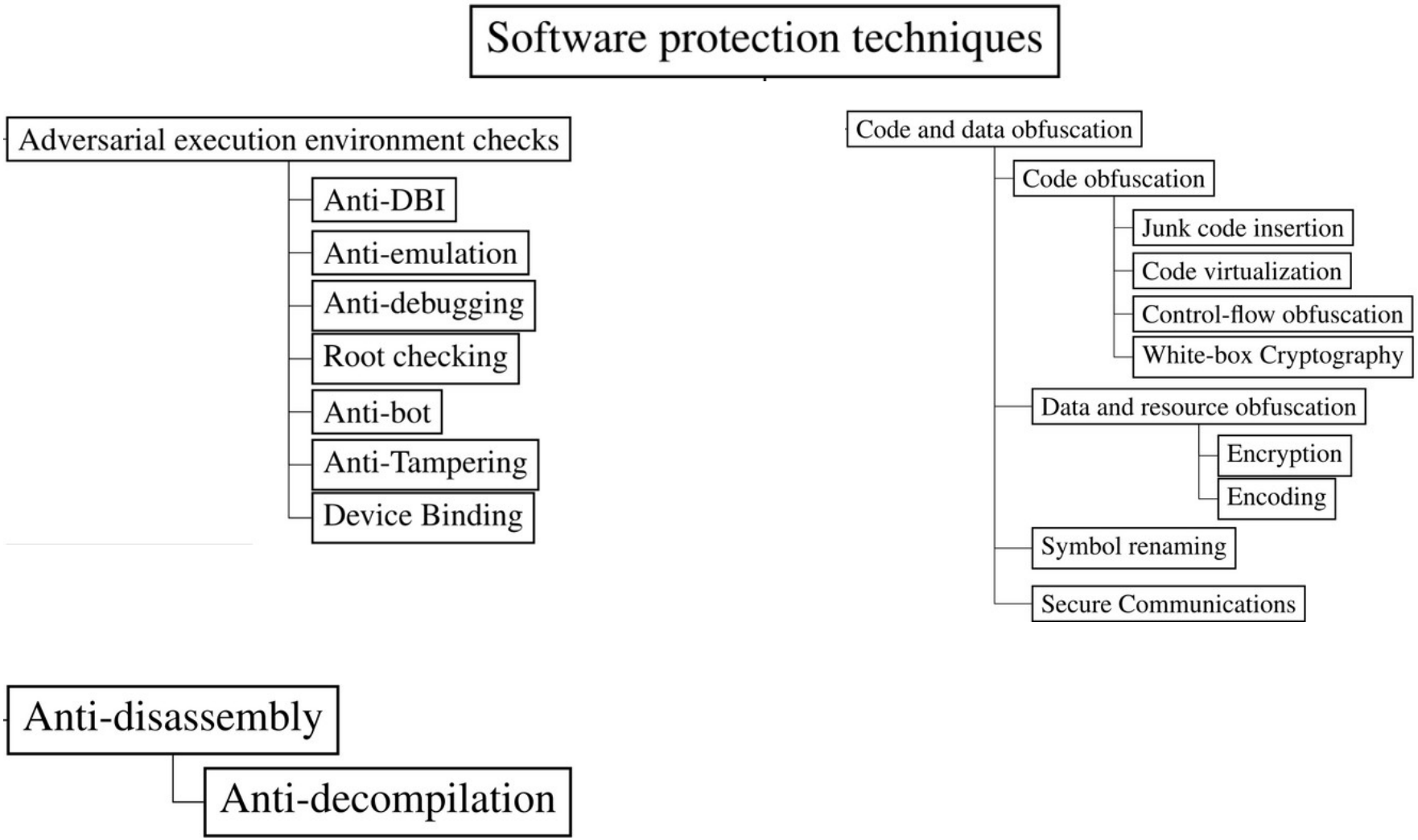
Protection Techniques



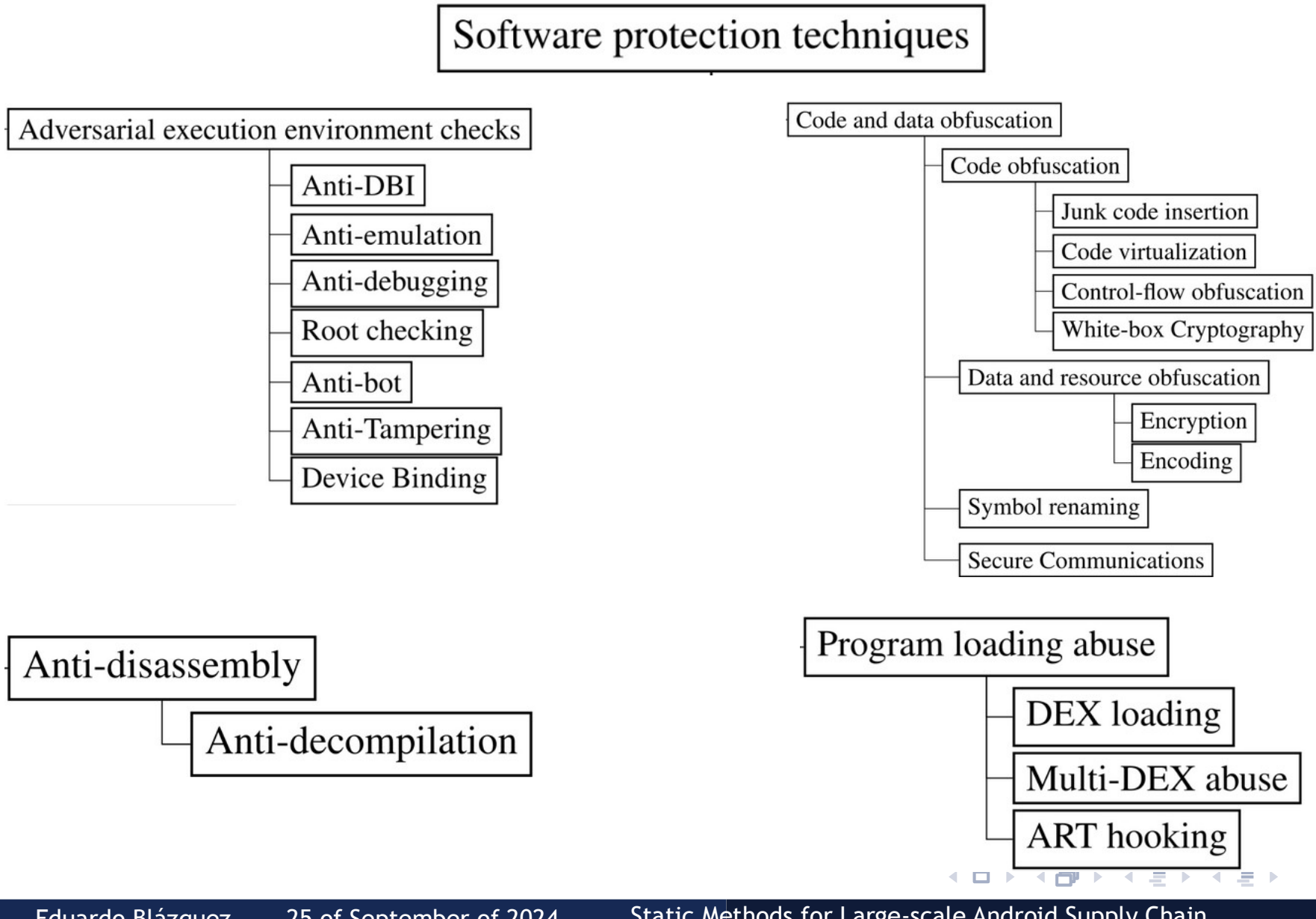
Protection Techniques



Protection Techniques



Protection Techniques



Android Code Protectors

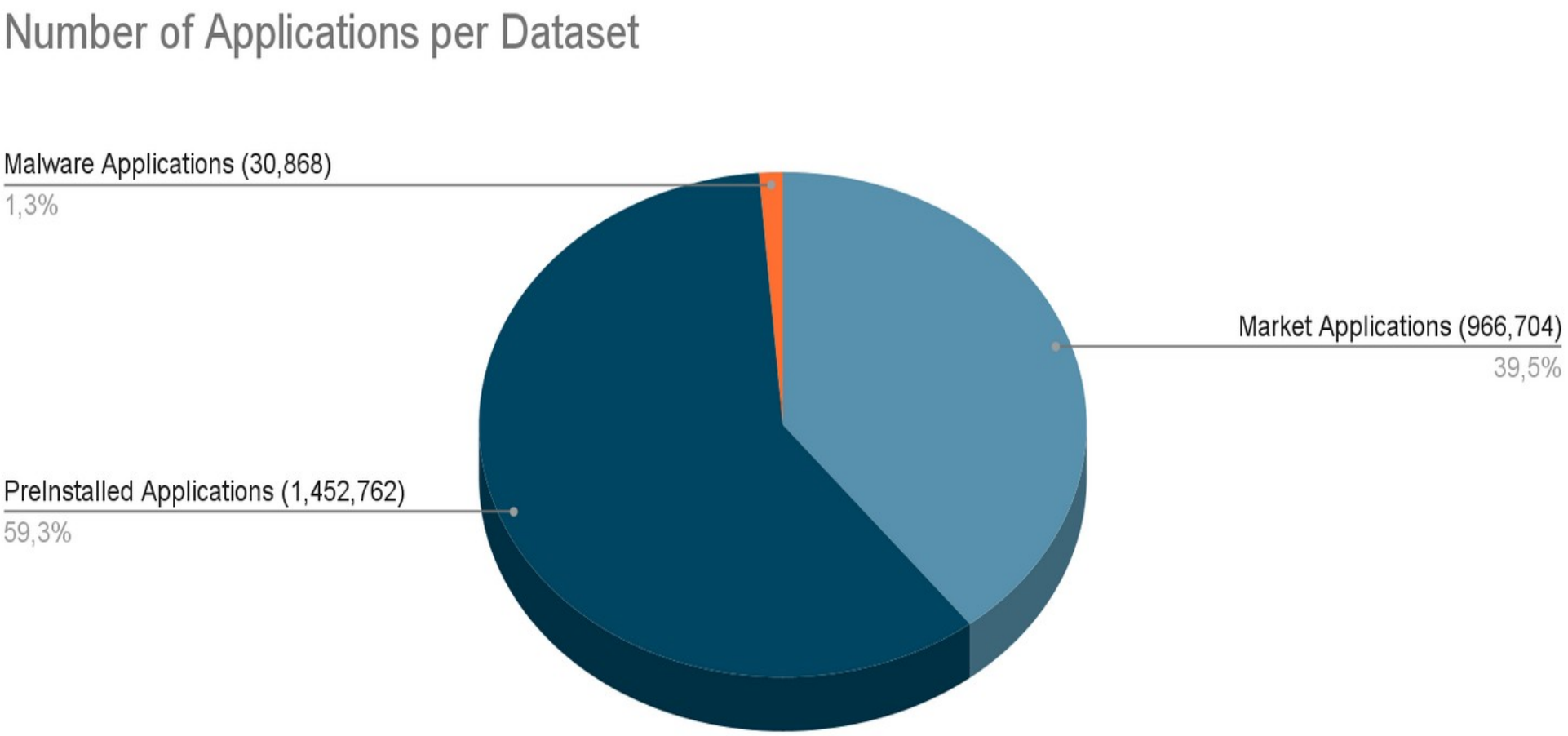
- We analyzed different solutions used to protect Android applications.
- For the analysis we used APKiD (Android version of PEiD), a tool for detecting “packers” through signatures (string or byte patterns).
- We identified 28 solutions divided in 3 categories:
  - Packers (16): compress/encrypt the code and load them in memory during run-time.
  - Obfuscators (7): modify program’s code in order to make it harder to understand.
  - Protectors (5): different techniques are applied to protect the code (anti-emulation, anti-debugging, root checks, etc).

Android Software Protection in the Wild

Research questions:

- ➔ RQ1. How prevalent are different protection techniques globally in Android applications?
- ➔ RQ2. Is a protection more commonly found in certain categories of applications?
- ➔ RQ3. What protection techniques are typically used by Android malware?
- ➔ RQ4. How has the use of Android software protection evolved over time?

Dataset


Number of Applications per Dataset
Malware Applications (30,868)
1,3%
Market Applications (966,704)
39,5%
PreInstalled Applications (1,452,762)
59,3%

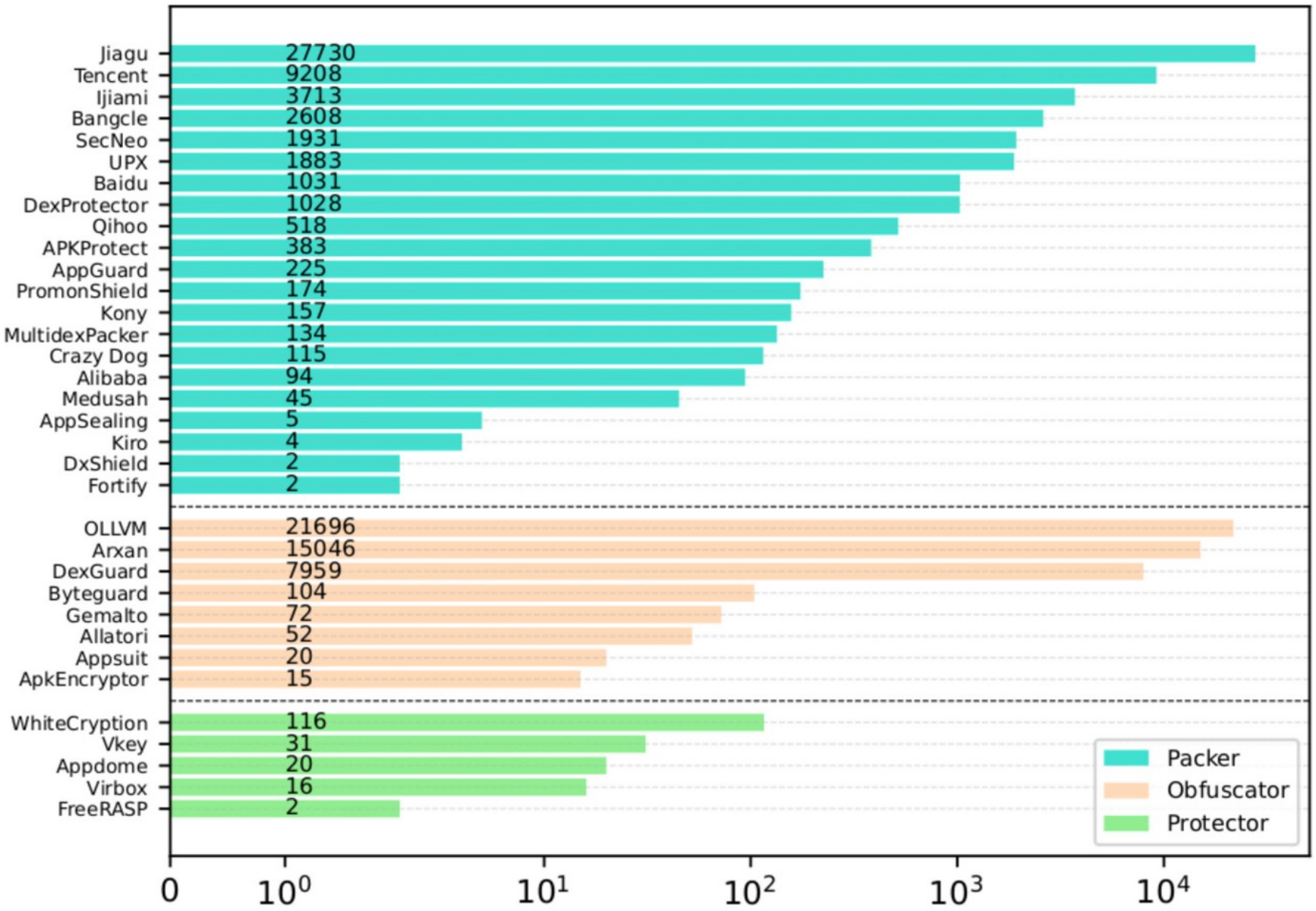
RQ1: How prevalent are different protection techniques globally in Android applications?

Source	Packer	Obfuscator	Protector
Huawei AppGallery [178]	43.17%	20.97%	0.32%
Qihoo 360	40.32%	21.9%	0.51%
MI [222]	33.93%	12.97%	0.06%
Baidu [82]	25.19%	56.92%	0.5%
Anzhi	18.2%	4.1%	0.01%
Tencent [293]	13.93%	5.1%	0.01%
App China [70]	6.2%	3.86%	0.01%
Google Play Store [164]	0.59%	2.65%	0.0%
APKMonk [67]	0.48%	4.12%	0.02%
Torrents	0.4%	0.0%	0.0%
HiAPK [170]	0.38%	0.0%	0.0%
APKMirror [66]	0.18%	5.95%	0.06%
1Mobile	0.11%	0.05%	0.0%
Angeeks	0.07%	0.0%	0.0%
Slideme [281]	0.0%	0.01%	0.0%
ApkBang	0.0%	0.0%	0.0%
F-Droid [141]	0.0%	0.0%	0.0%
Freewarelovers [150]	0.0%	0.0%	0.0%
ProAndroid Apps [260]	0.0%	0.0%	0.0%
Preinstalled [381]	0.03%	0.25%	0.0%
vxUnderground [313] + VirusShare [310] + Argus Collection [77]	9.63%	3.83%	0.01%
PRAGuard [405]	0.0%	14.87%	0.99%
Malgenome [459]	0.0%	0.0%	0.0%
Total	50,664	45,320	185

RQ1: How prevalent are different protection techniques globally in Android applications?

Source	Packer	Obfuscator	Protector
Huawei AppGallery [178]	43.17%	20.97%	0.32%
Qihoo 360	40.32%	21.9%	0.51%
MI [222]	33.93%	12.97%	0.06%
Baidu [82]	25.19%	56.92%	0.5%
Anzhi	18.2%	4.1%	0.01%
Tencent [293]	13.93%	5.1%	0.01%
App China [70]	6.2%	3.86%	0.01%
Google Play Store [164]	0.59%	2.65%	0.0%
APKMonk [67]	0.48%	4.12%	0.02%
Torrents	0.4%	0.0%	0.0%
HiAPK [170]	0.38%	0.0%	0.0%
APKMirror [66]	0.18%	5.95%	0.06%
1Mobile	0.11%	0.05%	0.0%
Angeeks	0.07%	0.0%	0.0%
Slideme [281]	0.0%	0.01%	0.0%
ApkBang	0.0%	0.0%	0.0%
F-Droid [141]	0.0%	0.0%	0.0%
Freewarelovers [150]	0.0%	0.0%	0.0%
ProAndroid Apps [260]	0.0%	0.0%	0.0%
Preinstalled [381]	0.03%	0.25%	0.0%
vxUnderground [313] + VirusShare [310] + Argus Collection [77]	9.63%	3.83%	0.01%
PRAGuard [405]	0.0%	14.87%	0.99%
Malgenome [459]	0.0%	0.0%	0.0%
Total	50,664	45,320	185

RQ1: How prevalent are different protection techniques globally in Android applications?



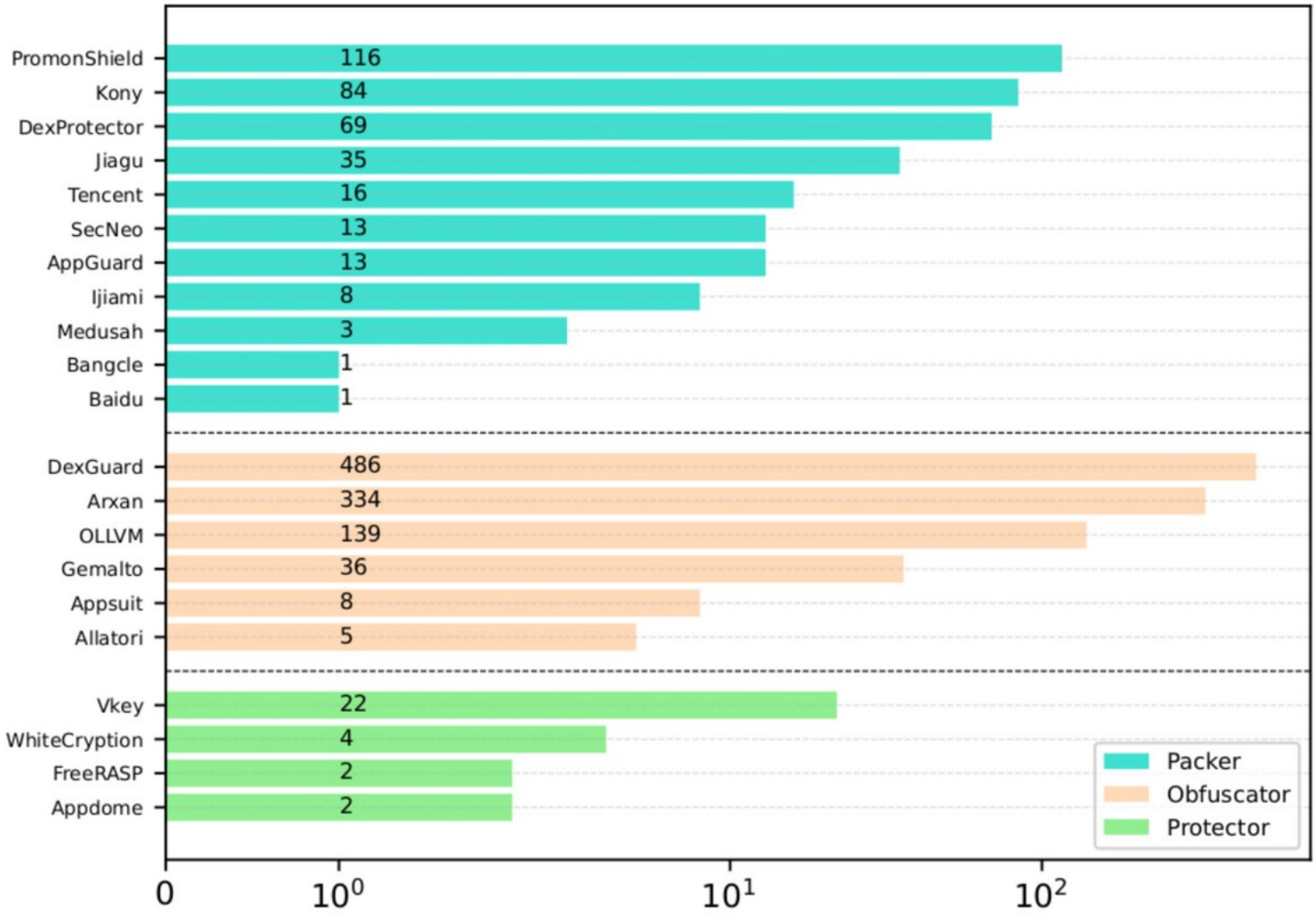
RQ2: Is a protection more commonly found in certain categories of applications?

Play Store GenreID	Packer	Obfuscator	Protector
ART_AND_DESIGN	0.13%	0.52%	0.0%
AUTO_AND_VEHICLES	0.85%	1.1%	0.0%
BEAUTY	0.12%	0.75%	0.0%
BOOKS_AND_REFERENCE	0.31%	0.86%	0.03%
BUSINESS	0.67%	0.88%	0.0%
COMICS	2.63%	1.49%	0.0%
COMMUNICATION	0.44%	2.65%	0.0%
DATING	0.35%	3.99%	0.0%
EDUCATION	0.23%	0.5%	0.0%
ENTERTAINMENT	0.59%	3.57%	0.02%
EVENTS	0.0%	0.0%	0.0%
FINANCE	2.52%	7.07%	0.21%
FOOD_AND_DRINK	0.16%	1.01%	0.0%
GAMES	0.51%	5.71%	0.0%
HEALTH_AND_FITNESS	0.76%	1.11%	0.02%
HOUSE_AND_HOME	0.39%	0.77%	0.0%
LIBRARIES_AND_DEMO	0.85%	0.51%	0.0%
LIFESTYLE	0.64%	1.21%	0.0%
MAPS_AND_NAVIGATION	0.65%	1.32%	0.0%
MEDICAL	0.17%	0.48%	0.0%
MUSIC_AND_AUDIO	0.49%	3.19%	0.0%
NEWS_AND_MAGAZINES	0.12%	1.66%	0.0%
PARENTING	0.45%	1.39%	0.0%
PERSONALIZATION	0.11%	2.44%	0.0%
PHOTOGRAPHY	1.55%	2.26%	0.0%
PRODUCTIVITY	0.74%	2.57%	0.05%
SHOPPING	0.77%	3.03%	0.0%
SOCIAL	0.26%	3.63%	0.06%
SPORTS	0.18%	1.44%	0.0%
TOOLS	1.57%	2.43%	0.01%
TRAVEL_AND_LOCAL	0.7%	3.06%	0.01%
VIDEO_PLAYERS	1.35%	3.28%	0.0%
WEATHER	0.61%	1.5%	0.0%

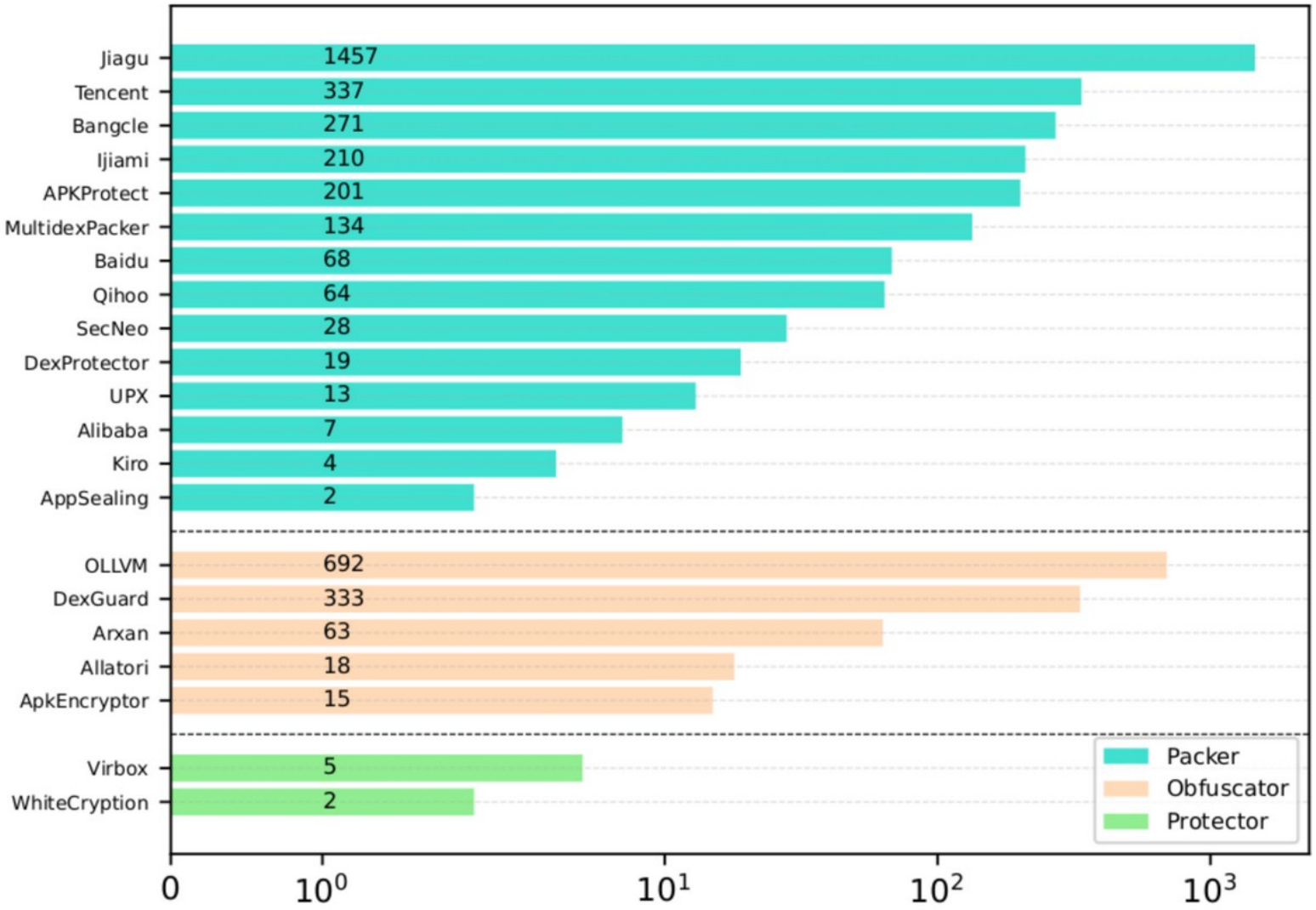
RQ2: Is a protection more commonly found in certain categories of applications?

Play Store GenreID	Packer	Obfuscator	Protector
ART_AND_DESIGN	0.13%	0.52%	0.0%
AUTO_AND_VEHICLES	0.85%	1.1%	0.0%
BEAUTY	0.12%	0.75%	0.0%
BOOKS_AND_REFERENCE	0.31%	0.86%	0.03%
BUSINESS	0.67%	0.88%	0.0%
COMICS	2.63%	1.49%	0.0%
COMMUNICATION	0.44%	2.65%	0.0%
DATING	0.35%	3.99%	0.0%
EDUCATION	0.23%	0.5%	0.0%
ENTERTAINMENT	0.59%	3.57%	0.02%
EVENTS	0.0%	0.0%	0.0%
FINANCE	2.52%	7.07%	0.21%
FOOD_AND_DRINK	0.16%	1.01%	0.0%
GAMES	0.51%	5.71%	0.0%
HEALTH_AND_FITNESS	0.76%	1.11%	0.02%
HOUSE_AND_HOME	0.39%	0.77%	0.0%
LIBRARIES_AND_DEMO	0.85%	0.51%	0.0%
LIFESTYLE	0.64%	1.21%	0.0%
MAPS_AND_NAVIGATION	0.65%	1.32%	0.0%
MEDICAL	0.17%	0.48%	0.0%
MUSIC_AND_AUDIO	0.49%	3.19%	0.0%
NEWS_AND_MAGAZINES	0.12%	1.66%	0.0%
PARENTING	0.45%	1.39%	0.0%
PERSONALIZATION	0.11%	2.44%	0.0%
PHOTOGRAPHY	1.55%	2.26%	0.0%
PRODUCTIVITY	0.74%	2.57%	0.05%
SHOPPING	0.77%	3.03%	0.0%
SOCIAL	0.26%	3.63%	0.06%
SPORTS	0.18%	1.44%	0.0%
TOOLS	1.57%	2.43%	0.01%
TRAVEL_AND_LOCAL	0.7%	3.06%	0.01%
VIDEO_PLAYERS	1.35%	3.28%	0.0%
WEATHER	0.61%	1.5%	0.0%

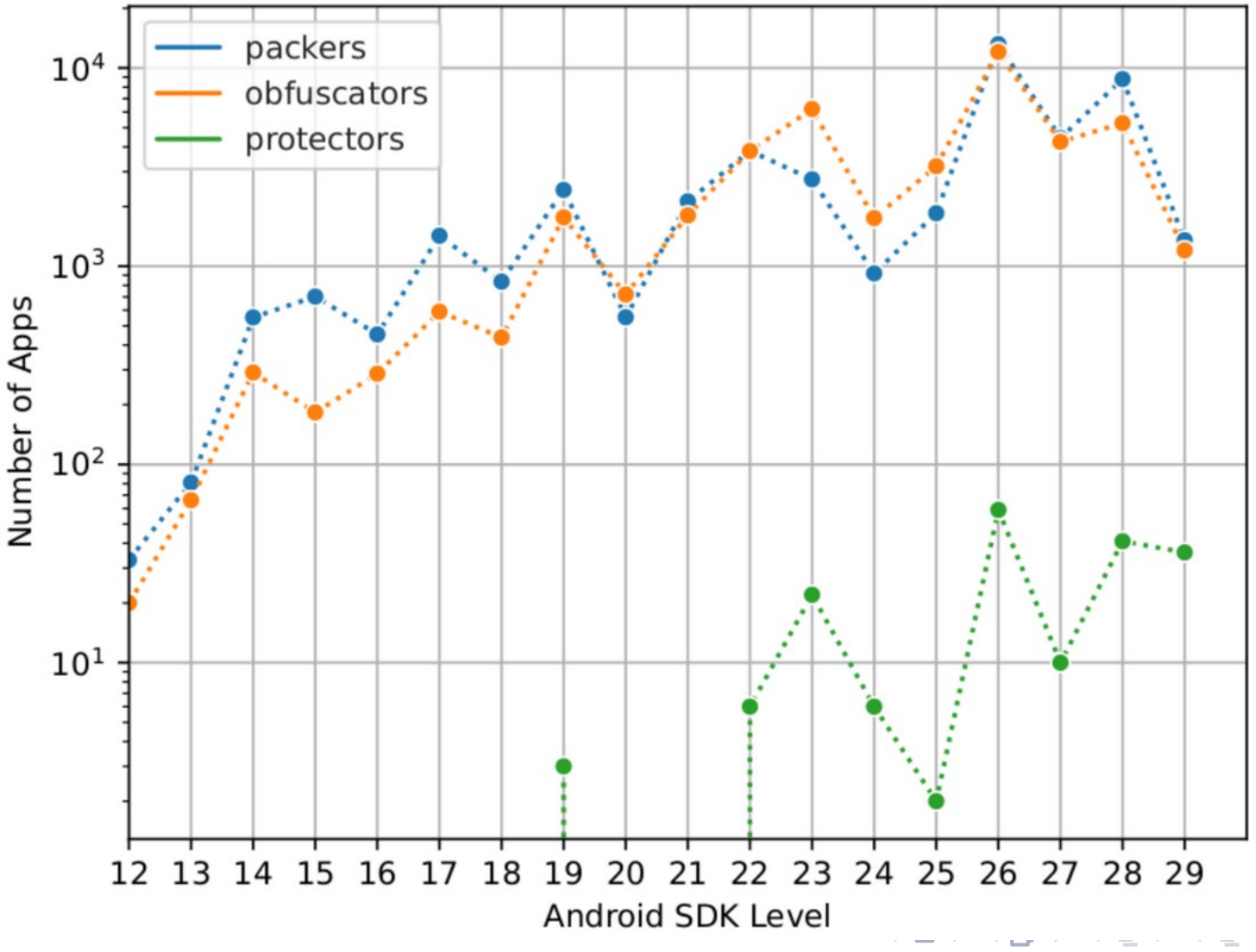
Most common protections in Finance category:



Most common protections in Malware:



Longitudinal analysis of software protection



Remarks

- Modern applications store increasingly sensitive information, making it crucial for developers to prioritize code protection.
- Detecting software protection tools remains challenging, but APKiD offers a best-effort approach using byte and string pattern analysis.
- While software protectors have low overall adoption, our analysis shows a growing use of packers and obfuscators.
- Certain app categories, such as Finance apps on Google Play and apps from Chinese markets, show greater concern for implementing security measures.

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

Published Research

- "Trouble Over-The-Air: An Analysis of FOTA Apps in the Android Ecosystem." E. Blázquez, S. Pastrana, Álvaro Feal, Julien Gamba, Platon Kotzias, N. Vallina-Rodriguez, J. Tapiador.
  - Published in: 2021 IEEE Symposium on Security and Privacy

- "Kunai: A static analysis framework for Android apps." Eduardo Blázquez and Juan Tapiador.
  - Published in: SoftwareX

- "Practical Android Software Protection In The Wild" Eduardo Blázquez and Juan Tapiador
  - Submitted

Other Contributions

- Other papers:
  - “Mules and Permission Laundering in Android: Dissecting Custom Permissions in the Wild” Julien Gamba, Alvaro Feal, **Eduardo Blazquez**, Vinuri Bandara, Abbas Razaghpanah, Juan Tapiador, and Narseo Vallina-Rodriguez
    - Published in: IEEE Transactions on Dependable and Secure Computing
- Books:
  - “Fuzzing Against the Machine: Automate vulnerability research with emulated IoT devices on QEMU” Antonio Nappa, Eduardo Blazquez
    - Book published by: Packt Publishing
- Other Conferences:
  - “Using MLIR for Dalvik Bytecode Analysis”
    - EuroLLVM 2023

Published Tools

- FOTA Finder - tool for discovering FOTA signals in Android applications.
- DEXtripador - plugin for FOTA Finder to extract DEX files from optimized ODEX files.
- OTA Hunter - tool for discovering installation signals on Android applications.
- KUNAI - library for static analysis of DEX files and APK files.

- Introduction
  - Key Takeaways
  - Motivation
- Trouble Over-The-Air: An Analysis of FOTA Applications in the Android Ecosystem
- Fantastic Installers and How to Find Them: An Analysis of Installer Applications.
- Kunai: A Static Analysis Framework for Android App
- Practical Android Software Protection In The WIld
- Final Remarks
  - Published Research
  - Conclusions

Conclusions

- An analysis of 2,013 FOTA apps revealed a fragmented ecosystem with multiple first- and third-party actors, leading to privacy issues and potentially harmful behaviors. We also found that FOTA apps can install non-system and potentially unwanted apps.
- We observed that app installations come not only from first-party code but also from third-party SDKs. The case of “Digital Turbine” shows that these SDKs can be used to install potentially unwanted apps.
- We created an efficient analysis library suitable for large-scale analysis, with a design that supports extensibility for new file formats.
- While security protections for Android apps are increasing, adoption remains low overall, except for higher usage in Chinese markets and financial apps like Google Pay.

Thank you very much

Time for Q&A