

Čo vie Play services

Statická reverzná analýza balíka **com.google.android.gms 26.40.32**, ako ho nezmenený a podpísaný Googlom distribuuje projekt GrapheneOS, a čo to znamená, keď je aplikácia zavretá v sandboxe.

402

OPRÁVNENÍ

468

ZDROJOV LOGOV

200

DEKLAROVANÝCH MODULOV

- | | | | |
|----|----------------------|----|---------------------|
| 01 | Zhrnutie | 02 | Pôvod a metóda |
| 03 | Anatómia aplikácie | 04 | Identita: rebrík |
| 05 | Čo opúšťa zariadenie | 06 | Kedy sa to deje |
| 07 | Kam to ide | 08 | Rozdiel GrapheneOS |
| 09 | Odporúčania | 10 | Nad rámec nastavení |
| 11 | Stiahnutý kód | 12 | Dôkazy |
| 13 | Podpora | | |

V sandboxe sú Play services aplikácia, ktorá potrebuje súhlas a dá sa jej ho odobrať. Súkromnou sa tým nestanú.

Zhrnutie

GrapheneOS dodáva Google Play services presne tak, ako ich zostavil Google. APK sa od Googlovho nelíši, súkromie vzniká tým, že operačný systém mu odmieta privilégiá, okolo ktorých bolo navrhnuté. Report meria obe polovice: čo aplikácia obsahuje a čo jej sandbox berie.

402 DEKLAROVANÝCH OPRÁVNENÍ	481 EXPORTOVANÝCH KOMPONENTOV	778 DOSLOVNÝCH HOSTITEĽOV	468 ZDROJOV TELEMETRIE
161 628 DEKOMPILOVANÝCH TRIED	200 DEKLAROVANÝCH MODULOV	12 h INTERVAL CHECK- INU	132 NIKDY NEUDEĽITEĽNÝCH OPRÁVNENÍ

Šesť zistení

- Identita vzniká pri prvom check-ine a prežije všetko okrem vymazania dát aplikácie.** Zariadenie dostane 64-bitové Android ID, logovacie ID check-inu a bezpečnostný token, zapísané do `/data/misc_ce/<user>/checkin` vo vlastnom sandboxe aplikácie. Na tejto trojici stojí každé neskoršie priradenie, ktoré Google dokáže urobiť.
- Odosielanie cez Clearcut je z návrhu viazané na účet.** Logovací kontext nesie `uploadAccount`, `logAndroidId`, `isAnonymous`, názov a verziu balíka a úroveň PII. V tomto zostavení existuje 468 samostatných zdrojov logov. ZISTENÉ
- Sandbox naozaj odoberá nebezpečnú polovicu.** 132 deklarovaných oprávnení je `signature / privileged` a nikdy sa nedajú udeliť bežnej aplikácii: čítanie IMEI bez súhlasu, MAC adresy, logy, zabezpečené nastavenia, dáta iných používateľov. Ďalších 175 AOSP vôbec nedefinuje. ZISTENÉ
- Aký kód beží, sa rozhoduje až za behu.** APK deklaruje 200 modulov a na auditovaný telefón sa stiahlo ďalších 13, ktoré v repozitári vôbec nie sú. Moduly, ktoré som stiahol z testovacieho zariadenia a dekompiloval, sú obyčajný kód (čítanie čiarových kódov a textu, vykresľovanie máp, reklamy, passkeys, vyhľadávanie v launcheri, vyhľadávací index v zariadení), no každý z nich beží so všetkými oprávneniami Play services a sám nežiada žiadne. Aj engine DroidGuard je stiahnuté APK s pripnutým certifikátom. Plán modulov je vzdialený flag, ktorý sa sťahuje spolu s identifikátormi zariadenia a účtu, takže Google môže jednému zariadeniu alebo účtu poslať iný podpísaný modul bez aktualizácie Play services. Načíta sa len kód podpísaný Googlom. Sandbox GrapheneOS tento kód obmedzuje, ale jeho príchodu nezabráni. **Neexistuje teda jedno „Play services“, ktoré by všetci spúšťali a ktoré by sa dalo raz zaudítovať pre všetkých.** Zariadenie pošle Googlu svoju konfiguráciu a Google rozhodne, aké moduly a v akých verziách dostane práve ono. Tento report auduje hostiteľskú aplikáciu (inštalčný balíček Play services, APK) a vzorku dynamického kódu, ktorý si stiahol jedno zariadenie. ZISTENÉ (kapitola 11)
- To, čo zostane, stále stačí na identifikáciu.** `READ_PHONE_STATE` (udeliteľné používateľom) odomkne IMSI a sériové číslo SIM pre check-in. `READ_CONTACTS` odomkne nahrávanie kontaktov. `QUERY_ALL_PACKAGES` je *normálne* oprávnenie, takže zoznam nainštalovaných aplikácií je dostupný bez akejkoľvek výzvy. ZISTENÉ

6. **Jediným tvrdým vypínačom je prístup k sieti a neodporúčam ho.** V GrapheneOS je samotné **INTERNET** nebezpečné oprávnenie, takže prepínač siete pre aplikáciu naozaj preruší telemetriu Googlu. Preruší však aj push notifikácie, čo je hlavný dôvod, prečo väčšina ľudí Play services inštaluje, takže rozbije väčšinu toho, kvôli čomu ich chcete. **ODVODENÉ**

Čo sa s tým dá robiť. Sekcia 09 uvádza nastavenia, ktoré existujú dnes; sekcia 10 ide ďalej: filtrovanie siete, úprava APK a patch GrapheneOS. Praktické je filtrovanie siete. Úpravu APK a patch GrapheneOS **neodporúčam**: sú tu ako teoretická úvaha o tom, či by to malo zmysel, a sú oveľa väčšia otrava, než sa zdá.

Ako čítať tento report. Všetko je statická analýza APK. Nič z toho nebolo zachytené z reálnej prevádzky zariadenia. Tvrdenia označené **ZISTENÉ** majú súbor a riadok v prílohe; **ODVODENÉ** znamená, že záver je úvaha nad videným. Správanie na strane servera (čo Google uchováva, ako záznamy spája) je mimo toho, čo APK dokáže povedať, a report o ňom nič netvrdí.

02

Pôvod a metóda

Pred akýmkoľvek tvrdením o správaní musí byť artefakt skutočne ten artefakt. APK bolo stiahnuté z vlastného repozitára aplikácií GrapheneOS a každý bajt bol overený voči podpísaným metadátam repozitára.

POLOŽKA	HODNOTA
Repozitár	https://apps.grapheneos.org/ · metadata.1.0.0.sjson (podpis signify)
Balík	com.google.android.gms · versionCode 264032035 · 26.40.32 (260400–995630914)
Kanál / SDK	stable · minSdk 35 · arm64-v8a + armeabi-v7a
base.apk	194 764 356 bajtov · sha256 1bfbf0c0ddab7470... (zhoduje sa s metadátami)
Podpisovateľ	CN=Android, O=Google Inc. · sha256 5f2391277b1dbd48...b5402
Analyzované splity	AdsDynamite · CronetDynamite · DynamiteLoader · DynamiteModulesA/C · GoogleCertificates · MapsDynamite · MeasurementDynamite · maps_core · config.en · config.xxhdpi
Spríevodné aplikácie	com.google.android.gsf 14 · com.android.vending 53.5.48–34

Prečo záleží na podpise. GrapheneOS Play services nepatchuje; patchuje *operačný systém*, aby nezmenená binárka Googlu bežala bez privilegovaných háčikov, ktoré očakáva. Vďaka zhode podpisu je fráza „oficiálne vydanie bez úprav“ v popise repozitára overiteľné tvrdenie, nie marketing.

Čo sa urobilo

Rozbalenie	apktool (zdroje + manifest), surový zip pre assets a natívne knižnice
Reťazce	tabuľky reťazcov DEX parsované priamo (773 000 reťazcov) plus surové prehľadanie bajtov každého assetu
Kód	jadx 1.5.6, úplný prechod: 161 628 tried, 920 MB Javy, plus cielené čítanie modulov check-in, Clearcut, Measurement, Phenotype, reklamy a verifikátor
Politika	úrovne ochrany oprávnení porovnané s GrapheneOS <code>frameworks/base</code> , vetva 17
Rozsah	GMS base + 11 splitov; GSF a Play Store len pre kontext identity
Ďalšia analýza	Dekódované manifesty modulov (200 modulov, 80 položiek modulových sád); triáž všetkých 39 natívnych knižníc arm64, tie podstatné pre súkromie dekompilované v Ghidre
Moduly	24 stiahnutých modulových APK (19 buildov), engine DroidGuard a register modulov z rootnutého testovacieho emulátora s Androidom 13, overené proti SHA-256 zo záznamov Play a dekompilované (kapitola 11)

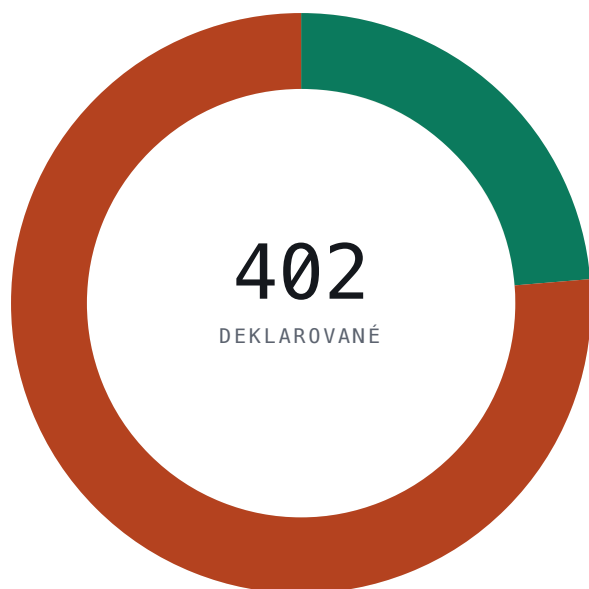
Čo sa nepodarilo zistiť

- Žiadny záchyt paketov: frekvencie nižšie vychádzajú z kódu a predvolených hodnôt príznakov, nie z pozorovania.
- Význam jednotlivých polí check-inu: protobuf požiadavky nemá mená, takže čísla polí sa nedajú označiť s istotou. Vidno však *vstupy* požiadavky a tie report cituje.
- Engine DroidGuard sa podarilo získať (x86-64 build z testovacieho emulátora, kapitola 11), ale program, ktorý vykonáva, prichádza s každou požiadavkou zvlášť a zachytený nebol. Čo pri konkrétnom volaní meria, preto nevieme. Natívny overovač atestácie (`liboak`) kontroluje servery Googlu, nie telefón.
- Všetko, čo Google robí po prijatí: uchovávanie, spájanie, trénovanie modelov.

03

Anatómia aplikácie

Play services sú hostiteľ modulov. Deklarujú 1,310 komponentov, ale 1 054 z nich patrí 200 modulom popísaným v `assets/ChimeraManifest.pb`. Druhý manifest, `assets/ChimeraModuleSetList.pb`, uvádza 80 položiek modulových sád: 12 z nich sú on-demand splity a 8 modulov sa dodáva dvakrát, raz ako APK v `assets/chimera-modules/` a raz ako install-time split. Na auditovanom Pixeli 10 Pro Fold ART kompiloval **trinásť** modulových APK z `/data/user_de/0/com.google.android.gms/app_chimera/m/` a ani jedno z nich nie je v repozitári GrapheneOS. 17 súborov DEX, 801 assetov, 29,5 MB natívneho kódu v base.apk a ďalšie 3 knižnice v dvoch splitoch. ZISTENÉ



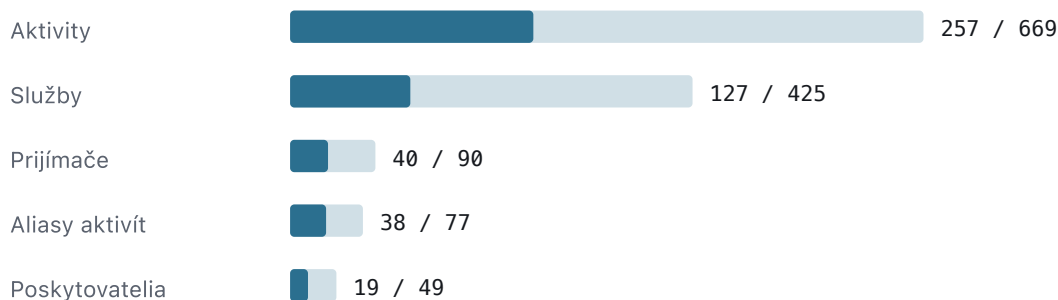
Obrázok 1: dá sa niekedy udeliť?



■ normálne · 55 ■ nebezpečné · 35 ■ špeciálny prístup · 5 ■ signature/privileged · 132 ■ nie je v AOSP · 175

Obrázok 2: rovnakých 402 oprávnení podľa úrovne ochrany

Povrch útoku



Obrázok 3: deklarované vs. exportované komponenty (exportované / spolu)

1,310 komponentov, z toho 481 exportovaných. Export je spôsob, ako sa iné aplikácie dostanú do Play services: poskytovateľ účtov, slice provider Fast Pair, konfiguračný provider Phenotype, sprostredkovateľ modulov Chimera, provider vyhľadávania v Nastaveniach. Len 60 z týchto exportovaných komponentov deklaruje `android.permission`. Ostatných 421 je na úrovni manifestu otvorených, vrátane poskytovateľa App Set ID a služieb U2F/FIDO, a volajúceho si overujú až v kóde.

Oprávnenia

GMS žiada o všetko, pretože je napísané ako privilegovaná systémová aplikácia. V AOSP ňou aj je: podpísaná kľúčom platformy, uložená v `/system/priv-app`, s oprávneniami typu signature udelenými pri štarte.

V modeli GrapheneOS je to normálna aplikácia s normálnym UID. 132 oprávnení typu signature/privileged je jednoducho zamietnutých a 175 ďalších odkazuje na oprávnenia, ktoré nedefinuje žiadny build AOSP, takže sú zamietnuté tiež. Zostáva oveľa menšia, ale nie prázdna množina schopností.

AndroidManifest.xml

```
android:sharedUserId="com.google.uid.shared"
android:isolatedSplits="true" android:forceQueryable="true"
android:compileSdkVersion="37"

com.google.android.gms.checkin.CheckinService
com.google.android.gms.checkin.CheckinServiceSecretCodeReceiver
com.google.android.gms.appset.service.AppSetIdProviderService (exportovaný, bez oprávnenia)
```

Zdieľané UID. `com.google.uid.shared` znamená, že Play services, Google Services Framework a Play Store bežia ako jeden linuxový používateľ. Zdieľajú dátový adresár, priestor kľúčov a Android ID. Sandbox ich izoluje od vás; neizoluje ich od seba navzájom.

04

Identita: rebrík

„Viazané na používateľa“ nie je jeden identifikátor. GMS udržiava rebrík identifikátorov a každá priečka prežije iný druh resetu.

01

Android ID (SSAID)

64-bitová hex hodnota viazaná na podpisový kľúč + používateľa + zariadenie. GMS a ostatné aplikácie Googlu zdieľajú jednu hodnotu, lebo sú podpísané rovnakým kľúčom (a bežia pod jedným UID). Prežije preinštalovanie aplikácie aj aktualizácie OS; vynuluje sa až obnovením továrenských nastavení.

ZISTENÉ Settings.Secure „android_id“

02

Logovacie ID check-inu + bezpečnostný token

Vydáva ich server check-inu pri prvom kontakte a ukladajú sa v sandboxe aplikácie (`/data/misc_ce/<user>/checkin, security_token`). Je to „GSF ID“ z folklóru. Robí z Android ID identitu zariadenia nezávislú od účtu. **ZISTENÉ** `bjsc.java:41-42,122-124`

03

Reklamné ID (GAID) a App Set ID

Generujú sa a ukladajú lokálne v Play services, používateľ ich môže resetovať a aplikáciám sa sprístupňujú cez binder `IAdvertisingIdService`. Limit ad tracking je lokálny príznak; identifikátory zostávajú, kým ich používateľ neresetuje. **ZISTENÉ** `adid/service/*, „limit_ad_tracking“`

04

Účet Google

Po pridaní účtu drží GMS jeho meno, Gaia ID a tokeny OAuth a Clearcut/Phenotype môžu k nemu viazať udalosti. Databáza Phenotype má tabuľku `accounts` a explicitný anonymný riadok (`accounts.name = ''`). **ZISTENÉ** `phenotype/sync/*, PlayLoggerContext`

05

Odtlačok zariadenia

Polia buildu, obrazovka, jazyk, operátor SIM a s `READ_PHONE_STATE` aj IMSI a sériové číslo SIM. Posiela sa pri check-ine. **ZISTENÉ** `bjtv.java:131,340,347`

Identita atestácie

DroidGuard je stiahnuté APK. Podpísaná *creation response* dodá `byteCode`, `vmUrl`, `vmChecksum` a dobu platnosti a jej podpis sa overí skôr, než sa použije ktorékoľvek pole. Engine sa stiahne z `https://www.gstatic.com/droidguard/`, uloží do `app_dg_cache/<key>/the.apk`, overí proti dvom pripnutým odtlačkom podpisového certifikátu a načíta cez `DexClassLoader` ako `com.google.ccc.abuse.droidguard.DroidGuard`. Google ho môže kedykoľvek vymeniť; jedinou lokálnou kontrolou je ten pin. Výsledky Play Integrity potom viažu požiadavku aplikácie na kľúč uložený v hardvéri. **ZISTENÉ** `bysw.java:90–92`, `bysr.java:14–15`, `bysq.java:53–60` · **ODVODENÉ** (význam výsledkov)

Ako spolu identifikátory súvisia

IDENTIFIKÁTOR	KTO PRIDEĽUJE	ŽIVOTNOSŤ	RESET	VIDITEĽNÝ APLIKÁCIÁM?
SSAID	OS, podľa podpisového kľúča	zariadenie	továrenský reset	áno, bez oprávnenia
ID check-inu	Google, pri check-ine	inštalácia	vymazanie dát aplikácie	nie
Bezpečnostný token	Google, pri check-ine	inštalácia	vymazanie dát / 401	nie
GAID	Play services, lokálne	riadi používateľ	Nastavenia → Reklamy → reset	áno, cez binder
App Set ID	Play services, lokálne	rozsah vývojára	reset / odinštalovanie všetkých	áno, cez binder
Účet / Gaia	Google	účet	odhlásenie	cez AccountManager
RLZ / staré klientske ID	Google, staršie	inštalácia	rôzne	nie (GServices)

Odhlásenie. Odstránenie účtu Google nevynuluje priečky 01–03. ID check-inu zostáva, Android ID zostáva a reklamné ID zostáva. „Odhlásený“ znamená „nepripojený k účtu na serveri“, nie „neidentifikovaný“. **ODVODENÉ**

Čo opúšťa zariadenie

Trinásť podsystémov odnáša dáta zo zariadenia. Pre každý uvádza tabuľka koncový bod nájdený v kóde, čo ho spúšťa, čo nesie a čo na ňom mení sandbox GrapheneOS.

PODSYSTÉM	KONCOVÝ BOD (AKO NÁJDENÝ)	ČO ODCHÁDZA	V GRAPHENEOS
Check-in registrácia zariadenia	android.googleapis.com/checkin android.clients.google.com/checkin checkin.gstatic.com/checkin	Android ID, logovacie ID, bezpečnostný token, odtlačok buildu, trieda zariadenia, jazyk, operátor SIM; IMSI a sériové číslo SIM, ak je udelené oprávnenie Telefón; inventár modulov. Odpoveď môže obnoviť Android ID (napr. po strate bezpečnostného tokenu).	Funguje. Bez READ_PHONE_STATE sú polia SIM prázdne. Všetko ostatné zariadenie stále identifikuje.
Clearcut play log	play.googleapis.com/log play.googleapis.com/log/batch	Dávky udalostí s balíkom, verzou, zdrojom logu, úrovňou PII a voliteľne účtom + Android ID; existuje 468 kategórií zdrojov logov.	Beží. Nepotrebuje nič okrem siete. Účet sa pripája, len ak existuje.
Firebase / Measurement	app-measurement.com/a app-measurement.com/s/d app-measurement.com/config/app/...	ID inštancie aplikácie, id/verzia aplikácie, OS, model zariadenia, jazyk, krajina, relácie a tok udalostí z aplikácií, ktoré vkladajú Firebase.	Beží. Predvolene používa vlastné ID inštancie aplikácie, nie Android ID.
Phenotype konfigurácia a experimenty	cloudconfig.googleapis.com (hostiteľ služby sa určuje za behu)	Konfigurácia zariadenia, tokeny experimentov, riadky podľa účtu; synchronizácia každých 24 h. Tokeny sa potvrdzujú a môžu sa odosielať v dávkach.	Beží. Tokeny experimentov sú pseudonymné, ale dlhodobé.
Autentifikácia	android.clients.google.com/auth/* oauth2.googleapis.com	Názov účtu, tokeny OAuth, overenie FRP, stav zariadenia pre obnovu účtu.	Beží. Tokeny sú v zdieľanom UID Googlu.
Verify Apps Play Protect	security.verifyapps. UPLOAD_APK → Google	Bajty APK, názov balíka, kód verzie, zdroj inštalácie pre skenované balíky.	Len ak je Play Protect zapnutý v sandboxovanom Play Store. Dá sa vypnúť.
Kontakty romanesco	people-pa.googleapis.com	Záznamy kontaktov a metadáta synchronizácie, ak je udelené oprávnenie Kontakty; tiež vyhľadávanie zhôd kontaktov.	Zamietnuté, kým používateľ neudelí Kontakty. Predvolená inštalácia: bez prístupu.
Poloha / Nearby	nearby.googleapis.com maps.googleapis.com fitness.googleapis.com	Žiadosti fused location, výsledky BLE/UWB, zaznamenané zariadenia Fast Pair, vyhľadávanie miest.	Potrebuje presnú/približnú polohu + skenovanie Bluetooth. Vlastná sieťová poloha GrapheneOS môže nahradiť Googlovu.
FCM / push	mtalk.google.com:5228 gcm.googleapis.com android.apis.google.com/c2dm/register3	Trvalý socket, registračný token, potvrdenia doručenia: živý signál „toto zariadenie je online z tejto IP“.	Beží, ak je online. Zablokovanie siete zabije push spolu s telemetriou.
Reklamy	googleads.g.doubleclick.net pagead2.googleadsyndication.com	Reklamné ID + príznak limitu aplikáciám, ktoré o ne požiadajú; pingy žiadostí o reklamy, keď aplikácie používajú AdMob.	GAID existuje lokálne; používateľ ho môže resetovať/obmedziť. Privilegovaná cesta AD_ID nie je dostupná.

PODSYSTÉM	KONCOVÝ BOD (AKO NÁJDENÝ)	ČO ODCHÁDZA	V GRAPHENEOS
Atestácia	gstatic.com/droidguard/* API playintegrity cez Play services	Dve rôzne atestácie, ktoré sa často zamieňajú. Atestácia zariadenia je DroidGuard: stiahnuté APK s pripnutým certifikátom (04.06), ktoré zbiera dôkazy o telefóne. Atestácia servera je <code>liboak_attestation_verification_jni.so</code> : overuje dôkazy AMD SEV-SNP a Intel TDX, reťazce DICE a záznamy v transparentnom logu Rekor o <i>vlastných</i> dôverných prostrediach Googlu. O zariadení nečíta nič a do siete nekomunikuje.	Základné úrovne môžu prejsť; úrovne device/strong závisia od certifikácie, ktorú GrapheneOS nemá.
Cobalt súkromná analytika	dodaná konfigurácia (assets/cobalt_config.pb)	Počítadlá so šumom v štýle RAPPOR („one_day_actives“, histogramy) namiesto surových udalostí.	Je súčasťou balíka, ale oproti Clearcut je to zanedbateľné; uvedené pre úplnosť.
SNET / SafetyNet telemetria stavu zariadenia	safebrowsing.google.com/safebrowsing/uploads/android	Vlastné UUID zariadenia (<code>snet_shared_uuid</code> , náhodné, obnoviteľné vzdialeným flagom <code>shared_uuid_reset_counter</code>); 2 % vzorka aplikácií z launchera s odtlačkami APK; metadáta DEX nainštalovaných aplikácií; root/setuid súbory; stav SELinuxu; systémový event log; logcat; app-ops; mount options; údaje o operátorovi.	Beží aj bez Google účtu. Prepínač nahrávania v Play Protect ho nezastaví, ten riadi len nahrávanie APK. Logcat a event log bežná aplikácia nedostane. ZISTENÉ · ODVODENÉ (čo blokuje sandbox)

Tri ukážky kódu

```
defpackage/jwgm.java:12
    return (String) a.h(1, "batch_server_url",
        "https://play.googleapis.com/log/batch").lw();

com/google/android/gms/clearcut/internal/PlayLoggerContext.java:67
    "PlayLoggerContext[package=" + a + ",packageVersionCode=" + b +
    ",logSource=" + c + ",logSourceName=" + f + ",uploadAccount=" + d +
    ",logAndroidId=" + e + ",isAnonymous=" + g + ",qosTier=" + h +
    ",appMobilespecId=" + i + ",scrubMccMnc=" + j + "piiLevelset=" + k + '']';
```

Ten druhý blok je celý zmluvný rámec telemetrie v jednom riadku: ktorá aplikácia, ktoré zostavenie, ktorý zdroj logu, **ktorý účet a či sa pripája Android ID**. `scrubMccMnc` a `piiLevelset` ukazujú, že Google výslovne klasifikuje, nakoľko je každá udalosť osobná. Je to zámerný návrh, nie náhoda logovania.

```
defpackage/bjtv.java:24
    https://%s/checkin with SystemProperties "gms.checkin.sw_domain"
                        default "android.clients.google.com"

defpackage/bjtv.java:131
    checkPermission("android.permission.READ_PHONE_STATE", ...) == 0

defpackage/bjtv.java:340,347
    getSimSerialNumber() · getSubscriberId()
```

Check-in zámerne siaha po IMSI a sériovom čísle SIM, dvoch identifikátoroch, ktoré prežijú továrenský reset na strane operátora. V GrapheneOS sú za oprávnením, ktoré môže používateľ odmietnuť. V origináli Androidu prichádzajú cez `READ_PRIVILEGED_PHONE_STATE` bez výzvy.

```
com/google/android/gms/security/verifier/ApkUploadChimeraService.java:20
    "com.google.android.gms.security.verifyapps.UPLoAD_APK"
InternalApkUploadChimeraService:
    package_name=%s, version_code=%d, state=%d → full APK upload
```

Play Protect nie je vyhľadanie hashu. Verifikátor zaradí do fronty samotné APK. Ako bezpečnostná funkcia je to obhájitelné, ako predvolené nastavenie súkromia nie, preto má v odporúčaniach vlastný vypínač.

Modulový systém je druhý distribučný kanál. Na auditovanom zariadení sa z `app_chimera/m/` skompilovalo trinásť modulových APK a žiadne z nich nie je v repozitári GrapheneOS. Z APK, ktoré si nainštalujete, sa nedá zistiť, ktoré moduly konkrétny telefón dostane: zoznam určuje konfiguračný súbor od servera (magic `CHMR`) v adresári aplikácie. Čo moduly obsahujú, opisuje kapitola 11. [ZISTENÉ](#) (`logcat` zo zariadenia; `affx.java:35`)

Socket otvárajú len tri natívne knižnice. `libbetocore` (Nearby, Rust/QUIC), `libjingle_peerconnection_so` (WebRTC, surové UDP) a `libopenscreen_jni` (Cast, mDNS + UDP). Žiadna z 39 knižníc nemá natívneho HTTP klienta, pevnú IP adresu ani natívne pinovanie certifikátov. Filter podľa hostiteľa teda vidí názvy hostiteľov a nie cesty, ako hovorí kapitola 10.1. Obava, že natívny kód skrýva ľubovoľnú HTTP prevádzku, sa však nepotvrdila.

[ZISTENÉ](#)

Čo GrapheneOS prejde a čo nie

Play Integrity dáva aplikácii tri úrovne hodnotenia. Ako na tom GrapheneOS je:

ÚROVEŇ	V GRAPHENEOS SO PLAY SERVICES V SANDBOXE
Strong integrity (hardvérová atestácia podpísaná Googlom)	GrapheneOS ju nevie preukázať, takže aplikácie, ktoré ju vyžadujú, ostanú zablokované, a nedá sa to obísť nastavením ani obmedzením kódu Googlu. Strong integrity stojí na hardvérovej atestácii kľúčov v Androide, ktorú GrapheneOS plne podporuje, ale Play Integrity API ju viaže na operačné systémy certifikované Googlom. Sprievodca kompatibilitou atestácie od GrapheneOS vysvetľuje, ako by aplikácie mohli namiesto toho akceptovať kľúče GrapheneOS. Ktoré bankové aplikácie sa to týka, sleduje komunita v zozname kompatibility PrivSec, za ktorý GrapheneOS neručí. ZISTENÉ (dokumenty) · NETESTOVANÉ (mnou)
Device integrity a basic integrity	Pokiaľ viem, so Play services v sandboxe GrapheneOS zvyčajne prejdú a mnoho aplikácií kontroluje len tieto úrovne. Príručka GrapheneOS uvádza, že prejde <code>basicIntegrity</code> , ale nie je certifikovaný Googlom, a hovorí o staršej službe SafetyNet. ODVODENÉ · NETESTOVANÉ (mnou)

Dôsledok pre každého, koho láka Play services oklieštiť: DroidGuard generuje podklady pre tieto výsledky. Bez neho sa nevyhodnotí ani to, čo dnes prechádza, takže aplikácia, ktorá dnes funguje, by mohla začať hlásiť chybu integrity. Čokoľvek, čo vypne DroidGuard alebo stiahnutý kód, ktorý potrebuje (pozri bod 8 v sekcii 09), toto riskuje. [ODVODENÉ](#)

Používajú samotné Play services Play Integrity?

Áno, v dvoch zmysloch, ktoré treba oddeliť.

- Obsahujú ho.** Zostavenie má modul Integrity API (`play/integrity/api/service/`: `IntegrityApiPersistentService`, služba `display-listener` a služba `location-trust`), samostatný modul `AutoProtect`, ktorého `AipTelemetryIntentOperation` loguje udalosti s `EventType`, `PackageName`, `VersionCode` a `Payload`, a tri služby `droidguard.*`, z toho dve exportované. [ZISTENÉ](#)
- Atestáciu používajú aj pre vlastné funkcie.** Približne 25 súborov mimo balíka DroidGuard volá klienta DroidGuard. Konkrétne prípady: modul reklám plánuje `ads.fetch_integrity_token.one_time` a `.periodic`, ukladá „device integrity token“ a kontroluje jeho podpis a verejný kľúč; tok overenia telefónneho čísla vo Firebase

`firebase_pnv_preflight` volá `getIntegrityToken` a berie snímku DroidGuard (`ckpu.java:51,62`); crowdsourcované hlásenia polohy Nájst moje zariadenie žiadajú výsledky DroidGuard (`LocationReportUploadIntentOperation.java:208` , tok `finder_crowdsourcedReport`); nastavenie zariadenia posiela `droidguardResult` ; a modul `PoTokens` má `IntegrityTokenRefreshTaskService` . [ZISTENÉ](#)

Nesledoval som, či každé z týchto volaní ide cez verejné Play Integrity API alebo priamo do DroidGuard, motora pod ním; názvy tried to nepovedia. V GrapheneOS sú výsledky slabšie ako na certifikovanom zariadení, takže funkcie, ktoré trvajú na silnom výsledku, môžu zlyhať alebo prejsť na náhradu. [ODVODENÉ](#)

06

Kedy sa to deje

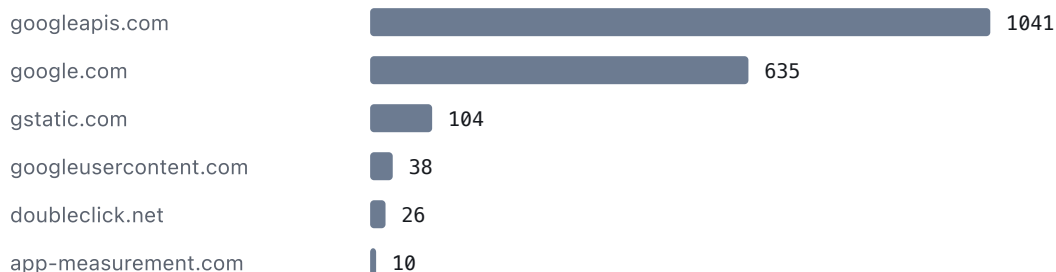
„Kedy“ je miesto, kde väčšina diskusií o súkromí zlyháva. GMS nie je hasičská hadica; je to plánovač s tuctom spúšťačov a väčšina z nich sú obyčajné udalosti Androidu.

prvé spustenie	Check-in. Zaregistruje zariadenie, vydá logovacie ID a bezpečnostný token; pri strate tokenu môže obnoviť Android ID.
každých 12 h	Periodický check-in. Predvolené <code>checkin_interval</code> = 43 200 s s flexibilitou 10 800 s; konfigurácia Chimera používa 43 201 s. ZISTENÉ
každých 24 h	Synchronizácia Phenotype. <code>Phenotype__checkin_interval_in_seconds</code> = 86 400. Sťahuje príznaky, potvrdzuje tokeny experimentov. ZISTENÉ
pri udalosti	Clearcut. Udalosti sa radia a odosielaajú v dávkach cez playlog uploader; pri zlyhaní sa predlžuje odstup. Štart, aktualizácia modulu a vynútené odoslanie sú výslovné spúšťače. ZISTENÉ
pri zmene	Opakovaný check-in. Zmena stavu SIM, zmena pripojenia, zmena účtu Google, zmena GServices, pridanie/nahradenie balíka, prítomnosť používateľa. ZISTENÉ <code>bttj.java</code> , <code>CheckinServiceTriggerReceiver</code>
stále	Socket FCM. Trvalé spojenie na <code>mtalk.google.com</code> (záložné alt1–alt8, :443 a :5228) udržuje zariadenie dosiahnuteľné a prezrádza, že žije. ZISTENÉ
na požiadanie	Measurement a reklamy. Analytika sa odosiela, keď aplikácia zaloguje udalosť; reklamný binder odpovedá vždy, keď aplikácia žiada o reklamné ID. ZISTENÉ

Čo to znamená v praxi. Telefón, ktorý deň nečinne leží, sa s Googlom stále spojí aspoň ~2× kvôli check-inu, ~1× kvôli konfigurácii, nepretržite kvôli push a vždy, keď aplikácia integrovaná s Play niečo robí. Prepínač siete preto nie je opatrenie „blokovať reklamy“, je to tvrdý offline vypínač pre celý ekosystém Play. [ODVODENÉ](#)

Kam to ide

V tomto zostavení sa ako literály vyskytuje 778 rôznych názvov hostiteľov. Zhlukujú sa na niekoľkých doménach Googlu; počty sú doslovné výskyty v APK, nie objem prevádzky.



Obrázok 4: doslovné názvy hostiteľov podľa registrovanej domény Googlu (prvých 6 zo 155)

HOSTITEĽ	ÚLOHA
www.googleapis.com	REST vstupná brána pre väčšinu API Googlu (auth, people, games, ...)
support.google.com	stránky pomocníka otvárané vo WebView
accounts.google.com	prihlásenie, výber účtu, OAuth
play.google.com	odkazy do obchodu, Play Points, install referrer
android.googleapis.com	predvolený koncový bod check-inu
android.clients.google.com	check-in, autentifikácia, overenie FRP, tap-to-pay
app-measurement.com	odosielanie udalostí Firebase / Google Analytics
play.googleapis.com	odosielanie logov Clearcut (/log, /log/batch)
mtalk.google.com	push socket FCM (trvalé spojenie)
gcm.googleapis.com	registrácia a stav FCM
nearby.googleapis.com	backend pre Nearby Share / Fast Pair
maps.googleapis.com	Places, statické mapy, geokódovanie
people-pa.googleapis.com	synchronizácia kontaktov a profilu
feedback-pa.googleapis.com	spätná väzba v aplikácii, snímky obrazovky
firebaseinstallations.googleapis.com	identifikátory inštalácií Firebase
safebrowsing.googleapis.com	dotazy na zoznamy hrozieb Safe Browsing (cez OHTTP)
ohttp-relay-safebrowsing-android.google.fastly-edge.com	relay Oblivious HTTP pred Safe Browsing

HOSTITEĽ	ÚLOHA
cryptauthvault.googleapis.com	úložisko kľúčov medzi zariadeniami (telefón ako bezpečnostný kľúč)
findmydevice-pa.googleapis.com	Nájsť moje zariadenie
androidpay-users-pa.googleapis.com	Google Wallet
googleads.g.doubleclick.net	žiadosti o reklamy, konverzné pingy

K zoznamu patria dve výhrady. Po prvé, mnohé koncové body sa skladajú za behu z konfigurácie ("https://" + authority + path), takže zoznam literálov je podhodnotený. Po druhé, názov hostiteľa v APK nie je paket: niektoré sú záložné, niektoré testovacie (*.sandbox.googleapis.com) a niektoré patria funkciám, ktorých sa používateľ nikdy nedotkne.

Tretia výhrada: jeden podsystém svoj cieľ zámerne skrýva. Safe Browsing používa Oblivious HTTP, takže SNI, ktoré filter vidí, patrí hostiteľovi Fastly (ohttp-relay-safebrowsing-android.google.fastly-edge.com), nie Googlu, a nahrávanie zo SafetyNet a Play Protect ide na safebrowsing.google.com/safebrowsing/uploads/android . Pre súkromie je to dobrá vlastnosť (relay vidí vašu IP, ale nie dotaz; Google vidí dotaz, ale nie vašu IP). Znamená to však aj, že blocklist zostavený z tejto tabuľky je neúplný. **ZISTENÉ** fykj.java:27–28 , fykf.java:21–22 , fysz.java:27

```
www.googleapis.com/auth/... (rozsahy, o ktoré žiadajú moduly GMS)
webhistory · plus.me · userinfo.email · userinfo.profile · peopleapi.readwrite
cryptauth · tapandpay · games · fitness · drive · photos.image.readonly
android_device_manager · kid.management · subscriptions · userlocation.reporting
```

Zoznam rozsahov OAuth je stručné vyjadrenie ambícií: história, profil, kontakty, kondícia, platby, fotky, hlásenie polohy, správa rodiny. Každý je dosiahnuteľný, len keď je prihlásený zodpovedajúci účet, ale presne v tejto situácii je väčšina používateľov.

08

Rozdiel GrapheneOS

Používateľov GrapheneOS nezaujíma, čo GMS robí vo všeobecnosti, ale čo robí aj tu.

SCHOPNOSŤ	ORIGINÁLNY AOSP	SANDBOX GRAPHENEOS
Čítanie IMEI / IMSI / sériového čísla SIM	áno, potichu (privilegované)	len s READ_PHONE_STATE udeleným používateľom
Čítanie MAC adresy, blízkych AP	áno (LOCAL_MAC_ADDRESS)	zamietnuté; náhodná MAC pre každú sieť
Čítanie logov, dropboxu, štatistík app-op	áno	zamietnuté
Prístup k štatistikám používania, overlay, zápis nastavení	áno	len cez špeciálny prístup v Nastaveniach
Zápis zabezpečených nastavení	áno	zamietnuté

SCHOPNOSŤ	ORIGINÁLNY AOSP	SANDBOX GRAPHENEOS
Vidieť všetky nainštalované aplikácie	áno	áno , QUERY_ALL_PACKAGES je normálne oprávnenie
Kontakty / SMS / história hovorov	áno	len ak používateľ udelí nebezpečné oprávnenie
Poloha (presná/na pozadí)	áno	len ak používateľ udelí; odvolateľné pre každú aplikáciu
Beh na pozadí	neobmedzený	štandardný standby aplikácií; používateľ môže obmedziť
Prístup k sieti	vždy	odvolateľný , INTERNET je v GrapheneOS nebezpečné
Jadro / skryté API	áno	blokové shimmi GmsCompat alebo zlyhá bez pádu

Čo sandbox odoberá

- 132 oprávnení signature/privileged, natrvalo.
- Systémové háčiky sa neuplatnia: privilegovaný stav telefónu, zabezpečené nastavenia a úložisko iných používateľov ostávajú mimo dosahu.
- Identitu „dôveryhodnej systémovej aplikácie“, ktorú používa Play Integrity a niektoré toky účtov.
- Tiché privilégia na pozadí; GMS súťaží o zdroje s ostatnými aplikáciami.

Čo neodoberá

- Rebrík identity check-inu (SSAID, logovacie ID, token).
- Clearcut, Measurement, Phenotype, FCM, reklamy a vodovod účtov.
- Zoznam nainštalovaných aplikácií (normálne oprávnenie).
- Všetko, čo udelíte za behu: kontakty, telefón, poloha, SMS.

Zhrnutie. Sandbox mení Play services zo systémovej súčasti s trvalým tichým prístupom na aplikáciu s rovnakými právami ako každá iná, ktorá by však chcela veľmi veľa. Súkromnou ju neurobí, ale je *podmienená súhlasom a odvolateľná*, čo je iná a užitočnejšia vlastnosť.

Dynamické načítavanie kódu je prepínač pre každú aplikáciu zvlášť a pre Play services je predvolene povolené. GrapheneOS má pre používateľom nainštalované aplikácie tri prepínače ochrany pred zneužitím: dynamické načítavanie kódu z pamäte, dynamické načítavanie kódu z úložiska a JIT vo WebView. Keď systém načítanie z úložiska odmietne, ukáže zablokovanú cestu. Play services sú používateľom nainštalovaná aplikácia, takže *predvolene nie je zablokované nič*: podľa poznámok k vydaniám GrapheneOS dal GrapheneOS výnimku aplikáciám, ktoré závisia od Play services, lebo starší systém modulov Dynamite ešte nie je celý nahradený split APK. Vypnutie načítavania z úložiska zastaví dve veci: engine DroidGuard (DexClassLoader nad app_dg_cache) a modulové APK načítavané ako súbory z app_chimera . Install-time splity **nezastaví**, tie sú bežné splity platformy. V trackeri GrapheneOS je hlásenie o páde (issue #8714): Play services zlyhali na SecurityException: DCL via storage pri načítaní voliteľného modulu z app_chimera na telefóne so zapnutým obmedzením. **ZISTENÉ** (kód, poznámky k vydaniám, issue) · **NETESTOVANÉ** (či sa Play services zhoršia elegantne)

Tri profily používateľa

PROFIL	ČO MÔŽE GOOGLE ASOCIOVAŤ	RIZIKO
Bez účtu Google, bez aplikácií Googlu	Odtlačok zariadenia + ID check-inu + SSAID, Android ID sa dá na požiadanie rotovať; hrubé metadáta siete (IP, časovanie).	NÍZKE pseudonymné zariadenie, nie osoba
Prihlásený účet Google, nič ďalšie neudelené	Všetko vyššie spojené so skutočnou identitou; používanie aplikácií cez Clearcut/Masurement; push token; história nákupov v Play.	STREDNÉ graf zariadení viazaný na účet
Účet + Kontakty + Telefón + Poloha udelené	Graf kontaktov, IMSI/SIM, signály histórie polohy, nainštalované aplikácie, nahrávanie cez Play Protect.	VYSOKÉ prakticky viditeľnosť ako na origináli Androidu

09

Odporúčania

Zoradené podľa toho, koľko každý krok prinesie na jednotku nepohodlia. Všetko v tejto sekcii je nastavenie alebo funkcia, ktorá existuje dnes; nič nevyžaduje patch. Značky ukazujú, o čo sa tvrdenie opiera: **ZISTENÉ** v APK alebo v používateľskej príručke GrapheneOS, **ODVODENÉ** z videného, **NETESTOVANÉ** nevyskúšané na reálnom zariadení.

Kde klikat'

Názvy menu nižšie boli overené na zariadení s GrapheneOS (build 2026102101, Android 17). Na iných buildoch sa obrazovky môžu mierne líšiť. Väčšina nastavení Play je za jedným rozcestníkom: **Settings > Apps > Sandboxed Google Play**. Obsahuje *Play services app info*, *Play services special permissions*, *Play Store app info*, *Android Auto*, *Google Settings* a v časti *Geolocation* prepínač *Reroute location requests to the OS*. **ZISTENÉ**

CIEĽ	CESTA	NASTAVIŤ NA
Odobrať Play services polohu	Sandboxed Google Play > Play services app info > Permissions > Location	<i>Don't allow</i> . Prepínač <i>Reroute location requests to the OS</i> nechať zapnutý.
Odpojiť Play services od siete	Play services app info > Permissions > Network	<i>Don't allow</i> ich úplne odpojí, ale rozbije push; väčšine ľudí to neodporúčam (bod 3)
Skryť identifikátory SIM pred check-inom	Play services app info > Permissions > Phone	<i>Don't allow</i>
Zablokovať prístup k SMS	Play services app info > Permissions > SMS	<i>Don't allow</i>
Skenovanie okolia	Play services app info > Permissions > Nearby devices	<i>Don't allow</i> , ak nepoužívate Fast Pair ani Quick Share
Autentifikácia operátora s ID zariadenia	Sandboxed Google Play > Play services special permissions > <i>Allow ICC authentication with device identifiers</i>	Vypnuté, ak to váš operátor nepotrebuje pre RCS

CIEĽ	CESTA	NASTAVIŤ NA
Kontakty bez ich odovzdania	Settings > Apps > (aplikácia) > Permissions > Contacts and accounts > <i>Contact Scopes</i>	Zapnúť namiesto <i>Allow</i>
Reklamné ID	Sandboxed Google Play > Google Settings > All services > Privacy & security > Ads	Ťuknúť na <i>Delete advertising ID</i> (alebo <i>Reset advertising ID</i>)
Hlásenie používania	Google Settings > All services > Privacy & security > Usage & diagnostics	Vypnúť
Overenie telefónneho čísla	Google Settings > All services > Privacy & security > Phone number verification	Skontrolovať; súvisí s tokom <code>firebase_pnv_preflight</code> v sekcii 05
Nahrávanie APK cez Play Protect	Play Store > profilový obrázok (vpravo hore) > Play Protect > Settings (vpravo hore)	Vypnúť <i>Improve harmful app detection</i> („Send unknown apps to Google“). <i>Scan apps with Play Protect</i> je samotné skenovanie. Kód má dva samostatné prepínače, <code>package_verifier_user_consent</code> pre skenovanie a <code>upload_apk_enable</code> pre nahrávanie. Sledovala sa len cesta nahrávania; skenovanie veľmi pravdepodobne posielala Googlu aspoň identifikátory aplikácií. ODVODENÉ
Stiahnutý kód (engine DroidGuard, moduly)	Play services app info > Exploit protection > <i>Dynamic code loading from storage</i>	Vypnúť len ak nepotrebuje Play Integrity, skenovanie QR kódov ani passkeys (bod 8). NETESTOVANÉ
Vlastný profil	Settings > Security & privacy > Privacy > Private space	Voliteľné: Play tam nainštalujte, len ak ho potrebuje málo aplikácií. Push sa po zamknutí zastaví.
Push zostane živý	Play services app info > App battery usage	<i>Unrestricted</i> (potrebné pre push)

Oprávnenia s viac než dvoma možnosťami: *Location* ponúka *Allow all the time*, *Allow only while using the app*, *Ask every time or allow when you share* a *Don't allow*, plus prepínač *Use precise location*. *Network*, *Phone*, *SMS* a *Nearby devices* majú len *Allow* alebo *Don't allow*.

1. Rozhodnite, na čo Play services potrebujete

Ak potrebujete bankové aplikácie, push upozornenia a nákupy v Play, nechajte ich a počítajte s tým, že check-in a Clearcut pobežia. Ak nie, odinštalujte ich. Neexistuje konfigurácia GMS, ktorá by bola zároveň funkčná a bez telemetrie; sú to tie isté cesty kódu.

2. Rozhodnite, ktorý profil dostane Play

Príručka GrapheneOS hovorí, že Play v sandboxe sa dá nainštalovať do sekundárneho používateľa, pracovného profilu alebo Private Space a môžu ho používať len aplikácie v tom profile **ZISTENÉ**. Android drží dáta balíkov, Android ID a zoznam balíkov pre každého používateľa osobitne, takže to, čo GMS vidí, je obmedzené na ten profil. **ODVODENÉ** Pomôže to, ak Play potrebuje málo aplikácií a zvyšok chcete mať mimo Googlu. Hlásenie to nezastaví: Play services v bežiacom profile sa Googlu ozývajú aj tak. Private Space sa po zamknutí zastaví, takže push dostanete len vtedy, keď je odomknutý. **ODVODENÉ** Najjednoduchšia cesta podľa príručky je len profil Owner, alebo druhý profil iba vtedy, keď chcete vybrať, ktoré aplikácie Play používajú **ZISTENÉ**. Ak je jedným z dôvodov push,

nainštalujte Play do profilu, ktorý používate denne. Oddelený profil stojí prepínanie alebo odomkykanie v bežnom používaní.

3. Prepínač siete: funguje, ale neodporúčam ho

GrapheneOS robí z **INTERNET** odvolateľné oprávnenie, v tejto analýze vedené ako nebezpečné. Vypínač je *Sandboxed Google Play > Play services app info > Permissions > Network > Don't allow*. Vypnutie siete pre Play services zastaví check-in, Clearcut, Measurement, FCM aj reklamy naraz. Cena je, že všetko, čo Play services používa, sa zasekne, takže ide o rozhodnutie pre profil, nie pre jednu aplikáciu. **ODVODENÉ**

Prečo by som to nerobil. Play services bez siete sú takmer na nič. Niečo funguje ďalej (napríklad čítanie QR kódu modulom, ktorý už v telefóne je), ale hlavný dôvod, prečo ich väčšina ľudí chce, sú push notifikácie, a tie idú cez spojenie s Googlom, ktoré ste práve prerušili. Bankové aplikácie, messengery a kuriéri prestanú upozorňovať, zlyhá Play Integrity aj prihlásenie, pokazia sa mapy a všetko ostatné, čo sa rozpráva s Googlom. Ak s tým súhlasíte, je lepšie Play services vôbec neinštalovať. Realistický spôsob, ako cenu znížiť, sú nastavenia oprávnení a profilov vyššie, nie tento prepínač.

4. Zamietnite oprávnenia, ktoré Googlu prinášajú najviac

- **Telefón.** Je to cesta k IMSI a sériovému číslu SIM v check-ine a kód ju podmieňuje presne týmto oprávnením (`bjtv.java:131,340,347`). Zamietnite, pokiaľ ho nepotrebuje váš operátor (príručka menuje RCS a TS.43). **ZISTENÉ**
- **Kontakty.** Namiesto oprávnenia použite **Contact Scopes**: aplikácia sa tvári, že prístup má, a nedostane nič, zápis je blokovaný. **ZISTENÉ** (príručka)
- **SMS / História hovorov.** Zamietnite, ak sa nepoužíva.
- **Poloha.** Ponechajte predvolené presmerovanie na geolokačnú službu GrapheneOS; potom Play services nepotrebujú oprávnenie polohy pre aplikácie, ktoré o ňu žiadajú. **ZISTENÉ** (príručka) Čo to znamená pre Google Maps, vysvetľuje nasledujúca časť.
- **Fyzická aktivita, Zariadenia v okolí.** Udeľte len pre funkcie, ktoré používate (zdieľanie polohy, Fast Pair).

Potrebuju Play services polohu? (a čo Google Maps)

Nie, predvolene nie. Príručka GrapheneOS hovorí, že aplikácie používajúce geolokáciu Google Play sú presmerované na vlastnú implementáciu GrapheneOS nad štandardnou službou polohy v OS a že na fungovanie polohy v týchto aplikáciách nemusíte Play services udeľovať žiadne oprávnenie ani meniť nastavenia. **ZISTENÉ** (príručka) Presmerovanie zapína prepínač *Sandboxed Google Play > Geolocation > Reroute location requests to the OS* a tá istá obrazovka má položku *Potential issues > Geolocation*, ktorá varuje, keď sú požiadavky presmerované, ale Play services stále majú oprávnenie Location. **ZISTENÉ** (na zariadení)

Google Maps má vlastné oprávnenie Location ako každá aplikácia. Keď Maps požiada API polohy z Play o pozíciu, odpovie služba z OS, takže Maps by mali fungovať aj po odobratí Location Play services. **ODVODENÉ** Zamestnanci a používatelia GrapheneOS píšú, že Mapy bežia aj bez Play services, niektorí hlásia sivú mapu, takže si to vyskúšajte. Oficiálna dokumentácia k tomu nie je. Čo bez neho prestane fungovať:

- **Zdieľanie polohy v Google Maps** beží vnútri Play services a vyžaduje Location *Allow all the time* plus oprávnenie Physical activity. **ZISTENÉ** (príručka)
- **Vlastná sieťová poloha Googlu** (tá, ktorá posiela dáta o Wi-Fi a bunkách Googlu) funguje, len ak vypnete presmerovanie, udeľte Location *Allow all the time* a Nearby devices a zapnete ju v *Google Location Accuracy*. Príručka ju neodporúča. **ZISTENÉ** (príručka)

- **Fast Pair a podobné funkcie Nearby** používajú skenovanie Bluetooth a viažu sa na oprávnenie Nearby devices.

ODVODENÉ

Cenou za presmerovanie je spotreba: ťahá viac batérie ako služba Googlu, pokiaľ nezapnete voliteľnú sieťovú polohu GrapheneOS v *Settings > Location > Location services > Network location* (zlepší ju Wi-Fi scanning v *Settings > Location > Location services > Wi-Fi scanning*). **ZISTENÉ** (príručka) Pri studenom štarte môžu Maps vnútri budovy dlhšie čakať na prvú pozíciu, kým nepríde signál zo satelitov. **NETESTOVANÉ**

5. Reklamné ID zmažte, nielen resetujte

Reset preruší spojenie so starými reklamnými profilmi, ale vytvorí nový. Zmazanie, ak ho nastavenia Googlu ponúkajú (manifest obsahuje `AdsIdentityCollapseSettingsDeleteAdIdActivity`), dá aplikáciám nulovú hodnotu. Identity check-inu sa to nedotkne. **ZISTENÉ** (komponent) · **NETESTOVANÉ** (správanie)

6. Vypnite hlásenie používania

Zostavenie obsahuje `usagereporting.settings.UsageReportingActivity` a `UsageReportingIntentService`. Vypnite „Používanie a diagnostika“ na obrazovke nastavení Googlu v sandboxovanej aplikácii Play. Je to jeden z mála prepínačov, ktoré Google dáva pred tok Clearcut. **ZISTENÉ** (komponenty) · **ODVODENÉ** (že prepínač ovláda odosielanie)

7. Vypnite nahrávanie cez Play Protect, ak akceptujete kompromis

Verify Apps zaraďuje do fronty samotné APK: `ApkUploadChimeraService` posiela intent `UPLOAD_APK` do `InternalApkUploadChimeraService`, ktorý drží frontu odosielania a opakuje ju pri zmene siete a zmene balíka. Ak inštalujete mimo Play, je to podstatné odhalenie; ak inštalujete len z Play, hodnota je nižšia, než vyzerá. **ZISTENÉ**

8. Ak nepotrebuje Play Integrity, zablokujte dynamické načítavanie kódu z úložiska

Settings > Apps > Play services > App info > Exploit protection > Dynamic code loading from storage. Je to jediná kontrola, ktorú má používateľ nad dvomi cestami kódu, o ktorých rozhoduje server: nad enginom DroidGuard a nad stiahnutými modulovými APK. Počítajte s tým, že prestanú fungovať výsledky Play Integrity, skeny Play Protect a funkcie postavené na moduloch (skenovanie QR kódov, rozpoznávanie textu, passkeys). Týka sa to aj výsledkov, ktoré dnes prechádzajú: DroidGuard generuje podklady pre basic a device integrity, takže aplikácia, ktorá dnes funguje, by mohla začať hlásiť chybu integrity. Strong integrity neprejde tak či tak (sekcia 05). **ZISTENÉ** (cesty) · **NETESTOVANÉ** (čo sa rozbije a presná cesta v menu)

9. Neprihlasujte sa, pokiaľ to funkcia nevyžaduje

Účet je priečka 04 rebríka. Bez neho je zariadenie pseudonym; s ním môže Clearcut pripájať `uploadAccount` k udalostiam.

10. Uprednostnite aplikácie, ktoré Play services nepotrebujú

Každá aplikácia, ktorá integruje Firebase, Maps, FCM alebo AdMob, je ďalší prúd do toho istého účtu. Nie je to test čistoty; je to priame zníženie počtu podsystémov vyššie, ktoré o vás majú čo povedať.

Čomu sa vyhnúť. Nepredpokladajte, že „bez účtu Google“ znamená „anonymne“, a nepredpokladajte, že vymazanie dát Play services vynuluje všetko: vynuluje ID check-inu, ale SSAID prežije až do továrenského resetu a metadáta o IP a časovaní ste nikdy nemohli vymazať vy. Neberte aplikácii beh na pozadí kvôli batérii, ak chcete push; príručka odporúča výnimku z optimalizácie batérie práve z tohto dôvodu.

10

Nad rámec nastavení: blokovanie a patche

Nastavenia končia tam, kde Google rozhodol, čo odhalí. Táto sekcia ide ďalej: filtrovanie siete, vypínanie komponentov, úprava APK a patch samotného GrapheneOS. Sú to inžinierske návrhy postavené na inventároch komponentov a hostiteľov z tohto reportu. **Žiaden z nich nebol vyskúšaný na zariadení.** Každý je ohodnotený prínosom pre súkromie, rozbitím funkcií, prácnosťou a údržbou, aby ste ich mohli porovnať.

Sekcie 10.2 a 10.3 neodporúčam. Sú tu preto, aby odpovedali na otázku „malo by to zmysel?“ a teoreticky premysleli, čo by to mohlo priniesť. Je to myšlienkový experiment, nie praktická rada. V praxi sú obe oveľa väčšia otrava, než sa zdá: všetko znovu podpisujete alebo si stavate vlastný OS, nesiete pripnuté certifikáty alebo vlastný kľúč verified boot a prácu opakujete pri každom vydaní. Prínos je oproti tomu malý.

10.1 Filtrovať sieť zvonka

Bez patchu, plne vratné a APK ostáva podpísané Googlom. Použite Private DNS s filtrujúcim resolverom alebo firewall cez VPN nainštalovaný len v profile s Play (VPN v Androide sú pre každého používateľa osobitne, takže profil Owner nie je dotknutý **ODVODENÉ**). Filter na úrovni DNS alebo SNI však vidí len názvy hostiteľov, nikdy cesty.

HOSTITEĽ	PODSYSTÉM	BLOKOVÁŤ?	DÔSLEDOK
app-measurement.com	Odosielanie analytiky Firebase ZISTENÉ	ÁNO	Analytika aplikácií prestane odchádzať. Aplikácie samotné fungujú.
googleads.g.doubleclick.net, pagead2.googlesyndication. com	Reklamy ZISTENÉ	ÁNO	Reklamy AdMob sa nenačítajú.
feedback-pa.googleapis.com	Spätná väzba v aplikácii	ÁNO	Spätná väzba a hlásenia chýb sa neodošlú.
people-pa.googleapis.com	Synchronizácia kontaktov a profilu	AK NEPOUŽÍVATE	Synchronizácia a vyhľadávanie kontaktov zlyhá.
play.googleapis.com	Clearcut <code>/log/batch</code> ZISTENÉ , ale aj iné API Play	ZDIEĽANÝ HOSTITEĽ	Blokovanie hostiteľa zastaví Clearcut a zároveň nesúvisiace funkcie Play. Odrezať len <code>/log</code> vyžaduje inšpekciu ciest.

HOSTITEĽ	PODSYSTÉM	BLOKOVÁŤ?	DÔSLEDOK
cloudconfig.googleapis.com	Príznyky Phenotype	RISKANTNÉ	Príznyky zamrznú na zastaraných hodnotách; moduly sa môžu správať zle. ODVODENÉ
firebaseinstallations.googleapis.com	Identifikátory inštalácií Firebase	NIE	Aplikácie na Firebase ho potrebujú na získanie tokenov FCM. ODVODENÉ
android.clients.google.com	Check-in, autentifikácia, FRP	NIE	Závisí od neho push aj prihlásenie. Je to aj koncový bod identity.
mtalk.google.com, gcm.googleapis.com	Socket FCM	NIE	Toto je push. Ponechať.
ohhttp-relay-safebrowsing-android.google.fastly-edge.com	Safe Browsing cez relay ZISTENÉ	NIE	Zablokovanie zastaví vyhľadávanie v Safe Browsing, čo môžete chcieť. Zároveň ukazuje problém: hostiteľ, ktorého by ste blokovali (safebrowsing.googleapis.com), nie je ten, ktorý ide po sieti.

Obmedzenia. Zostavenie obsahuje split Cronet (`split_CronetDynamite_installtime.apk` , 590 tried, `libcronet.155.0.8059.30.so`). Cronet je natívny zásobník, ale Java ho ovláda cez `org.chromium.net.CronetEngine` , takže hák na tomto API by požiadavky videl a hák na `URLConnection` nie. Filter podľa hostiteľa naozaj nevidí UDP: WebRTC (ICE/STUN), Cast (mDNS + UDP) a Nearby (WebTransport/QUIC v `libbetocore`). V natívnom kóde sa nenašli pevne zadané IP adresy. Safe Browsing svoj cieľ zámerne skrýva za relay Oblivious HTTP (ďalší riadok tabuľky).

ZISTENÉ

Ako zistiť, či to funguje. Spustíte PCAPdroid (záchyt cez VPN na zariadení, zaznamenáva SNI) v profile s Play na 24 hodín ako základ, aplikujte filter, zachytávajúce ďalších 24 hodín a porovnajte počty hostiteľov. To je chýbajúce upozornenie, ktoré tento report urobiť nemohol.

10.2 Upraviť APK (neodporúča sa, teoretická úvaha)

Neodporúča sa. Táto sekcia je teoretická úvaha o tom, či by prebalenie Play services malo zmysel a čo by mohlo priniesť. Nie je to praktická rada; práca a údržba sú oveľa väčšie, než sa zdá zo zoznamu nižšie.

Dá sa to vôbec? Technicky áno: je to obyčajné podpísané APK a nič, čo som v tejto analýze našiel, vám nebráni ho rozbaľiť, upraviť a znova podpísať (inštaláciu som neskúšal a nehľadal som kontrolu integrity vnútri aplikácie). Praktická otázka je, či výsledok ešte funguje, a tam to zlyháva. Prekážky podľa tejto analýzy:

PREKÁŽKA	PREČO NA NEJ ZÁLEŽÍ
Reťazec podpisov	Manifest deklaruje <code>sharedUserId="com.google.uid.shared"</code> a <code>isolatedSplits="true"</code> ZISTENÉ . Znovu podpísané GMS vyžaduje znovu podpísaný Play Store (a GSF, ak je prítomný) tým istým kľúčom, a všetkých 11 splitov. Inštalácia podpísaná Googlom sa musí najprv odstrániť, čím sa zmažú jej dáta. Obchod s aplikáciami GrapheneOS ju potom nemôže aktualizovať. ODVODENÉ

Pripnuté certifikáty	Kód Googlu má pripnutý názov balíka spolu s SHA-1 certifikátu Play services a Play Store (<code>gtub.java:8</code> , <code>DataService\$1.java:105</code> , používa sa pri nastavení zariadenia). Komponenty, ktoré tieto pripnutia kontrolujú, odmietnu zostavenie podpísané vašim kľúčom. ZISTENÉ (pripnutia) · ODVODENÉ (odmietnutie)
Dôvera v kľúč Googlu	Zostavenie obsahuje split <code>GoogleCertificates</code> a desiatky súborov čítajú podpisy balíkov alebo podpisové certifikáty (<code>signingInfo</code> v 38 súboroch, <code>.signatures</code> v 42, <code>Signature[]</code> v 57). Ostatné komponenty Googlu a Play Integrity budú pravdepodobne považovať iného podpisovateľa za nie-Googlu. ZISTENÉ (kontroly) · ODVODENÉ (dôsledky)
Kompatibilná vrstva	Žiaden súbor v dekompilovanom kóde nespomína GrapheneOS ani GmsCompat, čo potvrdzuje, že aplikácia je originálna ZISTENÉ . Vrstva na strane OS ju musí rozpoznať; nezistoval som, či sa opiera o podpisovateľa. Ak áno, potrebujete aj patch OS. NETESTOVANÉ
Vzdialené príznaky	Predvolené správanie sa číta cez vyhľadanie príznaku, napríklad URL Clearcut v <code>jwgm.java:12</code> (<code>a.h(1, "batch_server_url", ...)</code>). Phenotype môže predvolené hodnoty prepísať pri ďalšej synchronizácii, takže upraviť predvolenú hodnotu nestačí; treba patchnúť miesto volania alebo zastaviť synchronizáciu. ZISTENÉ
Zmeny obfuskácie	Triedy sú premenované (<code>jwgm</code> , <code>bjtv</code> , ...) a mená sú špecifické pre zostavenie, takže patch v smali treba nájsť znova v každom vydaní. Verejné názvy v manifeste sú stabilné. ODVODENÉ
Integrita	Upravené Play services pravdepodobne znížia výsledky atestácie, ktoré kontrolujú bankové aplikácie, a tie sú už v GrapheneOS obmedzené. ODVODENÉ

Možnosti, od najmenej po najviac invazívnu

- 1. Patch len manifestu.** Nastavte `android:enabled="false"` na telemetrických a reklamných komponentoch (napríklad `clearcut.uploader.QosUploaderService` , brokery `measurement.*` , `usagereporting.service.UsageReportingIntentService` , `ads.AdRequestBrokerService`) binárnym editorom manifestu, DEX nechajte nedotknutý, zipalign a potom podpíšte každý split vlastným kľúčom. Vyhnete sa opätovnému zostaveniu 195 MB APK a je to jediná možnosť, ktorej patch prežije zmeny verzií, lebo názvy sú verejné. Nechajte na pokoji `checkin.CheckinService` , služby FCM a účtov a `droidguard.*` : závisí od nich push, prihlásenie a atestácia. Play services navyše za behu samy prepínajú stav svojich komponentov (`setComponentEnabledSetting` v `brtr.java:48` , `setComponentEnabledSettings` v `gwyn.java:63` a ďalší dvaja volajúci), takže vypnutý komponent sa môže znova zapnúť a aktualizácia môže zoznam vynulovať. **ZISTENÉ** (názvy a volania) · **NETESTOVANÉ**
- 2. Patch smali v metódach odosielania.** Zrušiť miesta volania uploaderov. Funguje, kým ich ďalšie vydanie nepremenuje. Neodporúča sa.
- 3. Prepísanie príznakov.** Dodať vlastné hodnoty Phenotype. Potrebuje to server alebo zachytávanie a je to väčší projekt než samotný problém.
- 4. Iná implementácia (v štýle microG).** Samostatný projekt s iným modelom dôvery, ktorý sa vzdáva binárok Googlu úplne. Mimo rozsahu tohto reportu.

Prijmú servery Googlu upravené zostavenie? (atestácia)

Toto je hlavné riziko prebaľovania. **Nikto nevie sľúbiť, že to bude fungovať, a táto analýza to nedokáže rozhodnúť.** Čo je známe a čo nie:

- Známe.** Play services stiahnu engine DroidGuard (APK, kapitola 11) z `gstatic.com/droidguard/` , pri každej požiadavke mu podajú program, spustia ho vnútri procesu Play services a výsledok pošlú Googlu. Bajtkód aj

posúdenie na serveri určuje Google a môže ich kedykoľvek zmeniť. Play services tieto výsledky využívajú aj samy (token reklám, overenie telefónneho čísla, hlásenia Nájst moje zariadenie). **ZISTENÉ**

- **Neznáme.** Či program z požiadavky meria podpis alebo kód hostiteľskej aplikácie. Samotný engine sa podarilo získať a prečítať (kapitola 11), ale program, ktorý vykonáva, prichádza s každou požiadavkou a zachytený nebol.

NEZNÁME

- **Opodstatnená obava.** Systém určený na odhaľovanie manipulovaných prostredí má zjavný dôvod pozeráť sa na aplikáciu, v ktorej beží, takže znovu podpísané zostavenie pravdepodobne dopadne horšie než originál. Je to odhad z účelu mechanizmu, nie pozorovanie. **ODVODENÉ**

Čo by zlý výsledok pravdepodobne stál, od najmenej po najviac vážne:

OBLASŤ	PRAVDEPODOBNÝ ÚČINOK
Aplikácie volajúce Play Integrity	Výsledok kontroly zariadenia dostane aplikácia zo serverov Googlu. V GrapheneOS sa dnes očakáva, že prejde len základná a device integrita, ako uvádza sekcia 05, takže upravené zostavenie môže prísť najmä o ne. Aplikácie, ktoré už vyžadujú device alebo strong, zlyhajú tak či tak. ODVODENÉ
Vlastné funkcie Play services	Token reklám, overenie telefónneho čísla a crowdsourcované hlásenia Nájst moje zariadenie môžu zlyhať alebo sa preskočiť. Sú to najmenej dôležité veci. ODVODENÉ
Prihlásenie a účet	Hodnotenie rizika na serveri môže vyžadovať ďalšie overenie alebo zariadenie obmedziť. Neviem o dôkaze ani jedným smerom. Na akýkoľvek pokus použijete účet na jedno použitie. NEZNÁME
Play Store	Musel by byť tiež znovu podpísaný. Či by ešte sťahoval, aktualizoval a licencoval aplikácie, je neotestované. NETESTOVANÉ

Ako to lacno zistiť. V jednorazovom profile s účtom na jedno použitie nainštalujte originálne Play v sandboxe a spustíte aplikáciu na kontrolu integrity; zapíšte si výsledky, doručovanie push, prihlásenie a jednu aplikáciu závislú od Play. Potom ho nahraďte znovu podpísaným zostavením a zopakujte. Rozdiel je vaša odpoveď. Počas testu neblokujte `gstatic.com`: servíruje bajtkód DroidGuard.

Pravidlo rozhodnutia. Každá možnosť, ktorá necháva podpis Googlu nedotknutý (sekcie 09, 10.1 a 10.3 A), nenesie z pohľadu aplikácie žiadne ďalšie riziko pre atestáciu, lebo Google vidí rovnakú podpísanú binárku ako vždy. Prebaľovanie je jediná možnosť, ktorá mení to, čo Google o sebe môže vidieť. Ak vám na atestácii záleží, je to najsilnejší argument pre cestu cez OS namiesto cesty cez APK.

Ak to urobíte, uchovávajte podpisový kľúč ako vydavateľský kľúč: zdieľané UID znamená, že vlastník kľúča riadi, čo beží s dátami Play services. Zostavujte automaticky z podpísaného súboru upstreamu po každej aktualizácii a výsledok nikdy nedávajte ďalším ľuďom.

10.3 Patchnúť GrapheneOS (neodporúča sa, teoretická úvaha)

Neodporúča sa. Rovnako ako 10.2 je to myšlienkový experiment o tom, čo by mohla priniesť zmena na úrovni OS, nie rada, aby ste si stavali vlastný GrapheneOS. Potrebovali by ste vlastný kľúč verified boot, prišli by ste o oficiálne aktualizácie cez vzduch a patche by ste museli niesť cez každé vydanie.

OS je miesto, kde podpis ostáva nedotknutý a oprava prežije aktualizácie GMS, ale musíte si zostaviť a podpísať vlastný GrapheneOS, čo znamená vlastný kľúč verified boot a žiadne oficiálne aktualizácie cez vzduch. Nezistoval som, ktoré z týchto nápadov už upstream existujú; overte to a pred zostavovaním otvorte issue.

NÁVRH	KDE	ÚČINOK	PRÁCNOSŤ / RIZIKO
A. Zoznam povolených domén pre aplikáciu („sieť len pre push“)	Sieťový zásobník, podľa UID aplikácie: eBPF hák na sockety plus zhoda podľa resolvera/SNI	Povoliť check-in, autentifikáciu a FCM; zahodiť Clearcut, Measurement, reklamy a Phenotype. Na rozdiel od VPN v používateľskom priestore je predvolene zapnutý a aplikácia ho nemôže obísť.	Vysoká. Zdieľaní hostitelia (play.googleapis.com) a ECH ho obmedzujú na názvy hostiteľov.
B. Rozsahy balíkov	Filter viditeľnosti balíkov v správcovi balíkov	QUERY_ALL_PACKAGES je normálne oprávnenie ZISTENÉ , takže zoznam aplikácií je dnes zadarmo a 122 dekompilovaných súborov sa pýta na balíky alebo intenty. Zaobchádzať s GMS, akoby ho nemalo, v štýle Contact Scopes.	Stredná. Play Protect a Nearby prídu o časť viditeľnosti.
C. Reset identity zariadenia	Nastavenia plus úložisko Android ID	Jedna akcia rotuje Android ID pre podpisový kľúč Googlu v profile a vymaže súbory check-inu. Dnes vymazanie dát aplikácie resetuje ID check-inu, ale nie SSAID ODVODENÉ .	Stredná. Vynúti novú registráciu push a opätovné získanie dôvery účtu.
D. Filter URL vnútri procesu	Háčky kompatibilnej vrstvy na hranici HTTP v Jave	Pravidlá na úrovni ciest bez prelomenia TLS.	Stredná a krehká: Cronet treba háčikovať na CronetEngine a UDP (WebRTC, Cast, Nearby QUIC) je mimo dosahu.

Moje poradie: A, potom B, potom C a nakoniec D kvôli Cronet.

10.4 Ktorú vybrať

MOŽNOSŤ	PRÍNOS PRE SÚKROMIE	ROZBITIE	PRÁCNOSŤ	ÚDRŽBA
Nastavenia a oprávnenia (09)	Stredný	Nízke	Minúty	Žiadna
Prepínač siete (neodporúča sa: rozbije push)	Plný, offline	Všetko z Play	Sekundy	Žiadna

MOŽNOSŤ	PRÍNOS PRE SÚKROMIE	ROZBITIE	PRÁCNOSŤ	ÚDRŽBA
Filter DNS / VPN (10.1)	Stredný až vysoký	Nízke až stredné	Hodiny	Údržba zoznamu
APK s upraveným manifestom (10.2), teória	Stredný	Stredné až vysoké	Dni	Každé vydanie
Patch OS (10.3), teória	Vysoký	Nízke	Týždne	Každé vydanie OS

Odporúčaná cesta. Pre všetkých: sekcia 09, s Play v profile, ktorý používate denne, alebo vo vlastnom, ak ho potrebuje málo aplikácií. Ak chcete push bez väčšiny telemetrie: pridajte 10.1 v tom profile a zmerajte to. Sekcie 10.2 a 10.3 sú teória, nie odporúčanie. Ak chcete jednu aj tak preskúmať, na papieri je najmenej zlá 10.3 A, lebo nemení kód Googlu. Prebaľovanie APK preskočte a najprv si prečítajte jeho diskusiu o atestácii.

Stiahnutý kód: moduly

APK v repozitári nie je všetok kód, ktorý Play services spúšťajú. Google po inštalácii posiela do telefónu ďalšie modulové APK, ktoré sa neobjavia v žiadnom verejnom artefakte. Z telefónu sa bez rootu prečítať nedajú, preto som ich pre tento report vytiahol z rootnutého testovacieho emulátora s Androidom 13 a Play services 26.41, dekompiloval a každý súbor overil proti SHA-256, ktoré si zaznamenal Play. Nepochádzajú z GrapheneOS, ale sú to moduly, ktoré Google doručuje Play services všeobecne, takže ukazujú, čo Play services naozaj spúšťajú.

24	19	0	7
ZÍSKANÝCH MODULOVÝCH APK	RÔZNYCH BUILDŮV MODULOV	OPRÁVNENÍ, KTORÉ ŽIADAJÚ	GENERÁCIÍ MODULOV NA JEDNOM ZARIADENÍ

Čo znamená „dynamické načítavanie“: neexistuje jedno Play services

Je lákavé predstaviť si „Play services“ ako jeden komponent, ktorý by sa dal raz zaudítovať a potom pre všetkých rovnako dôverovať alebo nedôverovať. Takto to nefunguje. APK je hositeľ. Po inštalácii pošle Googlu opis zariadenia, na ktorom beží, a odpoveď určí, aké moduly má toto zariadenie mať a v akej verzii. Aplikácia si ich potom sama stiahne (cez Play Store) a sama aktualizuje, bez vášho zásahu a bez nového vydania Play services. Požiadavka nesie Android ID zariadenia a token pre každý prihlásený účet Google. Na testovacom emulátore sa plán modulov zmenil, keď sa zariadenie zaregistrovalo znova pod novou identitou, a jedno zariadenie skončilo so siedmimi generáciami modulov vedľa seba. [ZISTENÉ](#) (sekcia 11 nižšie) · [ODVODENÉ](#) (podľa čoho server rozhoduje)

Skoršie práce idú tým istým smerom. Analýza DroidGuardu v službe SafetyNet (predchodca Play Integrity) od Romaina Thomasa (SSTIC 2022) zistila, že Play services stiahnu najnovší DroidGuard, ak ho ešte nemajú, a ten si potom od Googlu stiahne bajtkód, ktorý je „unikátny pre každú atestačnú požiadavku“. Doug Leith z Trinity College Dublin (2025) pozoroval, že podsystém Experiments v Play services sťahuje nastavenia pre jednotlivé aplikácie spolu s tokenmi experimentov, takže Google môže zmenu najprv nasadiť malej podskupine používateľov, a medzi spojeniami, ktoré posielajú Android ID Googlu, uvádza `play-fe.googleapis.com/fdfe/moduleDelivery`. GrapheneOS samo muselo svoj runtime naučiť načítavať moduly Dynamite z Play services cez file descriptor (platform_art, pull request 1). Pokiaľ viem, nikto nezverejnil pohľad na samotné doručené moduly, s tabuľkou sieťových cieľov po moduloch a analýzou toho, na čo by sa dali cieľiť. To tento report dopĺňa.

Čo z toho plyní pre čítanie tohto reportu:

- **Dva telefóny s rovnakou verziou Play services môžu spúšťať rôzny kód.** Hostiteľské APK je rovnaké; moduly, ich verzie a zapnuté funkcie sa môžu líšiť podľa zariadenia, jeho účtu Google a rozhodnutí Googlu o postupnom nasadzovaní.
- **Audit je momentka.** To, čo som dekompiloval, je hostiteľské APK plus moduly, ktoré jedno testovacie zariadenie dostalo v jeden deň. Ukazuje, aký kód Google doručuje; nevie dokázať, čo dnes spúšťa iný telefón, vrátane vášho. Overiť váš telefón vyžaduje vytiahnuť z neho moduly (root) a porovnať ich hashe.
- **Google to môže zmeniť bez toho, aby to niekomu povedal.** Modul sa dá na serveri vymeniť alebo pridať. Jedinou kontrolou v telefóne je, že ho podpísal Google.
- **Sandbox stále platí.** V GrapheneOS dostane všetko, čo príde, len oprávnenia, ktoré ste dali Play services (alebo aplikácii, ktorá hostí mapu či reklamu).

Ako modul príde

Play services si cez súkromné rozhranie Binder (`BIND_PLAY_MODULE_SERVICE`) vypýtajú modul od Play Store, ten APK stiahne a Play services ho zapíše do `app_chimera/m/<n>/` vo vlastnom dátovom adresári. Ktoré moduly telefón dostane, určuje konfiguračný súbor od servera (`current_config.fb` , FlatBuffer s magic `CHMR`). Na emulátore sa tento súbor zmenil, až keď sa zariadenie zaregistrovalo nanovo pod novou identitou: samotná zmena build fingerprintu nespravila nič, čerstvá registrácia ako Pixel 8 priniesla iný plán modulov. [ZISTENÉ](#) (logy zo zariadenia, veľkosť súboru 578 156 → 576 380 bajtov) · [ODVODENÉ](#) (že plán závisí od registrácie zariadenia, nie od hlásenej hardvérovej identity)

Verzie modulov nezávisia od verzie Play services. Jeden emulátor mal naraz sedem generácií modulov (buildy od 23.11 po 26.41), šesť modulov v dvoch verziách a aktuálna konfigurácia odkazovala len na deväť z pätnástich súborov na disku. Staré kópie sa pri príchode novej nemažú. [ZISTENÉ](#)

Čo moduly obsahujú

MODUL	ČO TO JE	POZNÁMKA K SÚKROMIU
<code>MlkitBarcodeUi</code>	Obrazovka skenera QR a čiarových kódov (CameraX, MediaPipe)	Kameru používa na základe oprávnenia Play services
<code>VisionBarcode</code>	Dekodér čiarových kódov (Barhopper v3, tri modely TFLite)	V zariadení
<code>MlkitOcrCommon</code> · <code>VisionOcr</code> · <code>VisionOcrDynamite</code>	Pipeline a modely na rozpoznávanie textu	V zariadení
<code>TfliteDynamiteDynamite</code>	Runtime TensorFlow Lite / LiteRT, ktorý ostatné ML moduly volajú ako službu Binder	Volajúcim odovzdáva surové natívne ukazovatele
<code>MapsCoreDynamite</code>	Vykresľovanie máp Google Maps SDK pre aplikácie s vloženou mapou	Dlaždice máp od Googlu
<code>AdsFdrDynamite</code>	Vykresľovacia vrstva Google Mobile Ads (8 758 tried, <code>ads.nonagon.*</code> , generátor signálov)	Požiadavky na reklamy a signály pre aplikácie s AdMob
<code>IdentityCredentialsPlatform</code>	Poskytovateľ Android Credential Manager: passkeys, import a export prihlasovacích údajov, vekové signály; obsahuje interpreter WebAssembly napísaný v Jave	Pracuje s passkeys a prihlasovacími údajmi
<code>PlayCloudSearch</code>	gRPC server za vyhľadávacím poľom launcheru, komunikuje s <code>playgateway-pa.googleapis.com</code>	Dotazy napísané v launcheri, ak sa naň launcher pripojí
<code>Appsearch</code>	Vlastné úložisko AppSearch v Play services (backend pre Jetpack <code>androidx.appsearch</code>) so vstavanými indexermi nainštalovaných aplikácií a hudby z MediaStore	Index zostáva v zariadení; telemetria uvádza volajúcu aplikáciu

Všetky riadky [ZISTENÉ](#) v dekompilovaných moduloch.

Čo majú spoločné

- **Žiadne vlastné oprávnenia a cudzí proces.** Ani jedno z 24 APK nedeklaruje jediné `uses-permission` . Používajú oprávnenia procesu, ktorý ich načíta, a „host“ znamená pri dvoch typoch modulov niečo iné. Moduly *Chimera* (obrazovka skenera, OCR, passkeys, vyhľadávanie v launcheri, Appsearch) bežia vnútri Play services s ich UID, všetkými oprávneniami, ktoré ste im dali, a ich sieťovým prístupom. Keď Play services odrežete od siete,

odrežete aj tieto moduly. Moduly *Dynamite* (vykresľovanie máp a reklám, Cronet, Analytics) si načíta aplikácia s Google SDK do *vlastného* procesu. Mapa v bankovej aplikácii tak spúšťa stiahnutý kód Googlu s oprávneniami a sieťou tej bankovej aplikácie a ak Play services zakážete prístup na sieť, tento kód sa na sieť dostane aj tak. Obrázok oprávnení neukáže ani jedno. Či taký modul pred spustením potrebuje od Play services ešte niečo (napríklad pri dlaždicach mapy), som netestoval. Na testovacom emulátore boli Play services privilegovaná systémová aplikácia so stovkami oprávnení; v sandboxe na GrapheneOS majú len to, čo ste im povolili. **ZISTENÉ** (manifesty; `DynamiteLoaderV2.java`) · **ODVODENÉ** (GrapheneOS)

- **Bez aktivít a služieb vo vlastnom manifeste.** Vstupné body sú prázdne triedy v hlavnom APK a skutočný kód dodá modul. Skenovanie QR spustené zo zástupnej obrazovky v Play services spúšťa kód zo samostatne podpísaného APK. **ZISTENÉ**
- **Obyčajný kód.** Bežný DEX a bežné natívne knižnice s normálnym zabezpečením, načítané obvyklým spôsobom. **ZISTENÉ**
- **Dva podpisové kľúče a jedna pečiatka zdroja.** Moduly `optional` podpisuje všeobecný kľúč Googlu pre Android, moduly `policy` samostatný kľúč `gmscore_modules_percy`. Všetky, aj engine DroidGuard, nesú rovnaký certifikát source stamp. **ZISTENÉ**
- **Moduly volajú jeden druhý.** Moduly OCR a čiarových kódov sa pripájajú na modul TFLite a cez Binder od neho dostávajú surové tabuľky natívnych funkcií. Hranica modulu je hranica balenia, nie bezpečnostná hranica. **ZISTENÉ**

Appsearch: index vašich aplikácií a hudby v zariadení

Modul Appsearch (9 028 tried, natívny engine `libicing_icu_with_reversejni.so`) je úložná služba za Jetpackovým `androidx.appsearch` pre aplikácie, ktoré používajú úložisko Play services. Nie je to platformový AppSearch Androidu. Okrem toho, čo doň vložia aplikácie, má dva vlastné indexery:

- **Nainštalované aplikácie.** Z každej spustiteľnej aplikácie vznikne dokument `MobileApplication` s názvom, popisom, ikonou, SHA-256 podpisového certifikátu, triedou launchera a časom aktualizácie, obnovovaný pri každej zmene balíkov. Riadi ho flag `enable_appsindexer`, predvolene vypnutý; Google ho môže zapnúť na diaľku. **ZISTENÉ** `m58/xs.java:32`, `m58/kgb.java:15`
- **Hudba.** Albumy, interpreti, playlisty a skladby z `MediaStore.Audio.Media.EXTERNAL_CONTENT_URI` sa ukladajú ako vyhľadávateľné dokumenty. Potrebuje to oprávnenie na zvuk, ktoré by ste na GrapheneOS museli Play services povoliť. Predvolene vypnuté. **ZISTENÉ** `m58/aga.java:64`

Index leží v súkromnom úložisku Play services bez šifrovania na úrovni aplikácie, chráni ho teda len sandbox a šifrovanie súborov. Nič v module index neodosiela. Štatistiky používania však idú do Clearcut: sedem typov udalostí, ktoré nesú názov balíka volajúcej aplikácie v čitateľnej podobe a názov databázy ako skrátený hash MD5. Jediné sťahovanie je asi 31,6 MB slovník ICU na delenie slov z `gstatic.com`, len na Wi-Fi počas nabíjania. Indexer kontaktov a indexer otvárania aplikácií sú v nastaveniach pomenované, ale ich kód v tomto builde nie je, a vektorový index v engine existuje, no je vypnutý a nemá model. **ZISTENÉ** `m58/aig.java:15`, `m58/ahh.java:136-146`, `m58/kgb.java:20` · **NEZNÁME** (ktoré funkcie Play services doň zapisujú)

Engine DroidGuard, prečítaný

Engine (`com.google.ccc.abuse.droidguard`, 459 KB) tvorí Java vrstva s 24 triedami a natívny interpreter bez symbolov. Je stavaný tak, aby sa ťažko čítal. Exportované `initNative` a `closeNative` sú návody, ktoré sa hneď vrátia; `JNI_OnLoad` na zásobníku dešifruje sedem mien metód a zaregistruje skutočné funkcie. Interpreter je jedna sploštená funkcia s 237 KB a dvomi bankami po 256 registroch a kód na troch miestach mapuje pamäť, do ktorej sa dá zapisovať aj ju spúšťať. **ZISTENÉ** (offsety `0xbcc4`, `0xdf97`, `0x16070`)

Čo lokálny kód meria bez pomoci servera:

- **Odtlačok GPU („droidguasso“).** Prečíta výrobcu, renderer a verziu GL, vykreslí dva shadery nasadené 48-bajtovou výzvou zo servera, zahashuje výsledok 32×32 a zahashuje každý grafický ovládač v `/vendor/lib/egl` a `/system/lib/egl`. ZISTENÉ
- **Pohybové senzory.** Priemer, odchýlka, minimum a maximum pre dva senzory plus surové vzorky, s frekvenciou a dĺžkou, ktoré určí server. ZISTENÉ
- **Dotyky,** najviac 32, ak ich vkladajúca aplikácia podá. ZISTENÉ · ODVODENÉ (len pri použití v aplikácii; samotné Play services ich nepodávajú)
- **Údaje z Play services.** Objekt, ktorý Play services enginu podajú (`RuntimeApi`), vracia Android ID, informáciu, či je zapnuté overovanie aplikácií, a zoznam škodlivých aplikácií z Play Protect s názvom balíka, hashom a kategóriou. ZISTENÉ `droidguard/loader/RuntimeApi.java:39–110`

Čo engine dosiahne ďalej, určuje program, ktorý dostane. Natívny kód obsahuje surový stub `syscall` a trampolínu na volanie ľubovoľnej funkcie, ktorých adresy dostane interpretér, a funkcie hľadá podľa mien z programu. 64 importovaných symbolov teda neobmedzuje, čo dokáže prečítať: program si môže vypýtať akýkoľvek súbor alebo systémové volanie, ktoré smie urobiť proces Play services. Kontroly rootu, Fridy či systémových vlastností v lokálnom kóde nie sú, prichádzajú teda s programom z požiadavky. Vlastný sieťový kód engine nemá; výstup odosielajú Play services. ZISTENÉ (0x6bbc, 0x6b4c) · NEZNÁME (čo si pýta skutočný program)

Tri roky install-time modulov

Emulátor najprv bežal so starými Play services 23.18 zo systémového obrazu, takže osem install-time modulov Dynamite existuje vo verzii 23.18 aj 26.40. Rozdiel ukazuje, kam smeruje merací kód Googlu:

- **Measurement (Firebase / Google Analytics)** pribudol Consent Mode v2 (`ad_user_data`, `ad_personalization`), spracovanie DMA a TCF, nahrávanie pre server-side tagging (dáta môžu ísť na server, ktorý prevádzkuje vývojár aplikácie) a hlásenie konverzií Privacy Sandbox na `google-analytics.com`. Požiadavka na odložený deep link stále nesie reklamné ID. Ani jedna verzia nečíta App Set ID ani Android ID. ZISTENÉ
- **Ads** prešli z nulového kódu Privacy Sandbox na Topics (predvolene vypnuté) a Attribution Reporting (predvolene zapnuté) a v oboch verziách čítajú MD5 hash Android ID. Kód Protected Audience tam nie je. Stiahnutý modul na vykresľovanie reklám (`AdsFdrDynamite`) ide ďalej: keď zlyhá alebo vyprší zistenie reklamného ID, vloží do požiadavky na reklamu *nehashované* Android ID ako `pdid`. Zmazanie reklamného ID to nespustí (zistenie vtedy uspeje a vráti nuly); spustí to zistenie, ktoré skončí chybou. Reklamné ID posiela spolu s príznakom obmedzeného sledovania, drží si vlastné UUID `paidv1_id` a číta build fingerprint, kód operátora, typ siete a pohybové senzory. Keďže beží v aplikácii, ktorá reklamu ukazuje, číta Android ID tej aplikácie. ZISTENÉ `ads/nonagon/signals/gmscore/c.java:86`, `d.java:31–46`
- **Maps** stále posielajú bootstrap s logovacím ID a krajinou SIM. ZISTENÉ
- **GoogleCertificates** je zoznam dôveryhodných podpisovateľov aplikácií, nie piny TLS. Narástol z 1 635 na 2 017 odtlačkov. **Cronet** sa posunul z Chromia 113 na 155. ZISTENÉ
- **Podpisový kľúč** sa menil: moduly 23.18 podpisuje stredný kľúč z trojčlennej línie v APK 26.40. ZISTENÉ

S kým sa ktorý modul rozpráva

MODUL	BEŽÍ V	KAM	ČO POSIELA
AdsFdrDynamite	aplikácii, ktorá ukazuje reklamu	reklamné servery Googlu a akýkoľvek hositeľ, ktorého uvedie odpoveď s reklamou : URL zobrazení, kliknutí, videa a viditeľnosti, kreatíva vo WebView, mediačné SDK tretích strán	reklamné ID s príznakom obmedzeného sledovania (aj keď je obmedzené), App Set ID, surové Android ID, keď zlyhá zistenie reklamného ID, model a build fingerprint, operátor, poloha, ak ju aplikácia podá
MapsCoreDynamite / MapsDynamite	aplikácii, ktorá ukazuje mapu	len Google	dlaždice pre zobrazený výrez, API kľúč aplikácie, jej balík a odtlačok podpisového certifikátu; starší bootstrap s logovacím ID a krajinou SIM
PlayCloudSearch	Play services	<code>playgateway-pa.googleapis.com</code> (gRPC), obrázkové servery Googlu	čo napíšete do vyhľadávania v launcheri, jazyk, OAuth token. Každé volanie je predvolene vypnuté a zapínajú ho flagy zo servera
DynamiteModulesC	volajúcej aplikácii	Google Analytics, Tag Manager, Firebase; akýkoľvek hositeľ , ktorého vývojár zadá do kontajnera Tag Manager alebo konfigurácie Firebase	klientske ID Analytics a reklamné ID v hitoch Analytics; čokoľvek, čo definuje kontajner vývojára
MeasurementDynamite	volajúcej aplikácii	<code>app-measurement.com</code> cez Play services, <code>googleadservices.com</code> ; v 26.40 aj server-side tagging na serveri vývojára	udalosti aplikácie, ID inštancie aplikácie, reklamné ID v požiadavke na odložený deep link
CronetDynamite	volajúcej aplikácii	kamkoľvek, kam volajúci požiada	predvolený User-Agent uvádza aplikáciu, jej verziu, model telefónu a build
Appsearch	Play services	<code>gstatic.com</code> (slovník), Clearcut	štatistiky používania s názvom volajúcej aplikácie
ML a vision moduly, TFLite, passkeys, DynamiteModulesA	Play services alebo volajúcej aplikácii	vlastné spojenia žiadne; Clearcut cez Play services	počítadlá používania; vision logger povoľuje Android ID. Odosiela sa len so zapnutým <i>Usage & diagnostics</i>
Engine DroidGuard	Play services	vlastné spojenia žiadne; výsledok posielajú Play services	čokoľvek, čo zozbiera program z požiadavky, vrátane Android ID, ktoré mu podajú Play services
GoogleCertificates, DynamiteLoader	n/a	žiadne	nič

Všetky riadky **ZISTENÉ** v dekompilovaných moduloch, súbor a riadok v mojich pracovných poznámkach (nezverejnené). Každé spojenie navyše prezradí IP adresu. Ktorých hositeľov tretích strán uvádzajú skutočné odpovede s reklamou, sa bez zachytenia živej prevádzky zistiť nedá. **NEZNÁME**

Dve z toho si zaslúžia vetu navyše. Reklamný modul zavolá každú adresu, ktorú mu reklamný server podá (napríklad na započítanie zobrazenia alebo kliknutia), nasleduje presmerovania a hositeľa nijako nekontroluje. Reklama v aplikácii tak môže prinútiť telefón kontaktovať ľubovoľný počet inzerentov a meracích firiem. **ZISTENÉ**

ads/internal/util/client/s.java:88–131 Moduley pre mapy a reklamy navyše bežia vnútri aplikácie, ktorá ich používa, takže poloha, sieť a identifikátory, ktoré vidia, sú tie, ktoré dostala tá aplikácia.

Môže Google poslať inému človeku iný kód?

Technicky áno. Kód sa do telefónu dostane bez aktualizácie Play services piatimi cestami: Chimera moduley cez Play Store, on-demand Dynamite moduley (rovnaká cesta), engine DroidGuard, program DroidGuard posielaný pri každej kontrole a spiaci kód, ktorý zapne vzdialený flag. [ZISTENÉ](#)

- **Plán modulov je vzdialený flag.** Ktoré moduley má telefón mať, prichádza ako Phenotype flag `module_set_list` (alebo v odpovedi check-inu). Požiadavka Phenotype nesie Android ID, pseudonymné ID zariadenia od Googlu, profil buildu a OAuth token každého prihláseného účtu, takže server môže jednému zariadeniu alebo účtu odpovedať inak než všetkým ostatným. [ZISTENÉ](#) `bkaq.java:26`, `fvdd.java:70–77`, `fvet.java:9` · [ODVODENÉ](#) (odpovede pre konkrétne zariadenie)
- **Sťahovanie je viazané na zariadenie.** Play Store si modul vypýta cez `fdfe/moduleDelivery` pod Play účtom zariadenia a dostane URL, veľkosť a SHA-256 pre každý modul. [ZISTENÉ](#) (Play Store 53.5) · [ODVODENÉ](#) (že sa k tomuto volaniu pripája hlavička s ID zariadenia)
- **Spiaci kód sa dá zapnúť pre jeden účet.** Tým istým kanálom flagov sa ovládajú funkcie, ktoré prichádzajú vypnuté, napríklad indexer nainštalovaných aplikácií v Appsearch. [ODVODENÉ](#)
- **Čo zastaví cudzích.** Pred načítaním modulu Play services overia podpis APK, vyžadujú presne jedného podpisovateľa a porovnávajú ho so vstavaným zoznamom Googlu. Engine DroidGuard musí nieť jeden pripnutý certifikát a program z požiadavky musí byť podpísaný vstavaným kľúčom RSA. Kontrola nad sieťou alebo konfiguráciou na podstrčenie kódu nestačí; spustí sa len kód podpísaný vlastnými kľúčmi Googlu. [ZISTENÉ](#) `afho.java:1464–1481`, `bysr.java:14–31`, `bysq.java:53`

Mechanizmus teda Googlu umožňuje doručiť podpísaný modul alebo program jedinej registrácii zariadenia či jednému účtu tak, že ho nikto iný nedostane, a bez aktualizácie, ktorú by si mohol ktokoľvek prezrieť. Nič v kóde neukazuje, že sa to robí; to je politika na serveri a z APK ju nevidno. [NEZNÁME](#) Samotná požiadavka DroidGuard identifikuje len profil buildu, hoci engine si po spustení vie prečítať Android ID. [ZISTENÉ](#)

Na GrapheneOS je taký kód stále v sandboxe. Dostane oprávnenia, ktoré ste dali Play services (alebo aplikácii, ktorá ukazuje mapu či reklamu), a nič viac. Privilegovaný stav telefónu, logy a dáta iných používateľov ostávajú mimo dosahu. Vypnutie *dynamického načítavania kódu z úložiska* pre Play services zablokuje stiahnuté moduley a engine DroidGuard; nezablokuje kód, ktorý už je v nainštalovanom APK a zapne ho flag. [ODVODENÉ](#) · [NETESTOVANÉ](#) (účinnok prepínača na Play services)

Čo sa nepodarilo získať

Desať modulov z plánu do testovacieho zariadenia nikdy neprišlo: AmbientContext, AppsearchDynamite, BetterTogether, BrellaDynamite (federatívne učenie), CameraLowLight, Dck (digitálny kľúč od auta), Fastpair, HealthTracking, Idunderstanding a Mdocstore (mobilný vodičák a doklady totožnosti). Server ich pre toto zariadenie nikdy neoznačil ako čakajúce, takže si ich Play services nevypýtali. Závisia od hardvéru alebo funkcií a potrebujú telefón, ktorý ich používa. Na auditovanom Pixeli 10 Pro Fold sa kompilovalo trinásť modulov vrátane Fastpair, Dck, TensorGps, SpoonPc, MediaEffectEnhancement, Home a MlkitDocscanUi, ale ich súbory ležia v súkromnom úložisku aplikácie a bez rootu sa prečítať nedajú. [ZISTENÉ](#) (názvy) · [NEZNÁME](#) (obsah)

Čo to znamená pre vás. Oprávnenia, ktoré dáte Play services, a tie, ktoré dáte akejkoľvek aplikácii s vloženou mapou alebo reklamou Googlu, platia aj pre kód, ktorý ste nikdy neinštalovali a nevidíte ho v žiadnom zozname, a Google ho môže vymeniť bez aktualizácie Play services. Moduly, ktoré som skúmal, sú väčšinou tým, čo hovorí ich názov. Praktické kontroly zostávajú rovnaké: zamietnite oprávnenia, ktoré netreba, a ak sa zaobídete bez Play Integrity a skenovania, zablokujte dynamické načítavanie kódu z úložiska (kapitola 09, bod 8).

12

Dôkazy

Každé tvrdenie označené **ZISTENÉ** vedie na jedno z týchto miest v dekompilovanom kóde. Cesty sú také, ako ich pomenúva jadx 1.5.6 pri dekompilácii APK z indexu artefaktov.

Nezverejnené. Dekompilované zdrojáky, APK a modulové APK získané z testovacieho zariadenia sú proprietárny softvér Googlu a nešírim ich. Ak chcete tvrdenie overiť, stiahnite si rovnaké APK z repozitára GrapheneOS (hash je v indexe artefaktov), dekompilujte ho sami cez jadx 1.5.6 a otvorte uvedenú cestu. Moduly pochádzajú z jedného testovacieho zariadenia, takže zistenia o nich môže overiť len ten, kto si ich stiahne z vlastného zariadenia.

Krížové kontroly

Tvrdenia, na ktorých argument stojí, boli overené priamo v dekompilovanom kóde. Nižšie uvedené obstáli; nič nebolo vyvrátené.

- URL check-inu a brána `READ_PHONE_STATE` pred `getSimSerialNumber()` a `getSubscriberId()`: `bjtv.java:24,131,340,347`. **ZISTENÉ**
- Predvolená URL dávky Clearcut `https://play.googleapis.com/log/batch` čítaná cez príznak: `jwgm.java:12`. **ZISTENÉ**
- Manifest nesie `QUERY_ALL_PACKAGES`, `sharedUserId` a `isolatedSplits`, po jednom. **ZISTENÉ**
- Cesta nahrávania APK je fronta: intent `UPLOAD_APK`, posunutý do `InternalApkUploadChimeraService` s prijímačmi zmeny siete a zmeny balíka. **ZISTENÉ**
- Žiaden dekompilovaný súbor nespomína GrapheneOS ani GmsCompat, čo podporuje „nezmenené zostavenie Googlu“. **ZISTENÉ**

Overené navyše (počty sú súbory v dekompilovanom kóde, ktoré odkazujú na dané API, nie počty volaní):

Dotazy na nainštalované aplikácie	122 súborov používa <code>getInstalledPackages</code> , <code>getInstalledApplications</code> alebo <code>queryIntentActivities</code>
Štatistiky používania	26 súborov odkazuje na <code>UsageStatsManager</code> / <code>queryUsageStats</code> (chránené oprávnením špeciálneho prístupu)
Zdroj inštalácie	27 súborov číta balík inštalátora alebo zdroj inštalácie
Čítanie Android ID	35 súborov číta <code>"android_id"</code>

Skenovanie rádii	16 súborov číta výsledky skenovania Wi-Fi alebo informácie o bunkách (zásobník polohy)
SMS	3 súbory sa dotýkajú providera SMS
Samoprepínanie	4 súbory menia stav povolenia komponentov (brtr , gwyn , ky , brtq)
Sieťový zásobník	55 súborov odkazuje na Cronet; split Cronet je súčasťou repozitára
Sú to počty odkazov, nie dôkaz použitia na konkrétnom zariadení. Či nejaká cesta beží, závisí od oprávnení, príznakov a účtu.	

Miesta v kóde

DŮKAZ	ČO PREUKAZUJE
defpackage/jwbq.java:6	predvolená URL servera check-inu
defpackage/bjtv.java:24,262,501,652	zostavenie URL check-inu a záložné adresy gstatic
defpackage/bjtv.java:131,340,347	brána READ_PHONE_STATE, čítanie IMSI a sériového čísla SIM
defpackage/bjzi.java:288–292	polia požiadavky check-inu podmienené jednotlivo príznakmi
defpackage/bjsc.java:41–42,122–124	súbor bezpečnostného tokenu a /data/misc_ce/<user>/checkin
com/.../checkin/CheckinChimeraService.java:187	interval 43 200 s, flexibilita 10 800 s
defpackage/jwpq.java:30	interval check-inu Chimera 43 201 s
defpackage/jxaa.java:15	interval Phenotype 86 400 s
defpackage/jwgm.java:12	koncový bod dávky Clearcut
com/.../clearcut/internal/PlayLoggerContext.java:67	uploadAccount / logAndroidId / úroveň PII
defpackage/brjn.java	468 konštánt enumu zdrojov logov
defpackage/dwrc.java:232,239	koncové body odosielania Measurement a SGTm
defpackage/jyqj.java:335	koncový bod registrácie FCM
defpackage/jyqc.java:11–12	zoznam záložných hostiteľov mtalk
com/.../security/verifier/ApkUploadChimeraService.java:20	intent nahrávania APK
com/.../core/providersettings/GoogleSettingsChimeraProvider.java:58	staré klientske ID GSF vrátane rlz
defpackage/brtr.java:48, defpackage/gwyn.java:63	samoprepínanie stavu komponentov
defpackage/gtub.java:8, wearable/node/DataService\$1.java:105	pripnutý názov balíka a SHA-1 certifikátu pre Play services a Play Store
assets/cobalt_config.pb	konfigurácia metrík so šumom RAPPOR
AndroidManifest.xml	názvy komponentov použité v 10.2

DŮKAZ	ČO PREUKAZUJE
defpackage/bysw.java:90–92	engine DroidGuard načítaný cez <code>DexClassLoader</code> ako <code>com.google.ccc.abuse.droidguard.DroidGuard</code>
defpackage/bysr.java:14–15	dva pripnuté SHA-256 odtlačky podpisového certifikátu DroidGuard
defpackage/byre.java:52	<code>app_dg_cache</code> : kde leží APK enginu
defpackage/bysq.java:53–60	podpísaná creation response, potom <code>byteCode</code> / <code>vmUrl</code> / <code>vmChecksum</code>
defpackage/afgy.java:189 · afin.java:159 · afhx.java:81	<code>DelegateLastClassLoader</code> pre Chimera moduly zo súboru, zo splitu a z iného balíka
defpackage/afhi.java:25 + afho.java:1144	<code>assets/chimera-modules/*.apk</code> sa kopírujú do úložiska aplikácie
com/.../nativeloader/NativeHotLoader.java:58	<code>hotLoad(String path)</code> → <code>dlopen</code> v <code>libisolation_mgr_jni</code>
defpackage/brvn.java:84–155	reflexívny prepis <code>pathList.nativeLibraryDirectories</code>
defpackage/fymj.java:22–40	<code>snet_shared_uuid</code> : vytvorenie, uloženie, rotácia vzdialeným počítačom
defpackage/fyrt.java:55,1332 · fypo.java:51	zoznam kolektorov SNET, predvolene zapnutá podmnožina a balíky DEX-info
defpackage/fysz.java:27 · fykj.java:27–28 · fykf.java:21–22	koncový bod nahrávania SNET/Play Protect; relaye OHTTP pre Safe Browsing a zdroj kľúčov HPKE

Index artefaktov

GMS base	base.apk z apps.grapheneos.org · sha256 1bfbf0c0ddab7470...
Splity	split_*.apk · 11 súborov, všetky overené hashom voči metadátam repozitára
GSF	base.apk z rovnakého repozitára · 5,7 MB
Play Store	base.apk z rovnakého repozitára · 63,9 MB
Metadáta repozitára	metadata.1.0.0.sjson · 98 973 bajtov
Dekompilované	výstup jadx 1.5.6 na mojom stroji · 920 MB · 161 628 tried (nezverejnené)
Reťazce	773 000 záznamov extrahovaných z APK (nezverejnené)
Inventár hostiteľov	778 hostiteľov, 1 307 URL extrahovaných z reťazcov (nezverejnené)
Pracovné poznámky	zoznam modulov (280 riadkov), natívne knižnice (39), cesty dynamického načítavania (14), kontroly tvrdení (36); uchovávané súkromne
Moduly	24 modulových APK z jedného testovacieho zariadenia, SHA-256 zapísané; uchovávané súkromne

Nástroje

apktool · jadx 1.5.6 · aapt2 · apksigner · protoc --decode_raw · Ghidra 12.2 headless · radare2 · vlastný parser tabuľky reťazcov DEX · GrapheneOS frameworks_base vetva 17 (úrovne ochrany oprávnení) · metadáta repozitára aplikácií GrapheneOS (signify). Report vznikol postupom skillu leanrea (lokálne CLI nástroje bez MCP).

Zdroje

ODKAZ	MIESTO
Repozitár aplikácií	https://apps.grapheneos.org/ · metadata.1.0.0.sjson
Artefakt GMS	packages/com.google.android.gms/264032035/base.apk
Dokumentácia GrapheneOS	https://grapheneos.org/usage · sandboxed Google Play
GmsCompat	github.com/GrapheneOS/platform_packages_apps_GmsCompat
Úrovne oprávnení	github.com/GrapheneOS/platform_frameworks_base · vetva 17 · core/res/AndroidManifest.xml
Nástroje	jadx 1.5.6 · apktool · aapt2 · apksigner 36.0.0 · protoc

Hranice dôkazov

Tento report je meraním artefaktu, nie rozsudkom nad firmou. Kód ukazuje, čo je *možné* a čo je *zapojené*. Nevie ukázať, čo sa uchováva, spája alebo používa. Značky čítajte doslovne: „zistené“ znamená, že je to v kóde, „odvodené“, že ide o úvahu nad videným.

Podpora

Ak vám rozbor pomohol a chcete ho podporiť, môžete poslať ľubovoľnú sumu. Vyberte si, čo vám vyhovuje.



Lightning (BOLT 12)

<lno1pgvqhmrp0ys8xetjwe5kxetnypskuctv09ekjueq2d93vggz7yjsxrhlfgjjju2xf4qnvqd9fjk8acdy4cf7e7kep33vspyw8kplcs>

[Zaplatiť v peňaženke](#)



Monero

<89jGcmA65CReki73oLFvxHUzyR45wMqTN8G93ZYRizwVF2XUQgYKbPLGQYgxj2YjX4fxAUPQ7tLr4ATP7fpR4dSeHH3r9iH>

[Zaplatiť v peňaženke](#)



Zcash (unified adresa)

<u1defv75w2x9p6u82qu8je0rrz5g98gnhvs0ervckctfrjvsscvcx00e4qgfk63vm0uar587exxhtp3ps7gwd54anwm32a3sg2432htr53nv22d23agngdpdkdt75lzd8pp0e3f6s2c6n52nk9ecmnjwvdxggfcl7ndsh5uq3wqptk60z>

Ostatné formy podpory: juraj.bednar.io/podporte-ma/