GOOGLE PLAY INTEGRITY
(OPENSOURCE ALTERNATIVE)

Securing Android Apps from Abuse

Unified Attestation proposal by Volla Systeme GmbH

What Is Attestation?

Attestation is a process where a system proves its **identity and integrity** to another system using cryptographic evidence.

In Android's context (e.g., SafetyNet or Play Integrity API):

- Attestation is Google's way of confirming that:
    - The **app is legitimate** and untampered.
    - The **device is genuine**, not rooted, not tampered, and is Play-certified.

Key properties of attestation:

- Uses hardware-backed cryptographic keys.
- Binds the attestation to:
    - Device state (bootloader lock)
    - App identity
    - A random nonce

- Returns a **signed Response** from Google.

WHAT IS GOOGLE PLAY INTEGRITY API?

- Part of Google Play services
- Verifies if:
  - The app is genuine and unmodified
  - The device is certified (Play Protect certified)
  - The app was installed from Google Play
- Server-side attestation model using signed responses from Google

ARCHITECTURE OVERVIEW

Client Side (App):

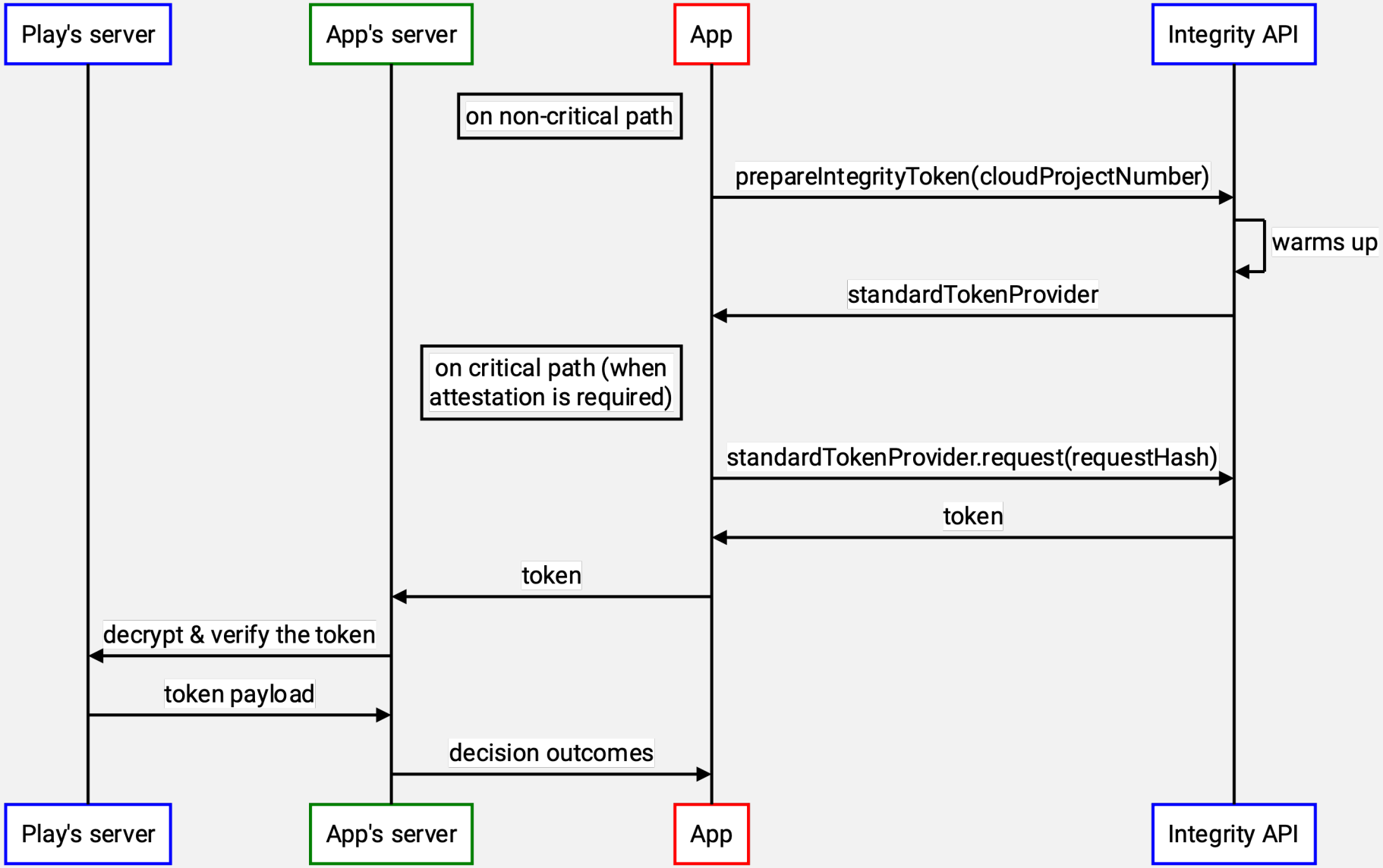
- Calls the Integrity API via Play Core library
- Receives signed response

Google Play Backend:

- Processes request
- Attests the app, device, installer
- Returns signed JWT payload

Developer Backend:

- Verifies and interprets response securely



https://developer.android.com/google/play/integrity/standard

Validate Request

- Confirms request came from a Play-distributed, untampered app.
- Uses app signature and installation source.

Perform Device & App Checks

- Evaluates:
    - App authenticity (e.g., PLAY_RECOGNIZED)
    - Device integrity (e.g., MEETS_DEVICE_INTEGRITY)
    - Licensing/account status (optional)

Generate Signed Response

- Signs a **JSON Web Token** with Google's private key.
- Token contains:
    - Nonce + timestamp
    - App info (package name, SHA-256 cert)
    - Verdicts

Send JWT to App

- App can now send this token to the backend for secure verification and decision-making.

HOW GOOGLE CERTIFIES A NEW ANDROID DEVICE ?

Steps in Certifying a New Device:

1. **OEM (e.g., Samsung, Xiaomi, etc.) submits the device** for certification.
    1. Includes firmware image, build fingerprint, and manufacturing info.

2. **Google runs automated tests**:
    1. Compatibility Test Suite (**CTS**)
    2. Security Test Suite (**STS**)
    3. Google Test Suite (**GTS**)

    These ensure the device adheres to Android compatibility standards and security policies.

3. **Attestation Key Injection**:
    1. If certified, Google securely provisions **attestation keys** (for SafetyNet or Play Integrity).
    2. These are stored in the device's **Trusted Execution Environment (TEE).**
    3. Only certified factories get access to this provisioning process.

4. **Certification Complete**:
    1. Device is marked **Play Protect certified** in Google's systems.
    2. Can now pass MEETS_DEVICE_INTEGRITY or MEETS_STRONG_INTEGRITY.

UNIFIED ATTESTATION

inspired from **unifiedPush** and **unifiedNLP**

What is UnifiedNLP?

- An **open-source alternative** to Google's proprietary location services (Fused Location Provider).

Key Features:

- Provides **location**, **geocoding**, and **activity recognition** without Google Play Services.

Modular backend system:

  - Wi-Fi, Cell, GPS backends (e.g., Mozilla Location Service, LocalGSMBackend).

What is UnifiedPush?

- A **decentralized push notification system** designed to replace Google's FCM (Firebase Cloud Messaging).

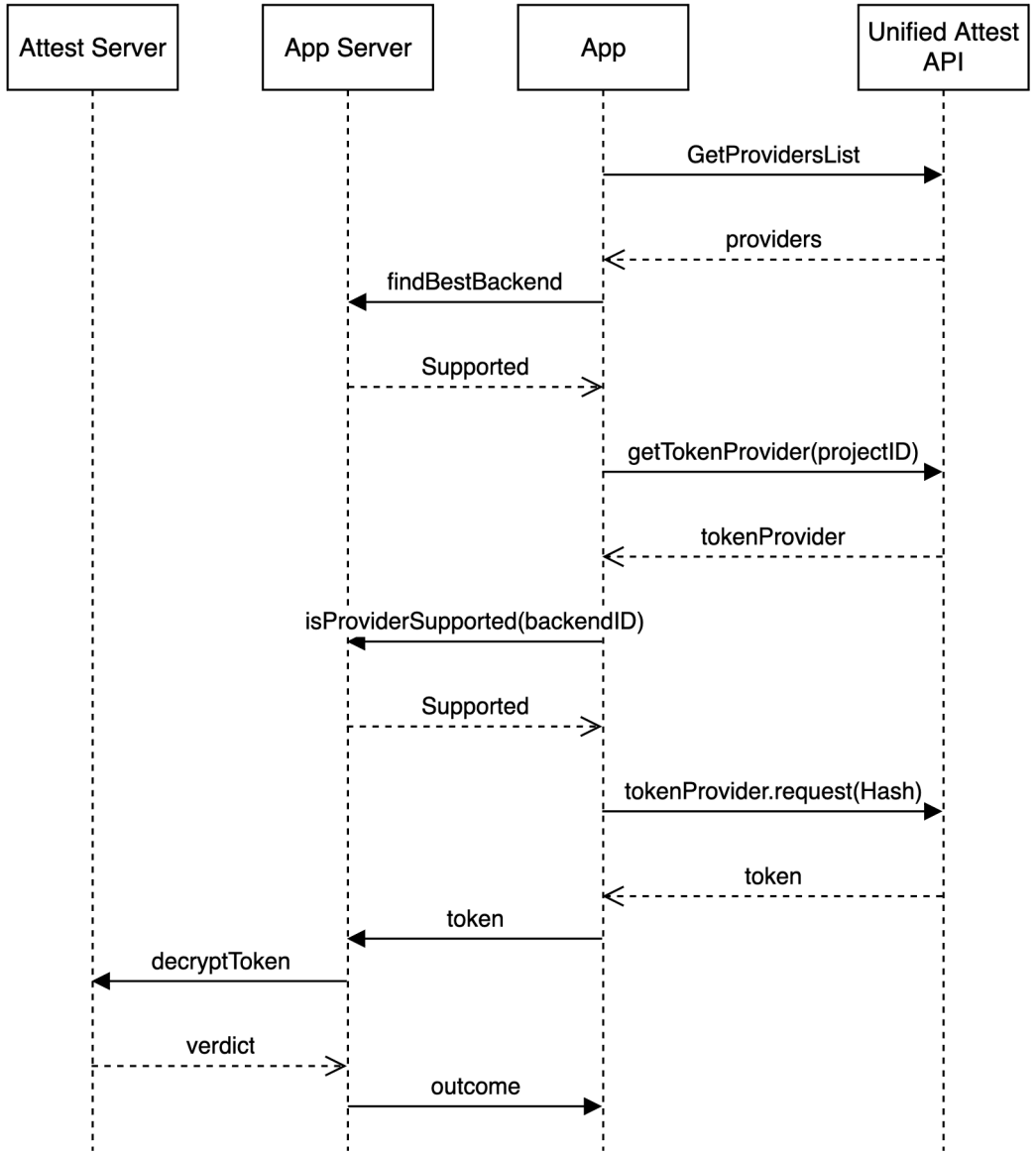
Key Components:

- **Distributor**: App or service that receives and delivers push messages (e.g., ntfy, Gotify, self-hosted).
- **App**: Registers with UnifiedPush API
- **Push Server**: Sends notifications to the distributor

CHALLENGES

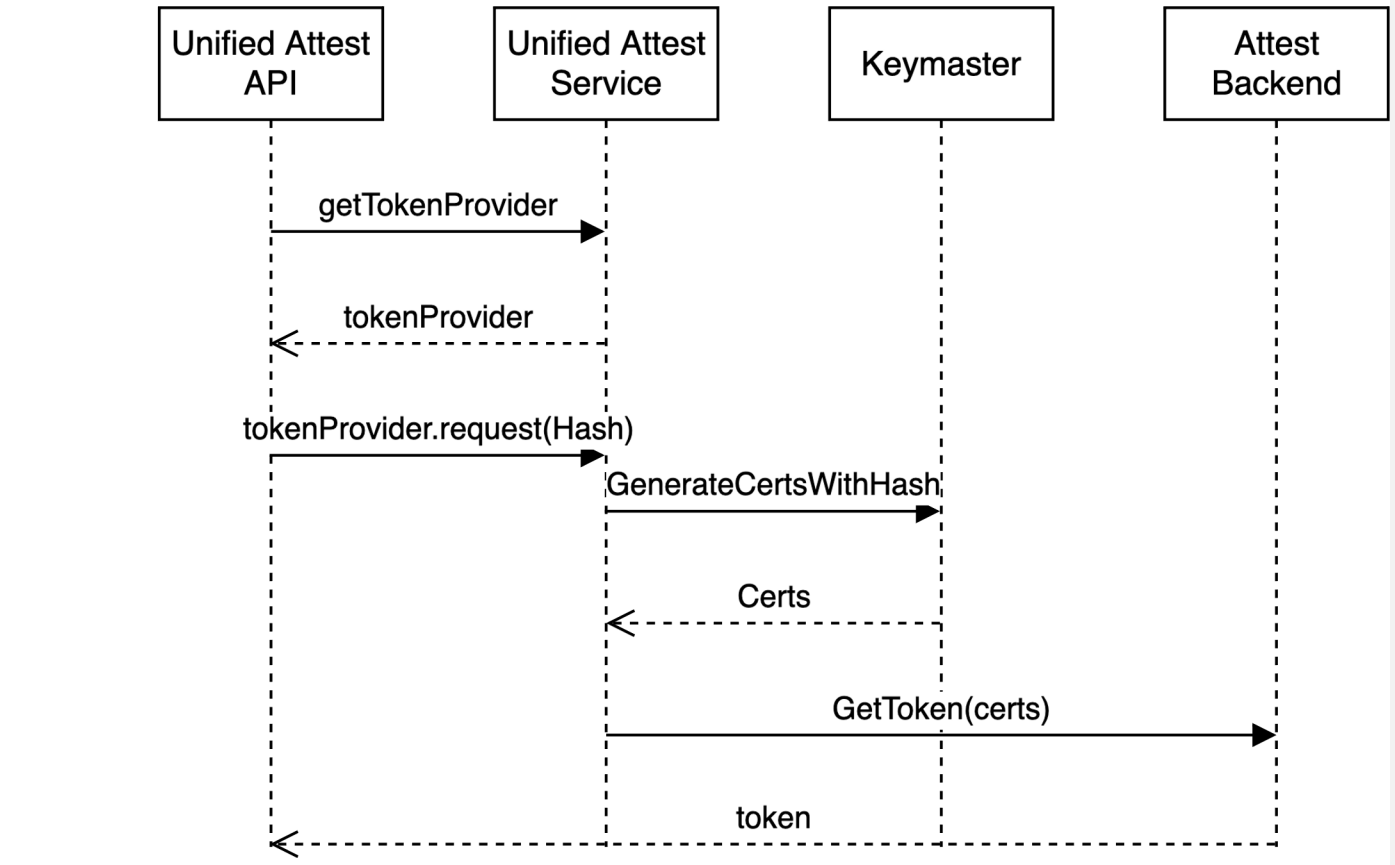
- Ask app **developers** to use this service on app and servers (Google Cloud Console)
- We should implement as **System Service** embedded in firmware (Google play services)
- We want this service to be **Federated** as much as possible (unlike Google play server)
- We have to verify devices security and sanity with **Tests** (Google CTS, STS …)
- We need trusted **laboratories** to run tests on devices (Google MADA third-party labs)

Application Side:

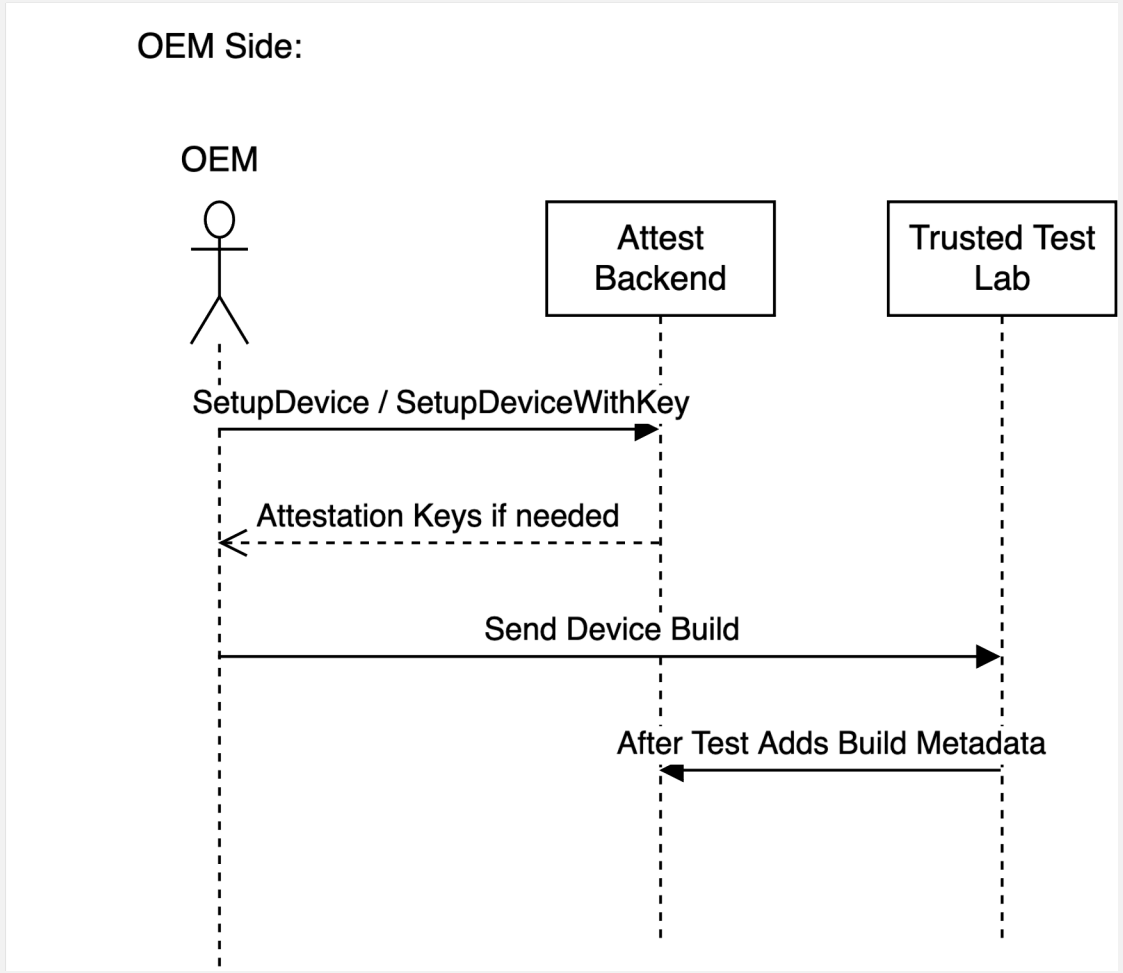


Android Google play services alternative:

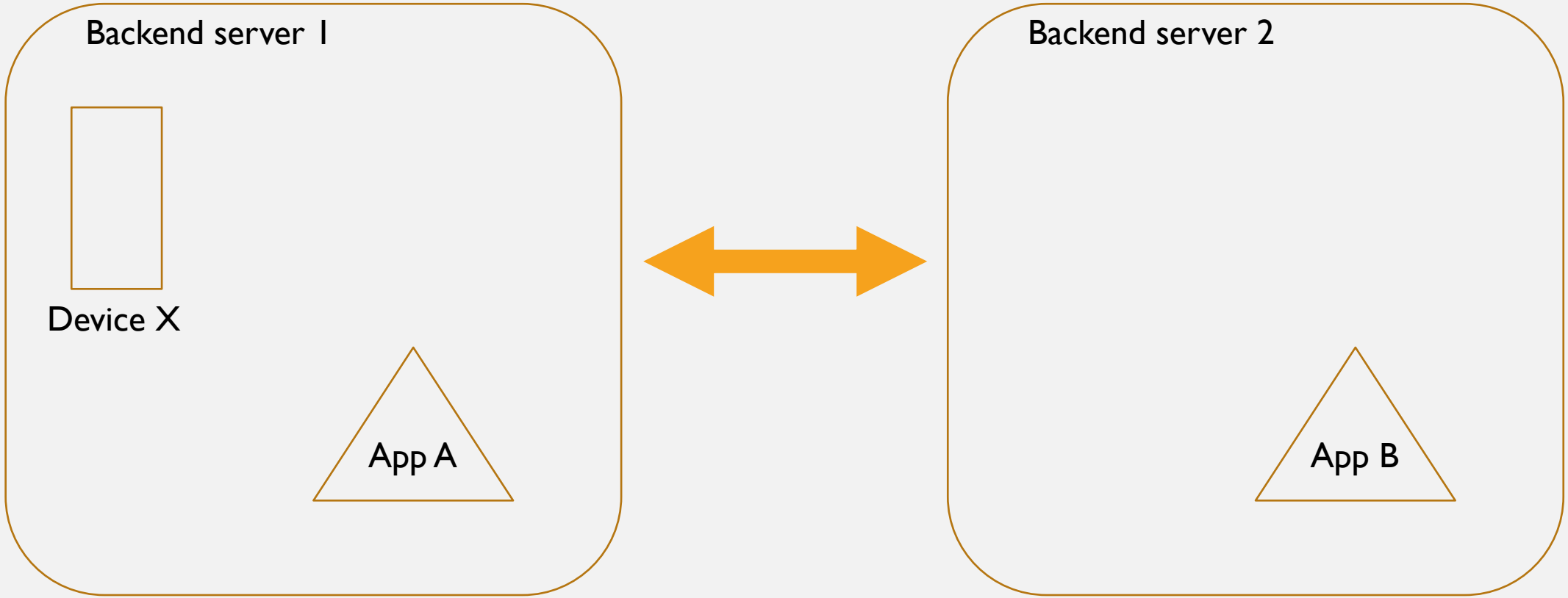
System Service Side:



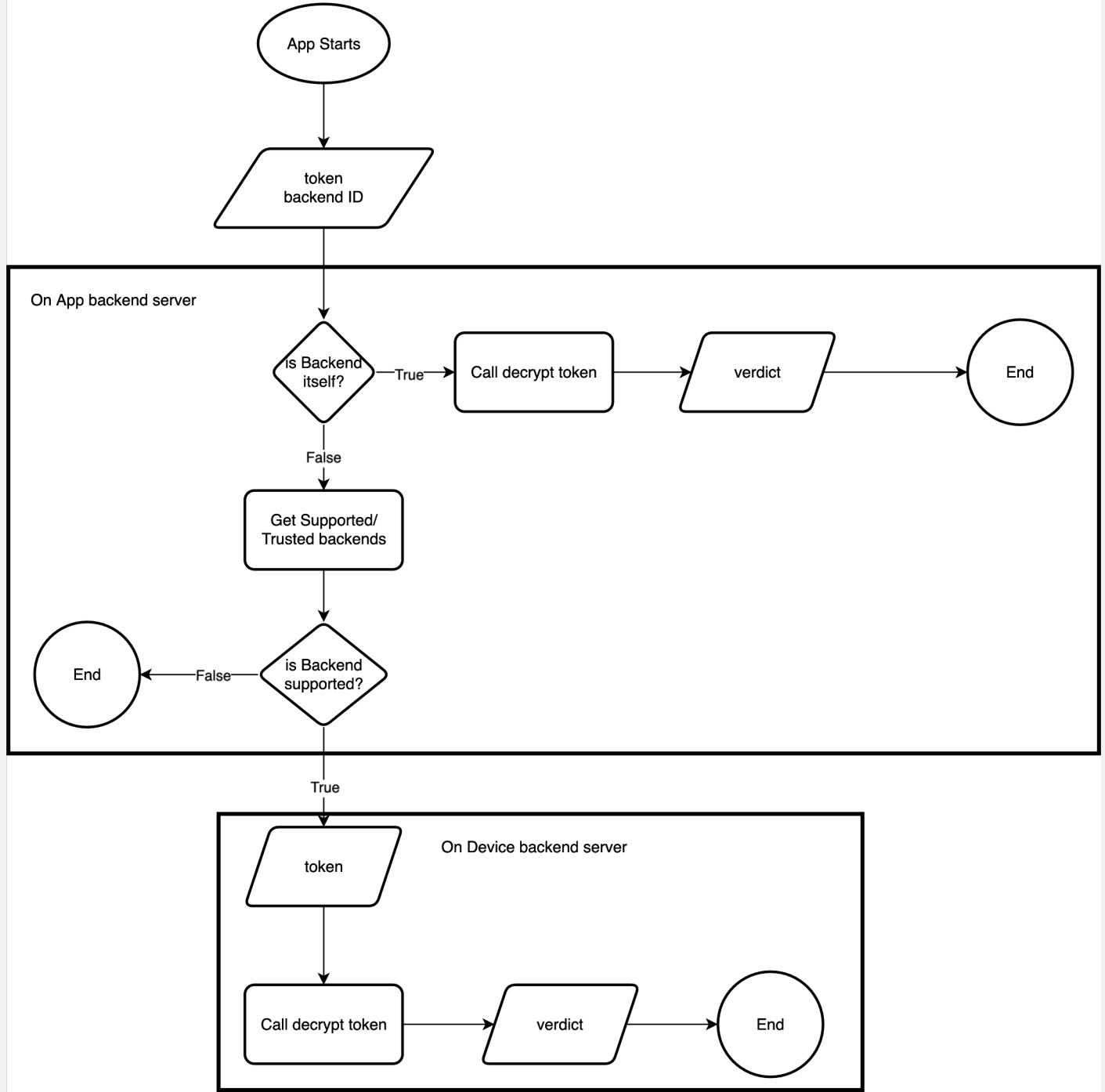
OEM Side:


OEM
Attest Backend
Trusted Test Lab
SetupDevice / SetupDeviceWithKey
Attestation Keys if needed
Send Device Build
After Test Adds Build Metadata

FEDERATED SERVER APPROACH



App Server Side:



API

DEVICE → BACKEND

```
POST /api/v1/device/process
{
  projectId,
  requestHash,
  attestationChain[],
  deviceMeta: {
    manufacturer,
    brand,
    model,
    device,
    buildFingerprint
  }
}
```

APP SERVER → BACKEND

```
POST /api/v1/app/decodeToken
{
  projectId,
  token,
  expectedRequestHash
}
```

OTHER OPENSOURCE PROJECTS

Feature	Unified Attestation	GrapheneOS Auditor	Warden Supreme
Freshness binding	**requestHash**	QR challenge or time-limited challenge	**Nonce challenge**
Proof carrier	Signed token (EAT/JWS-like)	Certificate chains	CSR with attestation extension
Backend verify logic	decodeToken (policy + federation)	Server verifies local device data	Backend verifies CSR + issues cert
Client SDK/Service	System service (AIDL)	Auditor app	Client lib + server verifier

OTHER OPENSOURCE PROJECTS

- **GrapheneOS Attestation**
- Not a Play Integrity–style replacement
- No OEM device submission or certification ingestion
- No CTS / lab-certified device registry model
- No backend-issued integrity tokens for apps
- No app server–verifiable attestation API
- No provider discovery or backend selection
- No federation or enterprise trust model
- Not designed for commercial app ecosystems
- No standardized backend policy enforcement

- **Warden Supreme**
- Server side is a library, not an authoritative backend
- No global or OEM-submitted verified device registry
- Cannot pre-trust CTS / lab-certified device fleets
- No backend identity (issuer) or federation model
- Challenge/nonce lifecycle must be implemented externally
- No provider discovery or backend selection logic
- No unified admin, OEM, or app-dev control plane
- No ecosystem-level trust continuity across deployments

backend-1 | {"level":30,"time":1770646303009,"pid":99,"hostname":"d0a12668af7f","reqId":"req-4g","requestId":"7d4cc3af-c45a-450d-923d-18ed0ed0dc72","chainInfo":[{"index":0,"serial":"01","subject":"CN=Android Keystore Key","issuer":"title=TEE\nserialNumber=05be945217a1cb39cca0fae2f308a088","isSelfSigned":false,"hasAttestationExtension":true},{"index":1,"serial":"96815F8701403CF1EAF51F4D26C4A37D","subject":"title=TEE\nserialNumber=05be945217a1cb39cca0fae2f308a088","issuer":"title=TEE\nserialNumber=3b7add54f66194b56a73d1e4c6db2f41","isSelfSigned":false,"hasAttestationExtension":false},{"index":2,"serial":"C9C9CAA3B6F974ED9AF597BAEF698913","subject":"title=TEE\nserialNumber=3b7add54f66194b56a73d1e4c6db2f41","issuer":"serialNumber=f92009e853b6b045","isSelfSigned":false,"hasAttestationExtension":false},{"index":3,"serial":"F1C172A699EAF51D","subject":"serialNumber=f92009e853b6b045","issuer":"serialNumber=f92009e853b6b045","isSelfSigned":true,"hasAttestationExtension":false}],"msg":"device.process chain summary"}
backend-1 | {"level":40,"time":1770646303022,"pid":99,"hostname":"d0a12668af7f","reqId":"req-4g","requestId":"7d4cc3af-c45a-450d-923d-18ed0ed0dc72","deviceMeta":{"manufacturer":"volla","brand":"volla","model":"Volla Phone Plinius","device":"ansuz","buildFingerprint":"volla/ansuz/ansuz:15/BP1A.250505.005/54-volla-15.0:userdebug/release-keys"},"candidates":[{"id":"cmlf2bn8n0007nh24ta53zr2p","codename":"ansuz","model":"vpp","manufacturer":"Volla","brand":"Volla"}],"msg":"device.process device prefilter mismatch"}

Unified Attestation Portal

App Dev OEM Admin LOGOUT

Account Management

Username | Password | App Dev

Create User

Username: volla
Role: oem
Disable Delete

New password | Change Password

Username: admin
Role: admin
Disable Delete

New password | Change Password

Backend Root Anchors

Generates the backend RSA + ECDSA roots used to chain OEM and device anchors.

Generate Root

UA Backend Root

RSA: 69C68566BC07CD5978D3E90D70BFB45A
ECDSA: 3E7AD8FD9CE1B04756AFA63BDB959359

Created: 2/9/2026, 5:56:00 PM

Revoke
Remove

Settings

Backend ID: 39bd2564-39bd-7164-8514-f8ba2194efc0

Public key: MCowBQYDK2VwAyEA6yQW9IA77h9tq0qF78sGgWTJ95305tC0gjg9ecwdj9M=

Rotate Signing Key

Attestation Authorities

Authority name

Base URL (https://...)

Add Authority

google
https://android.googleapis.com/attestation
RSA: ✅ · ECDSA: ✅
Status cached: 2/9/2026, 5:45:56 PM
Refreshed successfully.

Refresh Roots/Status

Unified Attestation (Local)
http://a.uattest.net/api/v1/info
RSA: ✅ · ECDSA: ✅
Status cached: never

Refresh Roots/Status

Federation Management

Backend URL

Display name (optional)

Add Backend

Unified Attestation Portal

OEM LOGOUT

Devices

ansuz
Model: Volla Phone Plinius

Register Device

Device codename

Model (optional)

Save

ansuz

ID: cmlf2bn8n0007nh24ta53zr2p

Device | Builds | Trust Anchors | Reports

Trust Anchors

Anchors

Select attestation authority

RSA leaf serial hex

RSA intermediate serial hex

ECDSA leaf serial hex

ECDSA intermediate serial hex

Register Anchor | Generate Keys

RSA: F1C172A699EAF51D
RSA intermediate: C9C9CAA3B6F974ED9AF597BAEF698913
ECDSA: F1C172A699EAF51D
ECDSA intermediate: C9C9CAA3B6F974ED9AF597BAEF698913
Authority: google
Created: 2/9/2026, 6:02:30 PM

Revoke | Remove

ansuz

ID: cmlf2bn8n0007nh24ta53zr2p

Device | Builds | Trust Anchors | Reports

Build Policies

volla/ansuz/ansuz:15/BP1A.250505.005/54-volla-15.0:userdebug/release-keys

60c6b25322ffaa185693638c21ae377c4f1c6ecbc54741dbc1a7812d79f41c88

60c6b25322ffaa185693638c21ae377c4f1c6ecbc54741dbc1a7812d79f41c88

150000

202602

☑ Enabled

Update Build

volla/ansuz/ansuz:15/BP1A.250505.005/54-volla-15.0:userdebug/release-keys

Boot key: 60c6b25322ffaa18...
OS version: 150000
OS patch level: 202602
Enabled: yes

Edit Delete

Profile

volla | volla | volla

Save

•••••••••••• | New password

Change Password

OEM Trust Anchor

Generate OEM RSA + ECDSA intermediate certs chained to the backend root.

Status: active

Generate OEM Trust Anchor

RSA Intermediate

C=DE O=Unified Attestation CN=UA volla OEM RSA Intermediate

Serial: 16987C5F0B0B73A7167641AA1E3FE761

ECDSA Intermediate

C=DE O=Unified Attestation CN=UA volla OEM ECDSA Intermediate

Serial: F7FC66313346120E0A17D981C624EB23

Created: 2/9/2026, 6:02:46 PM

Revoke | Remove

OEM trust anchor generated and downloaded.

Federation (Read-only)

```
example-app-server-1  | }
example-app-server-1  | [Example-App-Server] POST /select-backend {
example-app-server-1  |   projectId: 'net.uattest.example',
example-app-server-1  |   canonicalRequest: 'action=login&sessionId=123456&ts=1700000000',
example-app-server-1  |   backendIds: '[39bd2564-39bd-7164-8514-f8ba2194efc0]'
example-app-server-1  | }
example-app-server-1  | [Example-App-Server] POST /verify {
example-app-server-1  |   projectId: 'net.uattest.example',
example-app-server-1  |   canonicalRequest: 'action=login&sessionId=123456&ts=1700000000',
example-app-server-1  |   token: 'eyJ0eXAiOiJ1YS5pbnRlZ3JpdHkrand0IiwiYWxnIjoiRWREU0EiLCJraWQiOiJrMTc3MDE1Mzg1MTMyNiJ9.eyJpc3MiOiIzOWJkMjU2NC0z
OWJkLTcxNjQtODUxNC1mOGJhMjE5NGVmYzAiLCJpYXQiOjE3NzA2NDg2ODIsImV4cCI6MTc3MDY0ODc0MiwicHJvamVjdElkIjoibmV0LnVhdHRlc3QuZXhhbXBsZSIsInJlcXVlc3RIYXN
oIjoiZGNkZGY5Y2NiMTBkZjY5MGNhOTQxOTQwODMwNTQ2ZDJmZTM0YTE0MGVhMzdiMDIzMDI0OGJhZWY1NzJhYWZkOSIsImFwcCI6eyJwYWNrYWdlTmFtZSI6Im5ldC51YXR0ZXN0LmV4YW
1wbGUiLCJzaWduZXJEaWdlc3RzIjpbImExZTI3N2FhMmY2NWU3NGRlNjI4ZTQ5NTMzNjNjMDczY2FmNGIyNDQ4NWM4YTRlMWVkOWNjYzhmMDgwMTU0OTIiXX0sImRldmljZUludGVncml0e
SI6eyJhdHRlc3RhdGlvbkFwcGxpY2F0aW9uSWQiOnsicGFja2FnZU5hbWUiOiJuZXQudWF0dGVzdC5leGFtcGxlIiwic2lnbmVyRGlnZXN0cyI6WyJhMWUyNzdhYTJmNjVlNzRkZTYyOGU0
OTUzMzYzYzA3M2NhZjRiMjQ0ODVjOGE0ZTFlZDljY2M4ZjA4MDE1NDkyIl19LCJvcmlnaW4iOiJHRU5FUkFURUQiLCJ2ZXJpZmllZEJvb3RLZXkiOiIwMDAwMDAwMDAwMDAwMDAwMDAwMDAw
wMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAwMDAw
RCb290SGFzaCI6IjYwYzZiMjUzMjJmZmFhMTg1NjkzNjM4YzIxYWUzNzdjNGYxYzZlY2JjNTQ3NDFkYmMxYTc4MTJkNzlmNDFjODgiLCJvc1ZlcnNpb24iOjE1MDAwMCwib3NQYXRjaExld
mVsIjoyMDI2MDIsInZlbmRvclBhdGNoTGV2ZWwiOjIwMjUwNTA1LCJib290UGF0Y2hMZXZlbCI6MH0sInZlcmRpY3QiOnsiaXNUcnVzdGVkIjpmYWxzZSwicmVhc29uQ29kZXMiOlsiQlVJ
TERfUE9MSUNZX01JU01BVENIIl19fQ.cM7yaVTw71s1-9Ois4WLcMasTSswEaLL0cM8NidEi6Qgq2MUAkWs9LWgYQqXrudjrSc9ehn_X9nFEORuRa5PDw'
example-app-server-1  | }
example-app-server-1  | {
example-app-server-1  |   verdict: { isTrusted: false, reasonCodes: [ 'BUILD_POLICY_MISMATCH' ] },
example-app-server-1  |   requestHash: 'dcddf9ccb10df690ca941940830546d2fe34a140ea37b0230248baef572aafd9',
example-app-server-1  |   claims: {
example-app-server-1  |     iss: '39bd2564-39bd-7164-8514-f8ba2194efc0',
example-app-server-1  |     projectId: 'net.uattest.example',
example-app-server-1  |     requestHash: 'dcddf9ccb10df690ca941940830546d2fe34a140ea37b0230248baef572aafd9',
example-app-server-1  |     app: { packageName: 'net.uattest.example', signerDigests: [Array] },
example-app-server-1  |     deviceIntegrity: {
example-app-server-1  |       attestationApplicationId: [Object],
example-app-server-1  |       origin: 'GENERATED',
example-app-server-1  |       verifiedBootKey: '0000000000000000000000000000000000000000000000000000000000000000',
example-app-server-1  |       deviceLocked: false,
example-app-server-1  |       verifiedBootState: 'UNVERIFIED',
example-app-server-1  |       verifiedBootHash: '60c6b25322ffaa185693638c21ae377c4f1c6ecbc54741dbc1a7812d79f41c88',
example-app-server-1  |       osVersion: 150000,
example-app-server-1  |       osPatchLevel: 202602,
example-app-server-1  |       vendorPatchLevel: 20250505,
example-app-server-1  |       bootPatchLevel: 0
example-app-server-1  |     }
example-app-server-1  |   }
example-app-server-1  | }
root@uattest:~/dockers/apps/a#
```

Unified Attestation Portal

App Dev OEM Admin LOGOUT

Registered Apps

Example app
Project ID: net.uattest.example
Signer digest: a1e277aa2f65e74d...

Register App

App name

Project ID (package name)

Signing cert SHA-256 (hex)

Register

Edit App

Example app

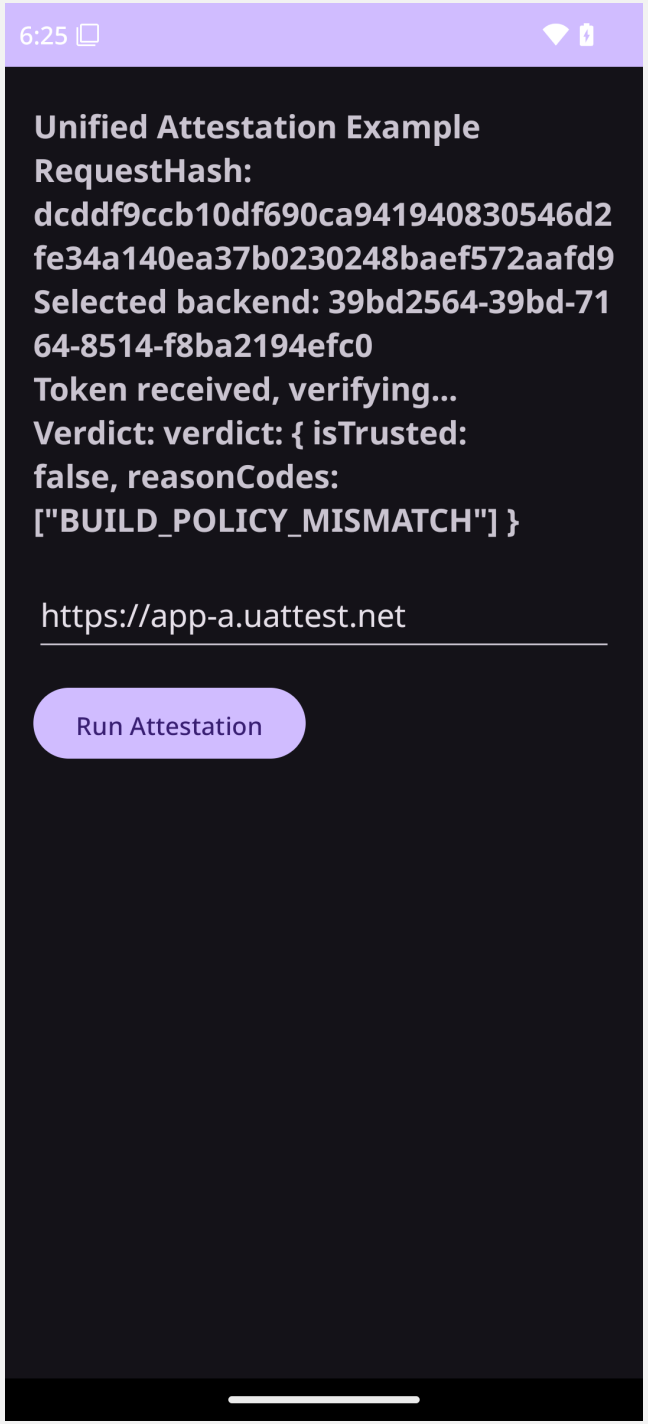
net.uattest.example

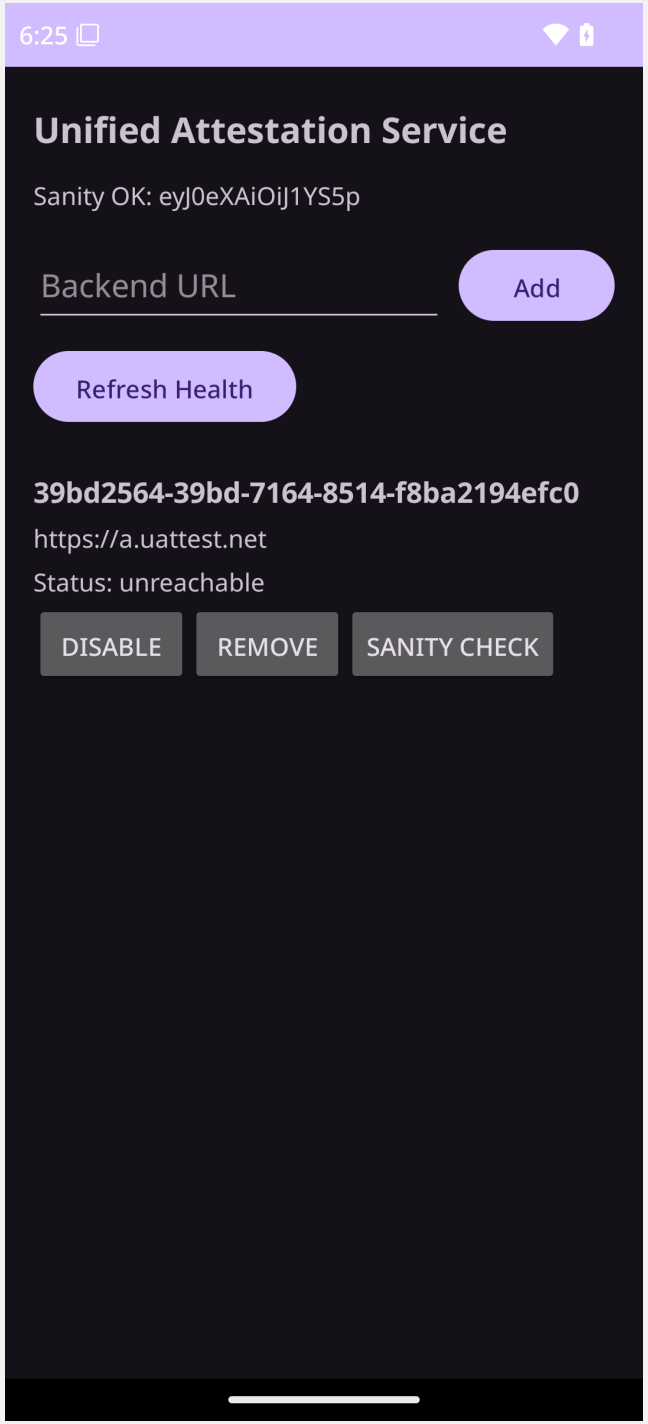
a1e277aa2f65e74de628e4953363c073caf4b24485c8a4e1ed9ccc8f08015492

Save Delete

App Server Secret

Rotate Secret


6:25
Unified Attestation Example
RequestHash:
dcddf9ccb10df690ca941940830546d2
fe34a140ea37b0230248baef572aafd9
Selected backend: 39bd2564-39bd-71
64-8514-f8ba2194efc0
Token received, verifying...
Verdict: verdict: { isTrusted:
false, reasonCodes:
["BUILD_POLICY_MISMATCH"] }
https://app-a.uattest.net
Run Attestation


6:25
Unified Attestation Service
Sanity OK: eyJ0eXAiOiJ1YS5p
Backend URL
Add
Refresh Health
39bd2564-39bd-7164-8514-f8ba2194efc0
https://a.uattest.net
Status: unreachable
DISABLE
REMOVE
SANITY CHECK

Device Reports

Device: 5fe0ad3306f608aa...
Issuer: 39bd2564-39bd-7164-8514-f8ba2194efc0
Last seen: 2026-02-09T14:51:22.693Z
Verdict: rejected

Device: 99e0cfede1bd9421...
Issuer: 39bd2564-39bd-7164-8514-f8ba2194efc0
Last seen: 2026-02-09T14:49:15.982Z
Verdict: trusted

Device: d0db56af5f068802...
Issuer: 39bd2564-39bd-7164-8514-f8ba2194efc0
Last seen: 2026-02-09T14:47:46.239Z
Verdict: rejected

Device: 1d00336e9acfc3e1...
Issuer: 39bd2564-39bd-7164-8514-f8ba2194efc0
Last seen: 2026-02-09T14:46:21.778Z
Verdict: rejected

Device: a76c7bf4faf60f11...
Issuer: 39bd2564-39bd-7164-8514-f8ba2194efc0
Last seen: 2026-02-09T14:45:31.720Z
Verdict: rejected

Device: 1168e524cf8266af...
Issuer: 39bd2564-39bd-7164-8514-f8ba2194efc0
Last seen: 2026-02-09T14:38:24.566Z
Verdict: rejected

ansuz

ID: cmlf2bn8n0007nh24ta53zr2p

Device | Builds | Trust Anchors | Reports

Failing Devices

Device: 906b145c9247fa5c...
Last seen: 2026-02-09T14:55:12.908Z
Build fingerprint: unmatched
Reasons: BUILD_POLICY_MISMATCH

Device: 5fe0ad3306f608aa...
Last seen: 2026-02-09T14:51:22.693Z
Build fingerprint: unmatched
Reasons: BUILD_POLICY_MISMATCH

Device: d0db56af5f068802...
Last seen: 2026-02-09T14:47:46.239Z
Build fingerprint: unmatched
Reasons: BUILD_POLICY_MISMATCH

Device: 1d00336e9acfc3e1...
Last seen: 2026-02-09T14:46:21.778Z
Build fingerprint: unmatched
Reasons: BUILD_POLICY_MISMATCH

Device: a76c7bf4faf60f11...
Last seen: 2026-02-09T14:45:31.720Z
Build fingerprint: unmatched
Reasons: BUILD_POLICY_MISMATCH

Device: 1168e524cf8266af...
Last seen: 2026-02-09T14:38:24.566Z
Build fingerprint: unmatched
Reasons: BUILD_POLICY_MISMATCH

TODO

- More testing
- General UI/UX fixes
- Federation Tests
- Better Docs
- Implementing RKP server
- Validate Origin of the installed package

SUMMARY

- **Open Source**: Transparent, vendor-independent attestation framework to verify app integrity and device trust. **without relying on Google services**.
- **App Integrity:** Ensure the app is untampered using digital signatures or checksums.
- **Device Integrity:** Detect rooted/emulated/unofficial devices using secure.
- **Signed Attestation:** Generate and verify tamper-proof attestation tokens using **public/private key pairs**.
- **Server-Side Validation:** Provide a backend API for securely checking client claims, nonces, and timestamps.
- **Operating System Independent:** This service is not android only and can be implemented on Ubuntu Touch or other Mobile OS

REFERENCES

- https://github.com/microg/UnifiedNlp
- https://unifiedpush.org/
- https://developer.android.com/google/play/integrity

UATTEST

- https://uattest.net/
- https://github.com/unifiedattestation/