Struttura del File System Android e Radice Root (/)

Mappatura completa di directory, sottocartelle, moduli APEX e processi di sistema

Architettura del File System Android: Sebbene basato su kernel Linux, Android differisce dalle distribuzioni Linux tradizionali. Utilizza una struttura a partizioni separate e isolamento sandbox per garantire sicurezza, stabilità e aggiornamenti modulari (Project Treble e APEX). La directory radice `/` contiene i punti di montaggio per le varie partizioni di sistema e utente.

1. /system — Partizione del Sistema Operativo

`/system` Core OS

Contiene i binari fondamentali, le librerie C/C++, il framework Java/Kotlin e le app di sistema preinstallate. È generalmente montata in sola lettura (read-only).

- `/system/app/`: App di sistema di base (Calcolatrice, Orologio, Contatti, SettingsProvider). Usate dall'utente e dai servizi di background.
- `/system/priv-app/`: App con permessi privilegiati ed elevati (SystemUI, Keyguard, Phone, Google Play Services Core). Utilizzate direttamente dal sistema per la gestione dell'interfaccia e sicurezza.
- `/system/framework/`: File JAR e DEX del Framework Android (`framework.jar`, `services.jar`). Usati dall'ambiente di runtime **ART (Android Runtime)** e dal processo **Zygote** per avviare le app.
- `/system/lib/` e `/system/lib64/`: Librerie nativo-C/C++ condivise (OpenGL, SQLite, Bionic libc, WebKit). Utilizzate sia dalle app utente che da quelle di sistema tramite NDK/JNI.
- `/system/bin/` e `/system/xbin/`: Eseguibili e strumenti a riga di comando (`surfaceflinger`, `vold`, `netd`, `pm`, `am`, `toolbox`). Usati dal sistema operativo per avviare servizi hardware e di rete.
- `/system/etc/`: File di configurazione globale (certificati CA, regole di rete, permessi `permissions.xml`, configurazioni APN per SIM). Usati dai moduli di connettività e telecomunicazioni.

2. /data — Dati Utente e Applicazioni Installate

`/data` Dati Utente & App

La partizione più dinamica. Contiene le applicazioni installate dall'utente, le configurazioni locali, i database SQLite e le cache.

- `/data/app/`: Contiene i file APK ed estratti delle app installate manualmente tramite Play Store o file esterni.
- `/data/data/<package_name>/`: Directory sandbox riservata a ciascuna app (es. `com.whatsapp`). Contiene sub-cartelle private:
  - `databases/`: Database SQLite privati (es. messaggi, preferenze).
  - `shared_prefs/`: Impostazioni dell'app in formato XML.
  - `cache/` e `code_cache/`: File temporanei e codice compilato localmente.
- `/data/system/`: Dati di stato del sistema (PIN/password di sblocco, configurazioni Wi-Fi salvate, registro delle autorizzazioni app). Usato da **LockSettingsService** e **PackageManager**.
- `/data/user/0/`: Link simbolico verso la sandbox dell'utente primario (coincide con `/data/data/`).

3. /vendor e /product — Hardware & Personalizzazioni

`/vendor` HAL & Driver Hardware

Introdotta con Project Treble per separare il codice di sistema da quello specifico dei produttori di chip (Qualcomm, MediaTek, Samsung Exynos).

- `/vendor/firmware/`: Microcodice caricato direttamente sulle componenti hardware (Wi-Fi, Bluetooth, Modem, GPU). Usato dai driver del kernel.
- `/vendor/lib/` e `/vendor/lib64/`: Librerie HAL (Hardware Abstraction Layer) per l'interfacciamento con fotocamere, sensori e audio. Usate dai servizi `cameraserver` e `audioserver`.
- `/vendor/bin/hw/`: Daemon proprietari di gestione hardware (es. daemon del sensore di impronte digitali o ricarica rapida).

`/product e /system_ext` Personalizzazione OEM

Contiene le personalizzazioni grafiche e le app specifiche del produttore dello smartphone (es. Samsung One UI, Xiaomi HyperOS, Google Pixel Features).

4. /apex — Moduli di Sistema Aggiornabili

`/apex` Android APEX Container

Introdotto in Android 10 (Project Mainline) per consentire a Google di aggiornare componenti critici del sistema direttamente tramite Google Play Store senza un aggiornamento firmware completo (OTA).

- `com.android.art/`: Il modulo contenente il runtime Android (ART), compilatore e librerie Java standard.
- `com.android.media/`: Codec multimediali ed estensioni di riproduzione audio/video.
- `com.android.conscrypt/`: Librerie crittografiche e gestione dei certificati TLS/SSL di sistema. Usato da browser e app per connessioni sicure.
- `com.android.btservices/`: Stack Bluetooth aggiornabile separatamente.

5. /storage e /mnt — Memoria di Archiviazione Visibile

`/storage e /mnt` Media & User Files

Gestiscono i punti di montaggio per l'archiviazione di massa condivisa (foto, video, documenti, download) e schede microSD esterne.

- `/storage/emulated/0/`: Il punto di accesso principale per l'utente e le applicazioni per accedere alla memoria interna. Corrisponde alle cartelle pubbliche:
  - `DCIM/` e `Pictures/`: Usate dall'app **Fotocamera** e dalle app di galleria.
  - `Download/`: Usata dai browser (Chrome, Firefox) e gestori di file.
  - `Android/data/`: Cartelle di supporto esterne delle app utili per file di grandi dimensioni (es. mappe offline, risorse di gioco).
- `/mnt/media_rw/` e `/storage/XXXX-XXXX/`: Punti di montaggio diretti per memorie SD esterne o chiavette USB OTG. Usati dal servizio `vold` (Volume Daemon).

6. Virtual File Systems e Hardware (/proc, /sys, /dev)

`/proc & /sys` Kernel Virtual Filesystem

Non occupano spazio su disco reale. Sono interfacce generate dal Kernel Linux in memoria RAM per comunicare lo stato dell'hardware e dei processi in tempo reale.

- `/proc/meminfo`: Utilizzato da Android per monitorare l'uso della RAM e dal processo **LMKD (Low Memory Killer Daemon)** per chiudere app in background in caso di esaurimento memoria.
- `/proc/cpuinfo`: Riporta frequenza, architettura e stato dei core della CPU. Usato da app di benchmarking e monitoraggio.
- `/sys/class/power_supply/`: Contiene le informazioni sullo stato della batteria, temperatura, voltaggio e velocità di ricarica. Usato dal servizio di sistema `batterystats`.

`/dev` Device Nodes

Interfacce di basso livello per i dispositivi hardware fisici ed emulati.

- `/dev/binder`: Il canale di comunicazione Inter-Process Communication (IPC) fondamentale di Android. Usato da tutte le app per comunicare con i servizi di sistema.
- `/dev/block/`: Rappresenta i blocchi di memoria fisica (partizioni UFS/eMMC come `boot`, `system`, `userdata`). Usato dai processi di aggiornamento e formattazione.
- `/dev/graphics/fb0` o `/dev/dri/`: Interfaccia per il controller del display usata dal servizio grafico `surfaceflinger`.

7. Altre Directory della Radice Root (/)

`/etc`

Link simbolico a `/system/etc` per compatibilità con i comandi Linux standard.

`/sdcard`

Link simbolico storico verso `/storage/emulated/0` per mantenere la retrocompatibilità con le app più datate.

`/apex e /bootstrap`

Cartelle temporanee di avvio usate dallo strumento `init` per montare i moduli fondamentali prima dell'avvio completo del sistema.

`/linkerconfig`

Generato dinamicamente all'avvio per definire quali librerie nativo-C le app e i daemon possono caricare.

Sintesi dei Permessi e Ruoli delle Applicazioni

In Android, le **app utente standard** possono accedere liberamente soltanto alla propria sandbox in `/data/data/<package>/` e alla memoria condivisa in `/storage/emulated/0/` (previo consenso dei permessi multimediali). L'accesso a `/system`, `/vendor`, `/data/system` e ai nodi di sistema in `/dev` è strettamente riservato ai **servizi con privilegi di Root o uid system**, proteggendo l'integrità dello smartphone da malware e modifiche non autorizzate.