Android Internals - A Confectioner's CookBook

Status (05/25/2022)

- Vol I: Done - v2.1
- Vol II: Done
- Vol III: Still pretty far
- Vol IV: A bit closer

Volume I : The Power User's View

Major rewrite for Android 11/12 - 440 pages, complete (now with Config chapter, too)

Changes/additions with respect to 1st edition shown in yellow

The original (ISBN: 978-0-9910555-2-4) remains free on this website (Thank you, CIA).
The book underwent a complete rewrite (effectively, a 2nd edition). Updates to align with Android 11, and latest SD865, Exynos, MediaTek and Kirin devices

- configfs
- debugfs
- functionfs (/dev/usb-ffs/adb)
- FUSE
- incrementalfs (11.0)
- overlayfs
- procfs (/proc)
- pstore (/sys/fs/pstore)
- sdcardfs/esdfs
- securityfs (/sys/fs/selinux)
- sysfs (/sys)
- tmpfs
- tracefs (/sys/kernel/debug/tracing))

4. Files & Directories

A detailed breakdown of directories and files Android, up to and including Android 11 (Spun off from first edition's discussion in Partitions & Filesystems chapter, and greatly expanded).

- The Root Filesystem
- /system
  - /system/bin
  - /system/lib[64]
    - Core Libraries
    - Other system libraries
    - Framework support libraries
    - External native libraries
  - /system/etc
- /vendor
  - /vendor/bin
    - Qualcomm specific binaries
    - Huawei specific binaries
    - Samsung specific binaries
    - MediaTek specific binaries
- /data (Excerpt)
  - /data/data
  - /data/misc
    - /data/misc_ce and /data/misc_de
  - /data/system
    - /data/system_ce and /data/system_de
  - /data/vendor
- /cache

5. Storage Management

Refactoring vold and OBB/ASEC, as well as adding new content on Storage* and APEX

- Mounting
  - Mount options
  - Loop mounting
  - Bind mounting
  - Mount namespaces
  - fs_mgr
  - The fstab files
  - External Storage
    - Portable Storage
    - Adoptable Storage
  - Scoped Storage (Android 10)
  - Incremental FS (Android 11)
- Daemons
  - vold
  - StorageManager
  - storaged
  - storagestats

- User mode memory management
- USS, PSS, RSS, VSS, etc
- procrank, librank, and /proc/../smaps
- Native binaries, libraries and ELF Tools
- Optimizations in Android native and Dalvik apps

12. Logging, Statistics & Monitoring
    - Android Logging
        - logd
    - Statistics
        - statsd
        - statscompanion
        - The IStats HIDL
        - Lesser Statistics Services
    - Incident Reporting
    - Vendor Diagnostics
        - Qualcomm's Diag (/dev/diag)
    - Debugging
    - Monitoring
        - inotify
        - `ptrace(2)`-based tools
            - `strace`
            - `jtrace`
        - Using eBPF for tracing

13. Power Management
    - Native APIs
    - The `PowerManagerService` and Friends
    - Battery Monitoring
    - Low-level CPU Control
        - MultiCore
        - Interrupt Affinity
        - Governors
        - Heterogeneous Multi-Processing (HMP) Scheduling
        - Energy Aware Scheduling (EAS)
    - Thermal Monitoring
        - Linux kernel support
        - Android support
            - `hardware_properties` service
            - `thermalservice`
            - The thermal HAL
        - Vendor thermal support
            - Qualcomm
            - Samsung
            - Huawei
            - MediaTek
            - Case study: Google Pixel
    - The Power HAL interface
    - Power Management Statistics

Volume II: The Developer View - Available ! 360 pages

1. Building Android from the source
    - The AOSP

7. Android RunTime Internals
    - ART - An alternative to Dalvik
    - ART Components (The com.android.art APEX)
    - The files: OAT, ART, CDEX, VDEX, etc
    - The runtime
        - Setup and initialization
        - Support threads
    - Compilation
    - JIT
    - Profiling/Tracing
    - Memory allocators and Garbage Collection

8. Binder, in depth
    - A Brief Overview of Binder
    - The App Developer's Perspective - AIDL
        - AIDL Syntax
        - AIDL code generation
        - The Parcel object
        - The Parcel wire format
    - The Framework Perspective - `android.os.Binder`
        - Binder.java
        - References
        - Default Transactions
        - 11.0: Extensions
        - Death Notifications
        - The JNI Layer
    - The native code Perspective - `libbinder.so`
        - `RefBase`: Strong and Weak Pointers
        - The `BpBinder` and `BBinder`
        - The native proxy and stub interfaces
        - `ProcessState` and `IPCThreadState`
        - Detailed case study: A Binder service in native code
    - The Binder kernel interface
        - The Binder character devices
            - 11.0; BinderFS
            - The ioctl(2) command set
        - The `BINDER_WRITE_READ` `ioctl(2)` code
        - Transactions
        - Flattened Objects
    - Binder Driver Internals
        - Module initialization
        - Device open
        - Transactions in-kernel processing
        - Kernel Driver State
        - Thread Management
        - Death Notifications
    - Tracing & Debugging

Volume III: The [Hacker/Security Analyst]'s View

This was chapter 8 in the 1st edition - but that was a long time ago, before TrustZone, Titan, AVB ... and a host of Android exploits & APTs..

1. Software-implemented security
   - Linux Native Permissions
     - AID ranges
     - Treble and the return of passwd/group files
   - SELinux
   - SECCOMP-BPF
   - Android Runtime permissions
   - Appops

2. Hardware-backed security
   - TrustZone
     - Theory & Design
     - Vendor Implementations:
       - Qualcomm: QSEE/QHEE
       - MTK/Older Samsung: Mobicore
       - Samsung: TEEGRIS
       - Google: Trusty
   - Beyond Trustzone: Hardware Security Modules
     - Titan M/M2
     - Qualcomm SPU

3. Authentication subsystems
   - The Lock Screen (`lock_settings` service)
   - The `auth` service
   - The `biometric` service
   - Face authentication (The `face` service)

4. Encryption facilities
   - DM-Crypt
   - Ext4Crypt
   - Keystore
   - Linux keyrings
   - Gatekeeper

5. Integrity & Attestation
   - Android Verified Boot
     - AVB 1.0
     - AVB 2.0
     - AVBMeta tool
   - DM-verity
   - 11: App Integrity, File Interity (fs-verity)
   - Samsung TIMA & Knox
   - Google SafetyNet

---

6. Introduction/Threat Modeling Android
   Lorem ipsum
   - Threat Modeling
   - Attack classes
     - ..
     - ...
   - Android Security Model

7. Rooting
   Rooting Android using boot-to-root methods

        - Android IOEMUnlock Interface
        - ...
    - Case Study: Magisk
    - Malware Case Study: Intellexa's "Alien"

8. Vulnerability/Exploit case studies:
   (Jury's still out on which of those I'll use - comments/suggestions welcome)
    - Linux Kernel: CVE-2021-1048 (epoll) or CVE-2022-0847 (Dirty Pipe)
    - AOSP Linux Kernel: Bad Binder (CVE-2019-2215) and/or num_not_so_valid CVE-2020-0041
    - Vendor: Pixel 6 - Samsung's MFC
    - TrustZone: likely Trusty
    - AOSP: (still looking for something nice here)
    - Vendor: MTK-su and/or Boot chain vulnerability?
    - Baseband: Samsung Exynos (Shannon) VoLTE/SIP vulns

9. Appendices:
    - Android App Hardening Guide
    - Android System Hardening Guide

Volume IV: The Implementer's View

1. HAL & Treble
    - The Hardware Abstraction Layer (pre-Treble)
    - hwservicemanager
    - HIDL and Binderized HAL
    - sensors/CHRE
        - Oslo/Soli as a case study
    - GPS
    - Implementing a custom HAL module

2. The Android Input Architecture
    - The Linux Kernel Layer
    - The Native Layer
        - InputManager
        - EventHub
        - InputReader
        - InputDispatcher
    - The Dalvik Layer
        - The Input Pipeline
        - Getting tot the user callback

3. Android Media
    - The Audio Architecture
        - Audio at the Linux Kernel level
        - The Audio HAL
            - Audio modules
            - Audio policy modules
            - Audio Effects
        - AudioFlinger
        - Media Player
        - The Dalvik APIs
    - Video
    - Digital Rights Management