SMD

Introduction

Lecture 1

Security of Mobile Devices

2022

Team, Schedule and Grading

Android Architecture

Application Development Overview

Security Mechanisms

Bibliography

Outline

Team, Schedule and Grading

Android Architecture

Application Development Overview

Security Mechanisms

Bibliography

Team and Schedule

- Team
  - Laura Ruse, Costin Carabaș, Florin Mihalache, Ionuț Mihalache, Cosmin Chenaru, invited speakers
- Schedule
  - Lecture: Wednesday 8-10, on MS Teams
  - Labs: Thursday, 8-10, 18-20, 20-22, on MS Teams

Syllabus

- Android OS:
  - SDK
  - Internals
  - Security architecture
  - Network security
  - Secure booting, system updates and root access
  - Vulnerabilities and malware
- Invited speakers from industry

Resources

- Wiki: `http://ocw.cs.pub.ro/courses/smd`
  - Courses
  - Labs
  - Class registrer
  - Calendar
- Moodle: `http://curs.upb.ro/`
  - https://curs.upb.ro/2021/course/view.php?id=5419

Grading

- **0.5 points** Lecture tests and attendance
- **1 point** Lab activity
- **4 points** Assignment
- **1.5 points** Mid-term exam
- **3.5 points** Final exam
- 50% (2.5 points) from the lab activity and the assignment are required to enter the exam.
- 5p are required to pass the class.

Grading

- **0.5 points** Lecture tests and attendance
  - the test will be held at the end of the lecture, on Moodle
  - the test will consist of one simple question
  - the question will be related to what was presented at the course

Grading

- **1 points** Lab activity
  - Android Studio, Java, Kotlin (if you want to)
  - Lab allocation
  - Github Classroom
  - The lab will be solved during the lab (it may be finished after the lab)
  - Submit until Sunday 23:55 (same week)

Grading

- **4 points** Assignment/project
  - `https://ocw.cs.pub.ro/courses/smd/res/assignment`
  - Project theme registration - April 11th, 0.3p penalty
  - Intermediary project presentation - May 19th, 0.5p penalty
  - Final project presentation - June 2nd

Grading

- **1.5 points** Mid-term exam (first 4 lectures)
- **3.5 points** Final exam (next 4 lectures)
- 20 multiple choice questions
- 20 minutes
- each question has 4 choices of which only one is correct
- correct answer - 1 point
- incorrect/no answer - 0 points

Grading

- For those who retake the course, all the forms of examination except the final exam will be equivalated/scaled

Recommended Bibliography

- Embedded Android: Porting, Extending, and Customizing, Karim Yaghmour, 2015
- Android Security Internals, Nicolay Elenkov, 2015
- Android Hacker's Handbook, Joshua J. Drake, 2014
- Introducere in sistemul de operare Android - Laura Ruse, Vlad Traistă-Popescu, 2021
- Securitatea sistemului de operare Android - Laura Ruse, Vlad Traistă-Popescu, 2021
- `http://developer.android.com`

Outline

Team, Schedule and Grading

Android Architecture

Application Development Overview

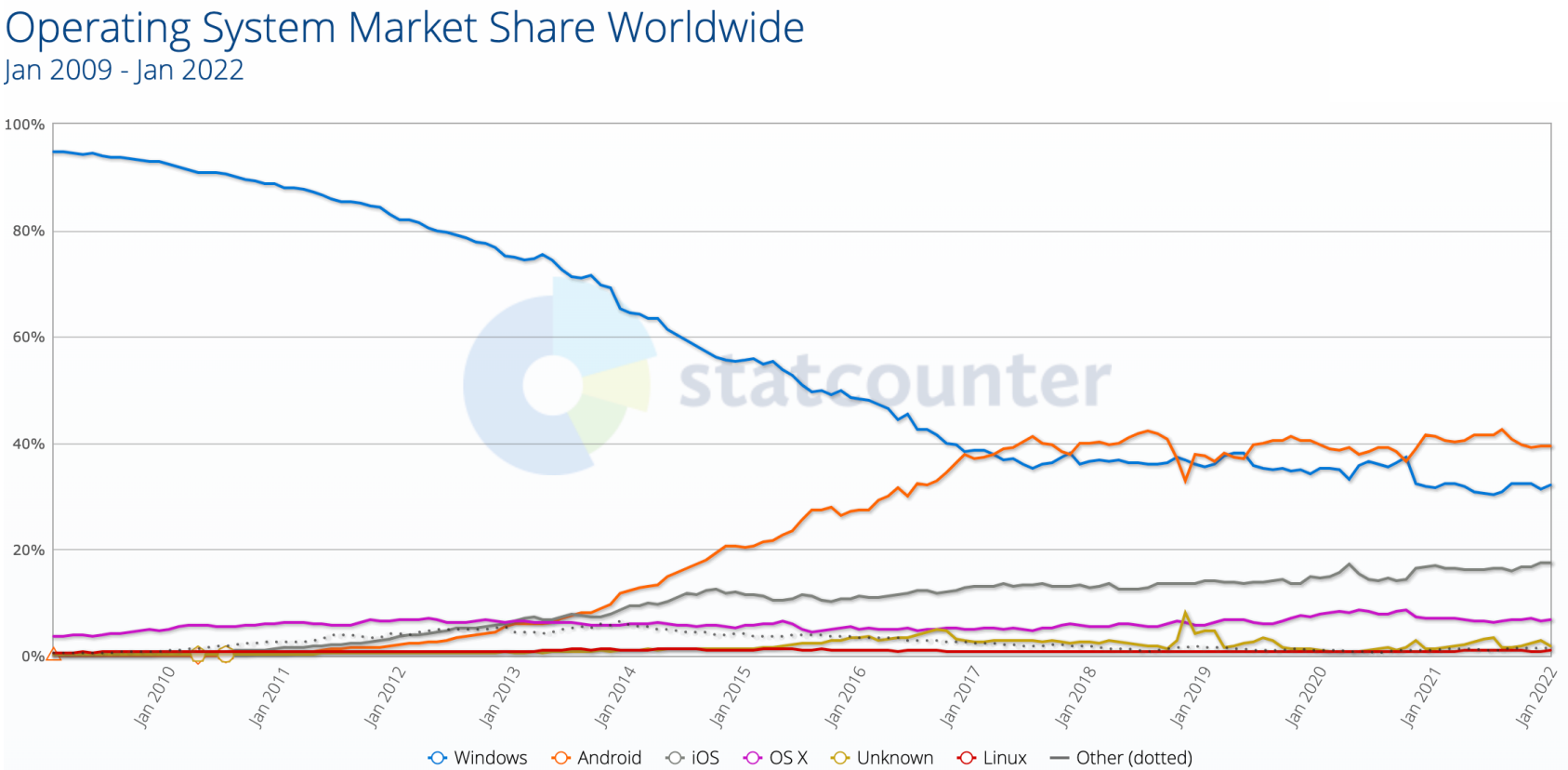
Security Mechanisms

Bibliography

Android popularity

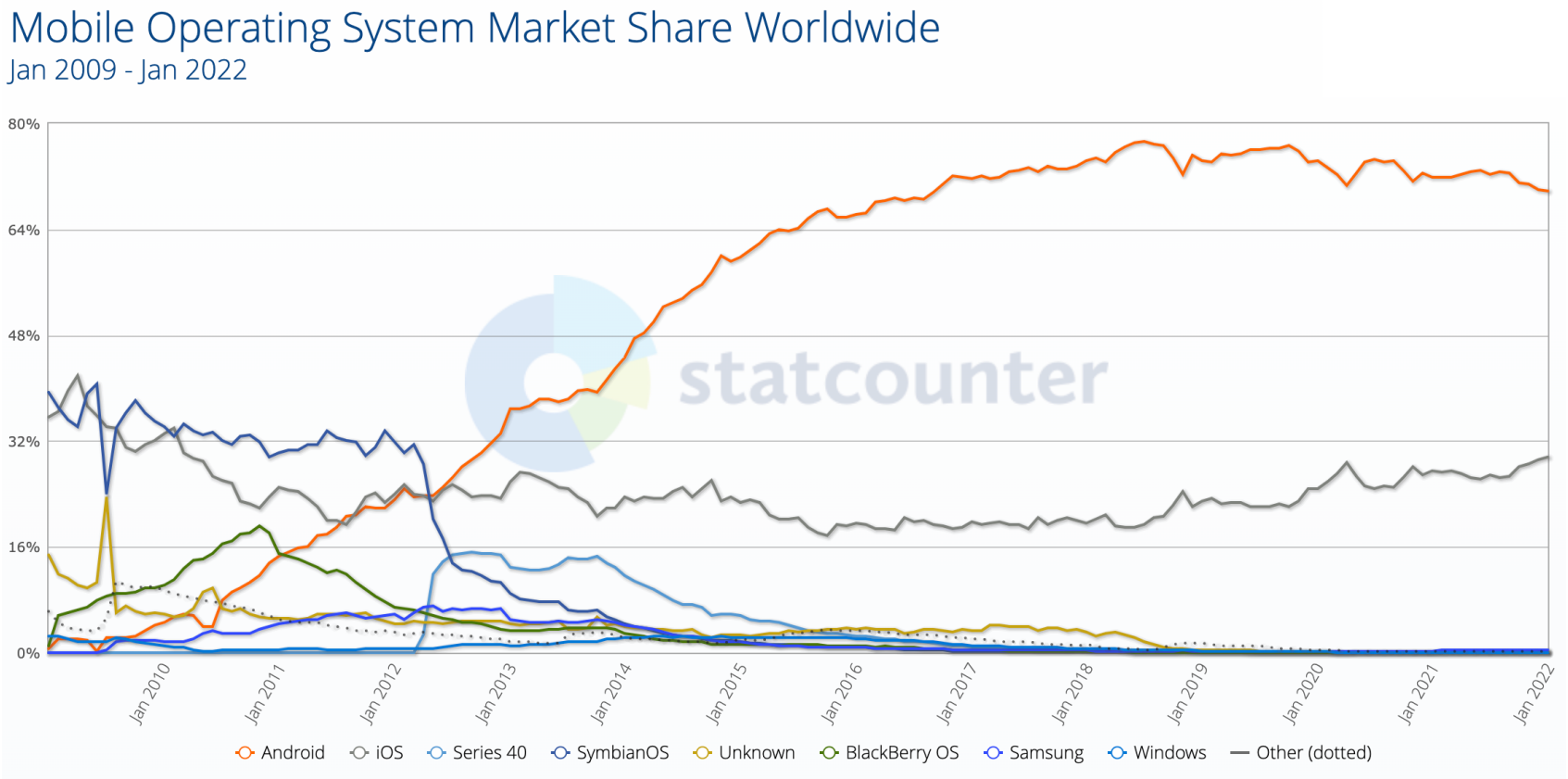
- Open-source OS for mobile devices
- 2.8 billion monthly active users (2021)
- Mobile OS market share (Jan 2022)
  - Android 69.74%
  - iOS 29.49%
- OS market share (across all devices) (Jan 2022)
  - Android 39.45%
  - Windows 32.11%
  - iOS 17.56%
  - OS X 6.74%
  - Linux 0.94%
- Source: Statcounter
- Official application market: Google Play Store

OS Market Share Evolution



Source: Statcounter

Mobile OS Market Share Evolution



Source: Statcounter

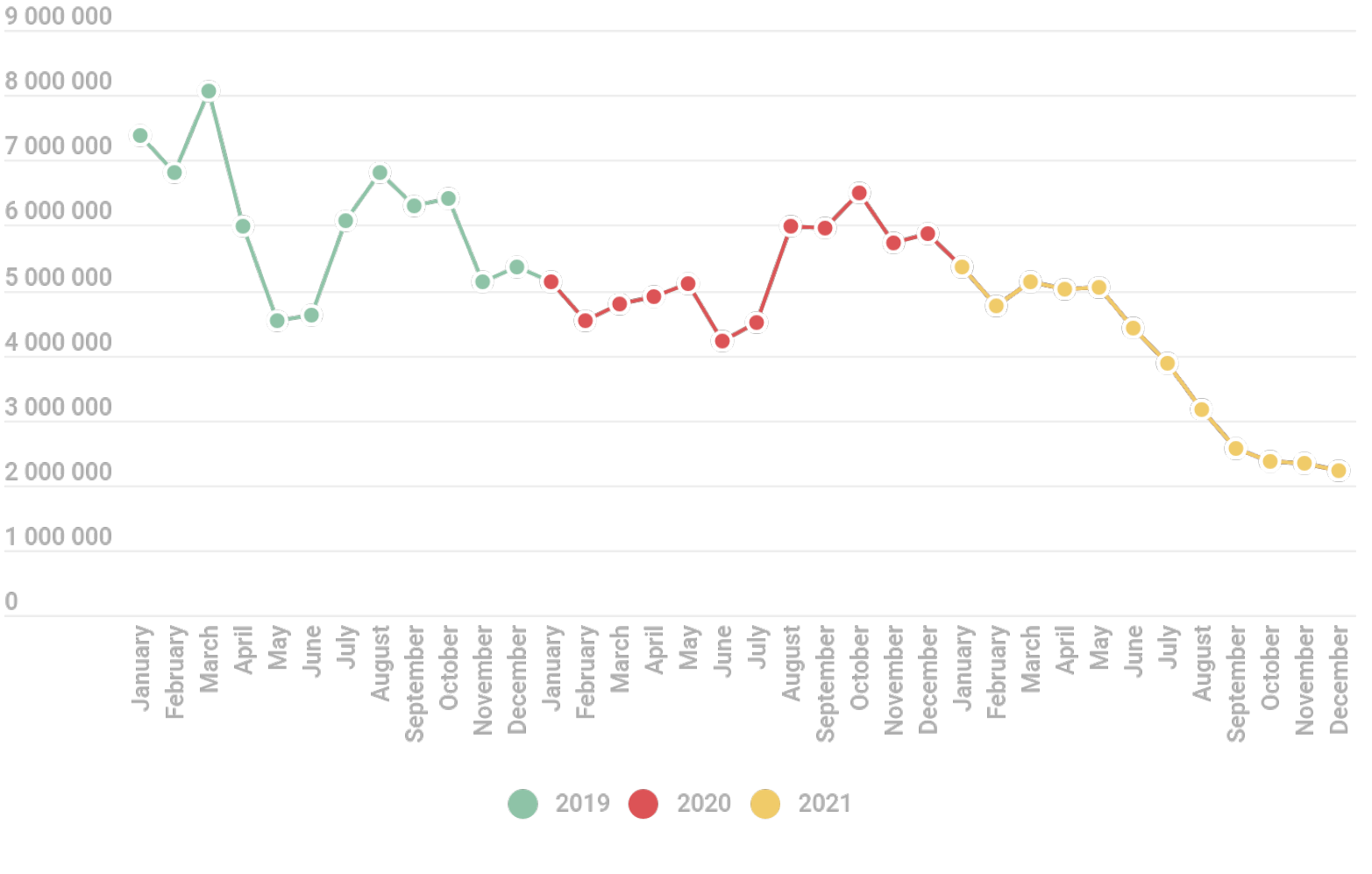
Potentially Harmful Applications (PHA)

- Apps that could put users, user data and devices at risk
- Why the keyword potentially?

Mobile Malware in 2021

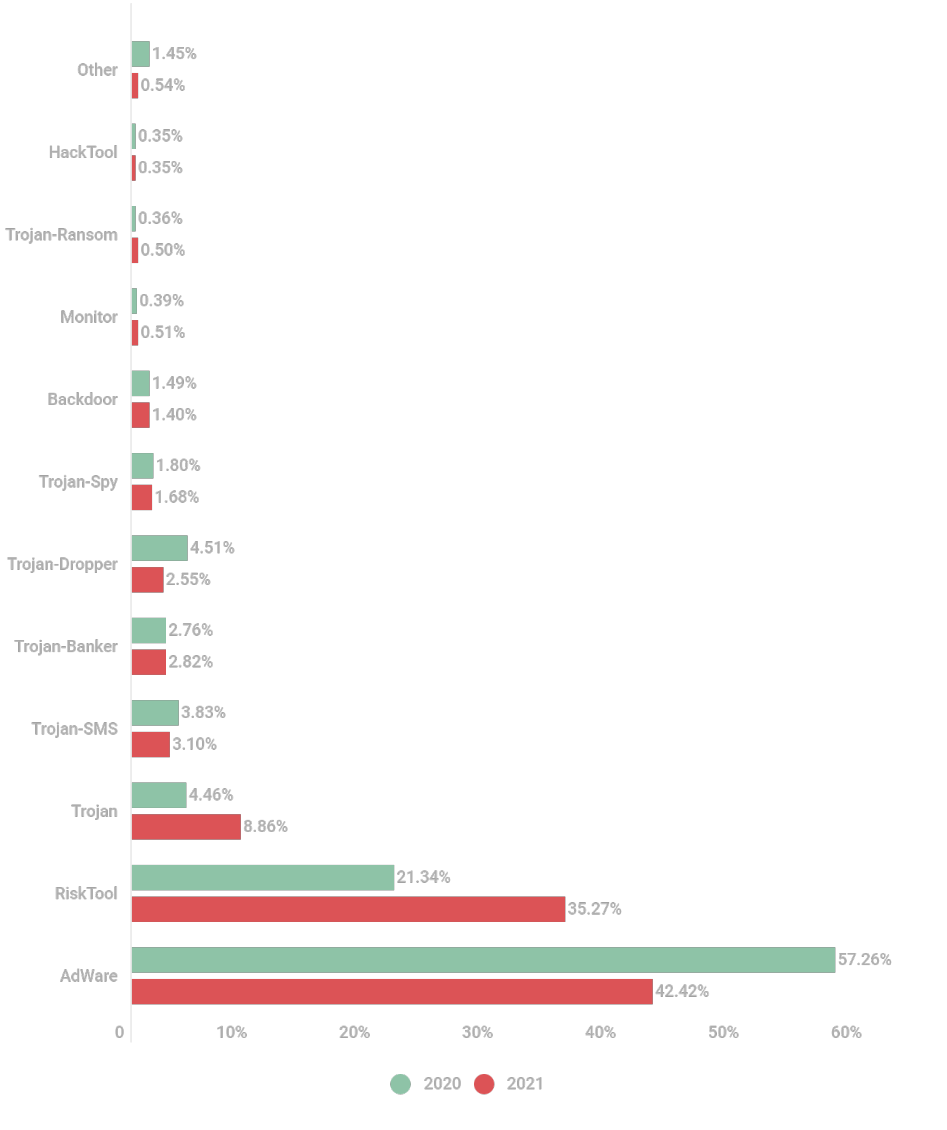
- `https://securelist.com/mobile-malware-evolution-2021/105876/`
- In 2021 Kaspersky detected:
  - 3,464,756 malicious installation packages
  - 97,661 new mobile banking Trojans
  - 17,372 new mobile ransomware Trojans

Attacks on Mobile Users



kaspersky

New Detected Mobile Threats in 2021

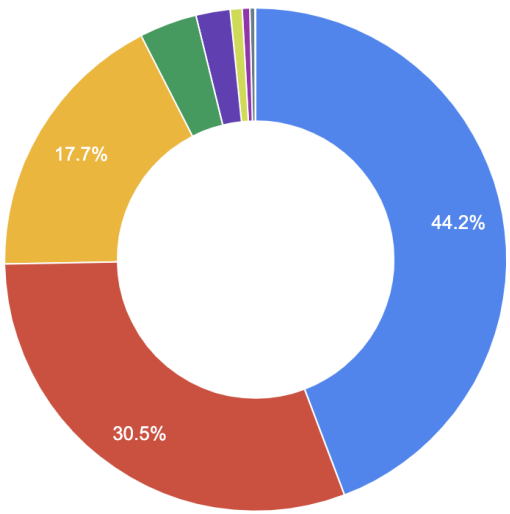


- Google Play Protect - detect and remove PHAs
- Statistics from Google:
- `https://transparencyreport.google.com/android-security/store-app-safety?hl=en`

PHAs detected by Google Play Protect

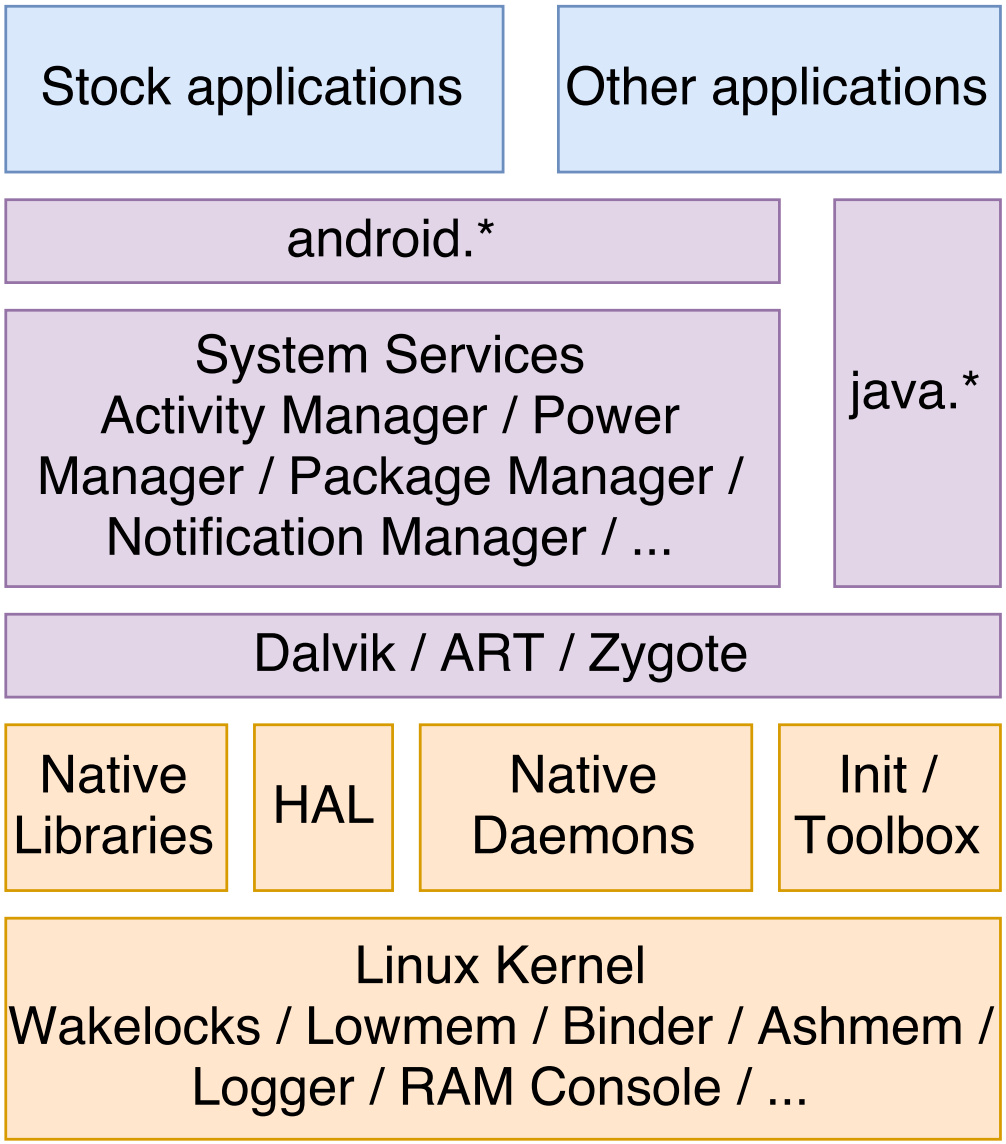
Jul 2021 – Sep 2021

Google Play



Category	PHA Install Rate
Toll fraud	0.014901969%
Spyware	0.0102640646%
Phishing	0.005973268%
Backdoor	0.001250422%
Privilege escalation	0.0007412477%
Trojan	0.0002609596%
Hostile downloader	0.0001614188%
Commercial spyware	0.0001128817%
DOS	0.0000050745%
Spam	0.0000036137%
Rooting	0.000002497%
Windows malware	0.000001482%
SMS fraud	0.0000011766%
Call fraud	0.0000000153%
Ransomware	0.0000000117%

Android Architecture



Android Architecture

- Linux kernel
  - Android Mainlining Project / Android Upstreaming
  - Androidisms
  - Advantages
- Hardware Abstraction Layer (HAL)
  - Standard interfaces
  - Multiple library modules
- Native userspace
  - init process
  - Native daemons
  - Native libraries
    - Through Java framework APIs
    - Through Android NDK

Android Architecture

- Android Runtime
  - Dalvik
  - ART
  - Ahead-Of-Time (AOT) compilation
- Java Runtime libraries
  - `java.*` and `javax.*`
  - Apache Harmony Project
  - Java Native Interface (JNI)

Android Architecture

- System services
  - Fundamental features of Android
  - Native and Java code
  - Service interface
- Android framework libraries
  - Base components for app development
  - Interaction with the hardware
  - Interaction with high level services
  - Framework APIs

Linux Kernel

- Modified to work on mobile devices
- Patches on top of mainline Linux
- Android Mainlining Project / Android Upstreaming
- Wakelocks (also added to Linux 3.5)
- Low-Memory Killer (3.10)
- Binder (3.19)
- Alarm (3.20)
- Logger (3.20)
- Only suspend to memory

Android Runtime - Dalvik VM

- Default until Android 5.0
- Runs Dalvik-specific byte-code
- Dalvik Executable Format (DEX)
  - Runs .dex files instead of .jar files
  - .dex is 50% smaller than corresponding .jar
- Just-In-Time compilation
  - From Android 2.2
  - Short segments of bytecode translated into native machine code at runtime
  - Improves performance

Android Runtime - ART

- From Android 5.0
- More advanced runtime architecture
- Ahead-Of-Time compilation
  - Just once, at installation
  - Entire DEX file -> executable for target device
  - Instead of JIT compilation and Dalvik interpretation
  - More efficient, reduced power consumption
  - More space to store the executables
- Improved memory allocation, GC, debugging and profiling

Native Libraries

- bioniC (libc)
  - Much smaller and faster than glibc
- SQLite
  - Managing SQL databases
- OpenGL ES
  - Standard software interface for 3D processing hardware
- WebKit
  - Display web pages
  - Android, Apple iOS, BlackBerry, Tizen
- SSL
  - Securing the communication over Internet

Application Framework

- System Services and Managers
  - Telephony
  - Location
  - Activity
  - Package
  - Notification
- System Content Providers
  - Calendar
  - Dictionary
  - Contacts
  - Settings

Outline

Team, Schedule and Grading

Android Architecture

Application Development Overview

Security Mechanisms

Bibliography

Application components

- User interaction
  - Activities
- Background functionality
  - Services
  - Broadcast Receivers
  - Content Providers
- Intents

Binder

- Lightweight RPC
- Remote object invocation
- In process and interprocess
- Transmit parcels of data
- Synchronous calls (blocking)

Outline

Team, Schedule and Grading

Android Architecture

Application Development Overview

Security Mechanisms

Bibliography

Sandboxing

- Linux kernel security
  - Isolate user resources (file permissions)
  - Process runs with user's UID/GID (SUID, SGID)
- In Android UIDs are used to identify applications
  - Isolate applications
  - Basis for sandboxing

Sandboxing

- Unique UID assigned to each application at installation time
- Dedicated process running as that UID
- Dedicated directory - only that UID has read/write/execute permissions
- Process-level and file-level sandbox
- Kernel level sandbox - all applications (native and VM)

UIDs

- Well-defined UIDs for system apps and daemons
- Very few daemons under root UID 0
- UIDs for system services start at 1000
- User *system* has UID 1000
  - Special priviledges
- App UIDs start at 10000

Application Files Permissions

- Each app - dedicated data directory
- rwx permissions only for that app UID/GID
- MODE_WORLD_READABLE, MODE_WORLD_WRITEABLE flags
  - Deprecated from Android 4.2

Shared UID

- Apps with the same UID
- Share files
- Run in the same process
- Frequently used by system apps
- Apps signed with the same code singining key
- Deprecated since Android 10

Application Permissions

- Operations outside sandbox
- Declared statically in the Manifest file
- Before Android 6
  - Granted at installation time
  - Cannot be revoked
- From Android 6
  - Granted at runtime
  - Revoked and granted from settings

Enforce Permissions

- Access to lower-level resources
  - Enforced by the Linux kernel
  - Check UID/GID vs resource's owner
- Access to high-level Android components
  - Enforced by Android OS or a certain component

Code Signing Key

- All apps signed by their developer
- Apk signing is based on jar signing
- Same origin policy
  - App updates from the same developer
- Platform keys for signing system apps
  - Shared resources, same process
  - Generated and controlled by the entity that compiled the Android OS

Outline

Team, Schedule and Grading

Android Architecture

Application Development Overview

Security Mechanisms

Bibliography

Bibliography

- https://gs.statcounter.com/os-market-share/mobile/worldwide/#monthly-200901-202201
- https://gs.statcounter.com/os-market-share#monthly-200901-202201
- https://developers.google.com/android/play-protect/potentially-harmful-applications
- https://securelist.com/mobile-malware-evolution-2021/105876/
- https://transparencyreport.google.com/android-security/store-app-safety?hl=en
- Karim Yaghmour, Embedded Android: Porting, Extending, and Customizing, Chapter 2

SMD

Bibliography

- https://developer.android.com/guide/components/activities/intro-activities
- https://developer.android.com/guide/components/services
- https://developer.android.com/guide/components/broadcasts
- https://developer.android.com/guide/topics/providers/content-provider-basics
- https://developer.android.com/guide/components/intents-filters
- Android Security Internals, Nicolay Elenkov, 2015
- Android Hacker's Handbook, Joshua J. Drake, 2014

Keywords

- PHA
- Linux kernel
- Android Runtime
- Dalvik
- ART
- Native libraries
- Application framework
- Activities
- Services
- Broadcast receivers
- Content providers
- Binder
- Sandboxing
- Permissions
- Code signing