Android-Device-Security.org

Towards a Transparent Database of Android Device Security Attributes

Speakers: Daniel R. Thomas, Alastair R. Beresford, René Mayrhofer

Team

Institute of Networks and Security,
Johannes Kepler University Linz (Austria)

- René Mayrhofer*
- Michael Roland
- Dominik Dirmeier
- Katrin Kern
- Tobias Höller

Department of Computer Science and Technology,
University of Cambridge (UK)

- Alastair R. Beresford
- Stan Zhang

Computer and Information Sciences,
University of Strathclyde (UK)

- Daniel R. Thomas

Including analysis and code from Billy Lau and others at Google

[* also affiliated with Google]

Motivation: Reveal security state of Android devices

Want to give *meaningful* data to users and organizations to make an informed decision concerning the security of a particular device

Provide an incentive for investing in improved security

Measuring Android security in 2015

- Measured *vulnerability* of devices in use to known critical vulnerabilities
- Produced a score out of 10
- Scores were predictably bad

- Only vulnerability measured; not all aspects of security
- Should now be in a better place
- Longitudinal work in progress

Daniel R. Thomas, Alastair R. Beresford, Andrew Rice (2015). Security metrics for the Android ecosystem. ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM). http://androidvulnerabilities.org/

Device Analyzer

App deployed 2011-2019: 30,000 contributors

Gathered wide variety of data including system statistics

- OS version and build number
- Manufacturer and device model
- Network operators

Combine running software version with known vulnerabilities

Daniel T. Wagner, Andrew Rice and Alastair R. Beresford. Device Analyzer: Understanding smartphone usage. International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (Mobiquitous) 2013

Classifying vulnerability of devices

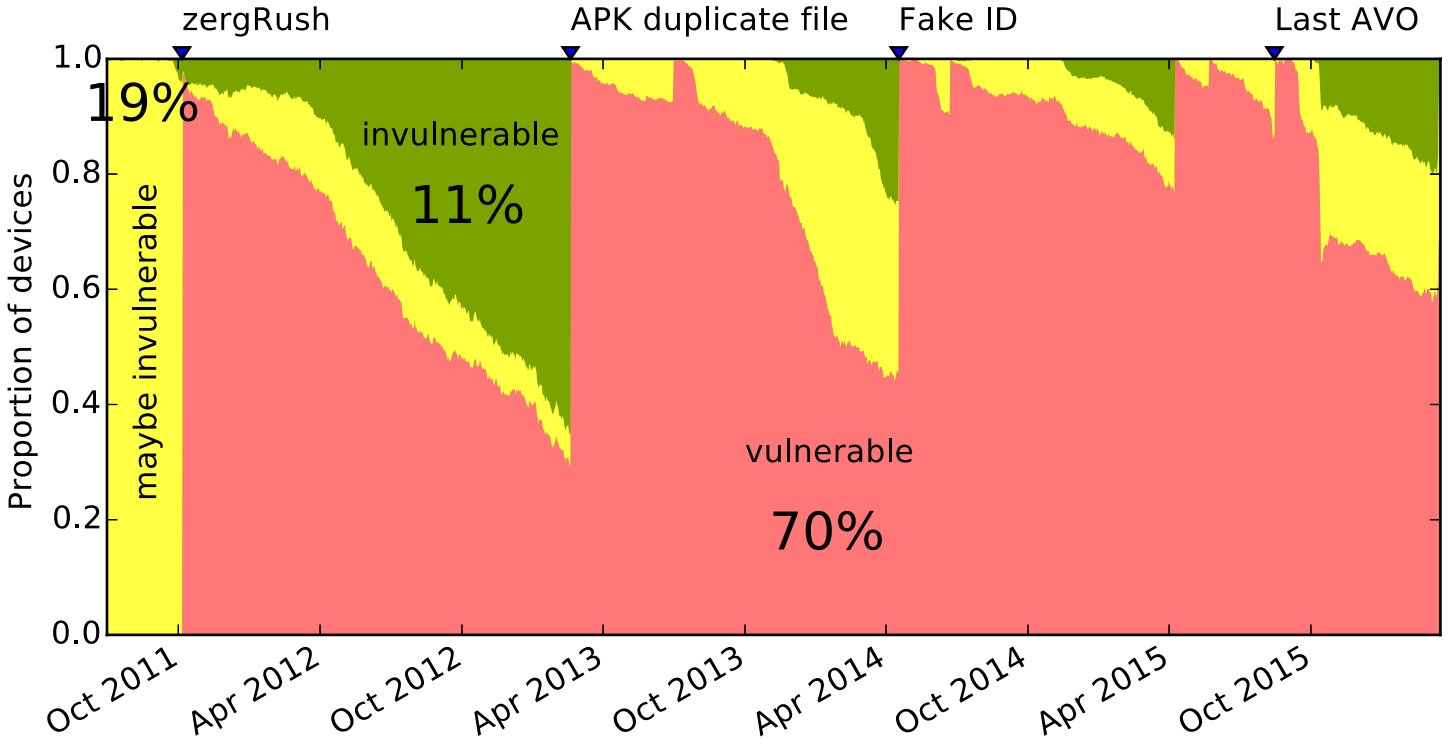
Each day for each device:

Vulnerable: Known critical vulnerabilities

Maybe invulnerable: May have a backported fix, insufficient data

Invulnerable: No known critical vulnerabilities

Many Android vulnerabilities between 2011 & 2015



FUM Score: Vulnerability of Android devices

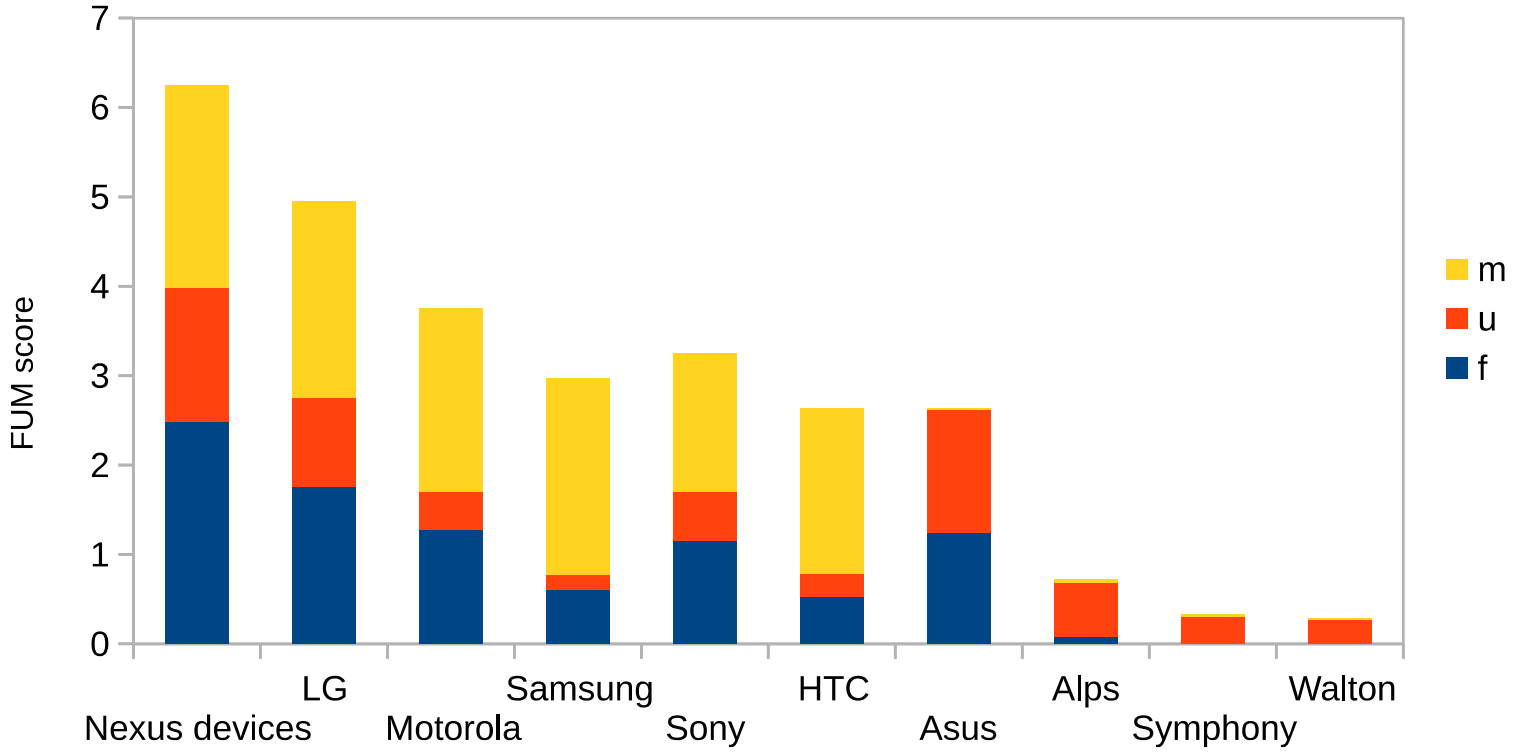
$$\mathbf{FUM} = 4f + 3u + 3\frac{2}{1+e^{m}}$$

$f$ree from known vulnerabilities

$u$pdated to the latest version

$m$ean unfixed vulnerabilities

Comparing manufacturers in 2015



Current area of focus: pre-installed apps

Android users' security and privacy at risk from shadowy ecosystem of pre-installed software, study warns

Natasha Lomas @riptari / 1:36 pm EDT • March 25, 2019

Comment

An open letter to Google

Privacy International and over 50 other organisations have submitted a letter to Alphabet Inc. CEO Sundar Pichai asking Google to take action against exploitative pre-installed software on Android devices.

Ads are taking over Samsung's Galaxy smartphones — and it needs to stop

288

When you buy a $2,000 smartphone, you shouldn't be the product

Max Weinbach

Jun 30, 2020

1.1K 12K

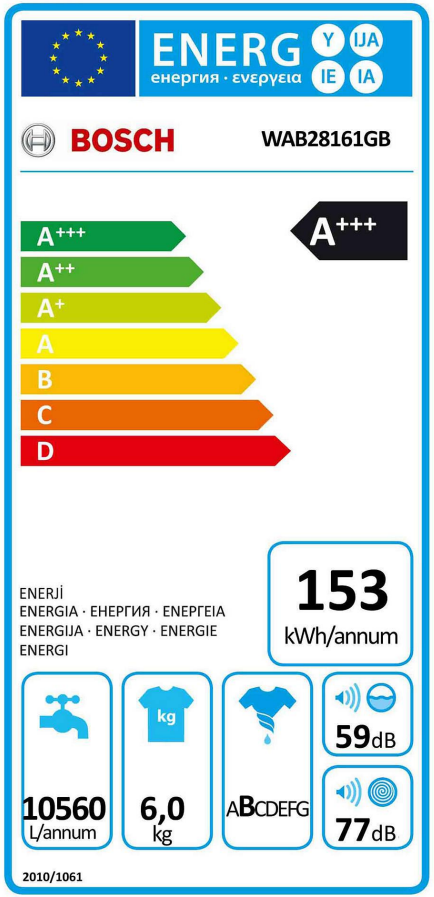
Pre-installed app risk: research objectives

- Increase transparency and accountability
- Develop an app risk into device risk rating
- Overall device risk calculated with respect to baseline
- Create a rating system
- Define risk scoring methodology that is simple to understand; peer reviewed

Scoring difficulties with numbers

Credit: This Is Spinal Tap (1984)

Scoring difficulties with letters



Credit: Bosch

Let’s give it five stars...

Credit: XKCD (CC BY-NC 2.5)

Initially no scores, just the raw data

No value judgement just raw measurements. Revisit if we can say something...

- **Robust**: not easily gamed or sensitive to measurement error
- **Objective**: aligned with measured risk, not personal perspective
- **Meaningful**: a better score implies better security; scores are comparative

Example: we would like a score for ioXt standard

"The ioXt Security Pledge is the result of industry working together to set security standards that bring **security**, **upgradability** and **transparency** to the market and directly into the hands of consumers."

Pre-installed app risk: next steps

Current plan involves looking at three areas of concern:

- Platform signed apps
- Privileged (pre-granted) permissions
- Cleartext traffic

- Publish an open source tool to collect preloaded app risk data
- Write a paper with a rigorous analysis
- Results used to build an ioXt certification profile

Collecting data from different sources

Publicly available data

- OEM documentation & support pages
- Commitments / promises by OEMs
- Device release dates, etc.

Dedicated device security test labs

- 20 devices at JKU Linz
- 10+ devices at Google
- (potentially future devices at Cambridge)
- Source code for data collection will be released soon

Device test lab v0.9

Currently 20 devices

- Low-end to high-end
- Different OEMs: Google, Huawei, Oppo, Nokia, Samsung, Sony, Xiaomi
- Tried to select devices with either noticeable **market share** in Europe **or** otherwise **interesting properties**
- Generally Android 9 or 10 (1 exception)
- Generally 2017 or younger (2 exceptions because of market share)
- **Querying security attributes through ADB and on-device** (permissionless test APK)
- Possible to **capture Wi-Fi traffic per device** (including multicast and MAC randomization)

Collecting data from different sources

Future: crowd-sourced data

- Collected by dedicated app from devices in the field
- App intends to give recommendations on security state/settings
- Allows to view comparison with other devices
- Collects anonymous security attributes about devices (e.g. patch dates as seen in the field)
- Currently under development

Examples of security attributes collected

- Average patch frequency [days]
- **Guaranteed patch availability** [years]
- **Latest security patch level** [date]
- **Latest Android release** [API number]
- **Multi-user support** [boolean]
- **Seamless updates** [boolean]
- **Device encryption type** ["file" or "block"]
- Preloaded apps with system privileges [count]
- Software mitigations: kernel / userspace CFI/SCS, integer overflow sanitization enabled, etc.
- Biometric sensors false accept/reject rates spoof/impostor accept rates, etc.

https://www.android-device-security.org/attributes/

Call to action 1: Talk to us

- Which other attributes would you like to see? Please tell us!
- Are there particular devices that should be measured (e.g. because they are used in specific important sub-fields)?
- Aggregated data will be public, raw data available under NDA for researchers
  → Do you have an idea on deriving **robust**, **meaningful**, and **objective** features from this (and other) data?

https://www.android-device-security.org/attributes/

Call to action 2: Improve the ecosystem together

- First goal: allow users to make informed decisions
- Second goal: strengthen the Android ecosystem security posture further
- Let’s do this together by joining efforts. We are open to including other interesting data to foster transparency and motivate stakeholders to improve.

https://www.android-device-security.org/