RATS
Internet-Draft
Intended status: Standards Track
Expires: 14 September 2026

R. Krishnan
JPMorgan Chase & Co.
N. Smith
Intel
D. Lopez
Telefonica
A. Prasad
Oracle
S. Addepalli
Aryaka

13 March 2026

Privacy Preserving Verifiable Geofencing with Residency Proofs for Sovereign Workloads

draft-lkspa-rats-verifiable-geo-fence-00

Abstract

Modern cloud and distributed computing rely heavily on software-only identities and bearer tokens that are easily stolen, replayed, or used from unauthorized locations. Furthermore, traditional methods of location verification — such as IP-address-based geolocation — are easily spoofed via VPNs or proxies and significantly compromise infrastructure security and privacy for sovereign workloads and high-assurance environments.

This document defines the *Verifiable Geofencing Attestation Profile (V-GAP)*, a profile of the RATS Architecture {{!RFC9334}}, that solves these challenges through hardware-rooted cryptographic verifiability. A host machine runs a Workload Identity Agent for managing the workload identities on that platform. This profile replaces implicit trust and spoofable indicators with cryptographically verifiable hardware-rooted Evidence of integrity and location for this agent. Critically, this framework prioritizes location privacy by utilizing Zero-Knowledge Proofs (ZKPs), allowing a workload to prove it is within a compliant zone without disclosing precise coordinates.

By binding software identities to persistent silicon identities and verified physical residency, V-GAP establishes a silicon-to-workload chain of trust. It ensures that sensitive operations are only performed by authorized workloads running on untampered hardware in cryptographically verified, privacy-preserving geographic boundaries, fulfilling the high-assurance requirements of the WIMSE Architecture {{!I-D.ietf-wimse-architecture}}.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Operators of sovereign and high-assurance workloads need cryptographic assurance that sensitive computation occurs only on approved hardware within approved geographic boundaries. Traditional methods — IP-based geolocation, region labels, bearer tokens — are easily spoofed, stolen, or replayed and provide no hardware-rooted verifiability. Key gaps include:

* *Unverifiable location metadata:* Location tags for data objects are typically unsigned, making provenance and integrity difficult to validate.
* *Token theft and replay:* Bearer tokens can be copied and replayed from unauthorized hosts or locations.
* *Implicit trust in "region" and transit:* A relying party often cannot cryptographically verify a server's physical residency, and requests may traverse intermediaries that expand the effective trust boundary.

This document defines the Verifiable Geofencing Attestation Profile (V-GAP), a profile of the RATS Architecture {{!RFC9334}} that makes *platform integrity* and *geofence residency* verifiable inputs to workload credential issuance. V-GAP enables a Relying Party (or credential issuer) to require Evidence that:

1. the Workload Identity Agent is running on an approved, measured platform (platform integrity); and
2. that platform is resident within an approved geographic boundary, optionally without revealing coordinates (residency).

To maintain location privacy while providing cryptographic verifiability, V-GAP supports Transparent Zero-Knowledge Proofs (ZKPs) — non-interactive, hash-based proofs that allow a platform to demonstrate "in-zone" residency without disclosing exact coordinates and without requiring a trusted setup.

1.1. Scope and Layering

V-GAP covers platform integrity verification (Layer 2) and residency verification (Layer 3). It assumes that the binding of individual workloads to the local Workload Identity Agent (Layer 1) is established by a co-location mechanism such as {{!I-D.mw-wimse-transitive-attestation}}. Together, these three layers form a complete chain of trust from silicon to workload, as required by the WIMSE Architecture {{!I-D.ietf-wimse-architecture}}.

Layer	Scope	Responsibility
Layer 1	{{!I-D.mw-wimse-transitive-attestation}}	Bind workload to a local Workload Identity Agent (co-location).
Layer 2	This document	Verify platform integrity for the Workload Identity Agent (platform Evidence).
Layer 3	This document	Verify platform residency within an approved boundary (location Evidence).

Table 1

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 {{!RFC2119}} {{!RFC8174}} when, and only when, they appear in all capitals, as shown here.

2.1. Abbreviations

* *AK*: Attestation Key
* *BMC*: Baseboard Management Controller
* *DAA*: Direct Anonymous Attestation
* *EAT*: Entity Attestation Token
* *EK*: Endorsement Key
* *GNSS*: Global Navigation Satellite System
* *IMA*: Integrity Measurement Architecture
* *IMEI*: International Mobile Equipment Identity
* *IMSI*: International Mobile Subscriber Identity

* *LAH*: Location Anchor Host
* *OOB*: Out-of-Band
* *PCR*: Platform Configuration Register
* *PoR*: Proof of Residency
* *SPDM*: Security Protocol and Data Model
* *STARK*: Scalable Transparent ARgument of Knowledge
* *SVID*: SPIFFE Verifiable Identity Document
* *TEE*: Trusted Execution Environment
* *TPM*: Trusted Platform Module
* *V-GAP*: Verifiable Geofencing Attestation Profile
* *ZKP*: Zero-Knowledge Proof

3. Key Terms

Data Residency: Requirement that data processing and storage remain within an approved geographic boundary.

Geofencing: Enforcement that workloads execute only on approved hosts within an approved geographic boundary.

Workload Identity Agent: On-host component that issues workload identities (for example, SVIDs) to local workloads, subject to verifier-approved evidence.

Location Anchor Host (LAH): Trusted host or device that produces location evidence used to establish residency within a geofence. In RATS terms, the LAH is the Attester.

Workload Host: Physical or virtual machine running the Workload Identity Agent and workloads; produces platform evidence. Unless otherwise stated, this document assumes the unified deployment model in which the Workload Host and the Location Anchor Host (LAH) are the same machine.

Composite Geolocation: Location estimate fused from multiple sources and accompanied by a quality indicator.

Proof of Residency (PoR) / Co-location: Evidence that binds a workload (or Workload Host) to an approved local environment and geofence for a specific attestation interval.

Silicon Root of Trust: Hardware trust anchor that supports measured boot and protects attestation keys.

Transparent Zero-Knowledge Proof: ZKP that does not require a trusted setup; used to prove "inside an approved zone" without revealing precise coordinates.

Workload Identity Management Plane: Issues and validates workload identities and trust bundles based on verifier results and policy. In RATS terms, this plane acts as the Relying Party and Credential Issuer.

Host Identity Management Plane: Verifies platform integrity and residency evidence, and manages attestation key registration and platform health state (often via OOB paths). In RATS terms, this plane acts as the Verifier.

V-GAP (Verifiable Geofencing Attestation Profile): Nested evidence

format defined in this document for binding identity to verified platform integrity and verified residency.

N_fusion (Workload Fusion Nonce): Fresh nonce used to bind identity issuance to a specific attestation interval, delivered by the Workload Identity Management Plane. Corresponds to the nonce field in the lah-bundle.

4. Use Cases

This profile supports attested data residency and geofencing for workloads and (optionally) users. Common use cases fall into: server-centric enforcement, user-centric enforcement, and compliance and risk reduction.

4.1. Server-centric Enforcement

Enterprises need cryptographic proof that workloads run only on approved hosts within an approved geographic boundary, and that data flows only between approved boundaries.

* *Workload-to-workload (general):* Relying parties accept workload identities only when the issuing host attests platform integrity and "in-zone" residency, preventing credentials from being used outside the approved boundary.

* *Agentic AI workloads:* An AI agent may access sensitive data or perform sensitive actions only when its Workload Identity Agent presents hardware-rooted integrity evidence and a verifiable "in-zone" proof (optionally privacy-preserving), binding identity to both platform state and residency.

* *Federated / edge AI (key or model release):* High-value artifacts (e.g., decryption keys or model weights in federated learning) are released only when the partner/edge host attests it is integral and resident within the required boundary. This is useful for intermittently connected sites.

* *User-to-server:* Clients validate that the server endpoint is operating within an approved boundary (e.g., by policy tied to the server’s attested identity and residency evidence).

4.2. User-centric Enforcement

Enterprises may also need trustworthy location signals for user-facing access decisions.

* *Geofenced access control:* User access is permitted only when the user (or user device) proves it is within an allowed boundary, ideally without requiring precise location disclosure.

* *On-premises boundaries:* Customer-premises equipment can define an enterprise boundary, with a network or enterprise infrastructure providing supporting evidence for policy enforcement.

* *Restricted support geographies:* Administrative or support actions can be allowed only when the operator proves presence within allowed geographies, reducing policy and insider-risk exposure.

4.3. Compliance and Risk Reduction

Geofence attestation provides audit-ready evidence to support data residency and sovereignty controls, and it can also reduce non-compliance risk from misconfiguration or spoofable signals. Even when not mandated, "in-zone" proofs help address: configuration drift, edge relocation/proxying, contractual residency requirements, and location-privacy minimization (proving "inside the zone" without storing coordinates).

5. Verifiable Geofencing Attestation Profile (V-GAP)

V-GAP is a profile of the RATS Architecture {{!RFC9334}} that binds a Workload Identity Agent to (1) hardware-rooted platform integrity and (2) verified residency within a configured geofence. The Attester (Location Anchor Host) produces a V-GAP Evidence structure (lah-bundle) that the Verifier (Host Identity Management Plane) appraises to produce an Attestation Result.

5.1. RATS Role Mapping

V-GAP instantiates the RATS Architecture {{!RFC9334}} with the following role assignments.

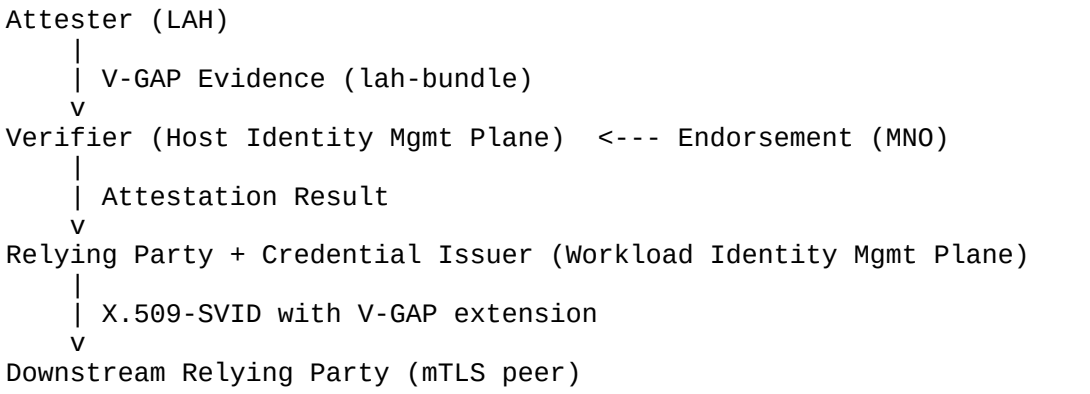
RATS Role ({{!RFC9334}})	V-GAP Entity	Function
Attester	Location Anchor Host (LAH)	Produces V-GAP Evidence (the lah-bundle), including TPM quotes and geolocation claims.
Verifier	Host Identity Management Plane	Appraises V-GAP Evidence — validates TPM quotes, checks PCRs, verifies geolocation proofs — and produces an Attestation Result.
Endorser	Mobile Network Operator (MNO)	Provides a location Endorsement (mno-endorsement) attesting device location within carrier visibility.
Relying Party + Credential Issuer	Workload Identity Management Plane	Consumes the Attestation Result and decides whether to issue or renew workload credentials (e.g., X.509-SVIDs). Also acts as the CA that signs the credential.
Downstream Relying Party (note 1)	mTLS peer / service consumer	Consumes the workload credential containing the V-GAP extension; trusts the CA signature as proxy for verified integrity and residency.

Table 2

Note 1: "Downstream Relying Party" is not a role defined by {{!RFC9334}}. It is used here to distinguish the entity that consumes the issued credential from the Relying Party that consumes the Attestation Result.

5.2. Evidence Flow

The V-GAP evidence flow follows the RATS background-check model ({{!RFC9334}}, Section 3.2): the Attester conveys Evidence to the Verifier, the Verifier appraises it and conveys the Attestation Result to the Relying Party.



The Relying Party in V-GAP also acts as a Credential Issuer (CA): it materializes the Attestation Result into an X.509 workload identity credential (for example, a SPIFFE SVID) containing the V-GAP Evidence as a CRITICAL extension. This "trust translator" pattern allows downstream consumers to rely on standard X.509/mTLS verification without needing to understand RATS or V-GAP directly.

Implementations of this profile MUST mark the X.509 extension containing the V-GAP Evidence Bundle as CRITICAL. When marked CRITICAL, any downstream Relying Party that does not understand the extension MUST reject the credential, enforcing fail-closed behavior for residency-constrained workloads.

5.3. V-GAP Evidence Structure

The lah-bundle is the RATS Evidence structure defined by this profile. It is a hardware-sealed object embedded as an X.509 extension (OID 1.3.6.1.4.1.65284.1.1) in the workload identity credential (for example, a SPIFFE SVID). It binds a workload identity to physically verifiable claims — TPM hardware identity, privacy-preserving geolocation, and agent binary integrity — without exposing PII.

5.3.1. Top-Level Structure

```
{
  "lah-bundle": { },
  "mno-endorsement": { },
  "workload": { }
}
```

5.3.2. lah-bundle Fields

Field	Type	Required	Description
tpm-ak	string (Base64URL)	Yes	TPM Attestation Key public key (PEM-encoded). Hardware identity anchor. The TPM enforces that only this key can produce tpm-quote-seal — proving co-residency.
geolocation-id-hash	string (Base64URL)	Yes	SHA-256 over tpm-ak-bytes concatenated with any sensor-specific identifiers (see Sensor Type Input Recipes appendix for per-sensor constructions). Binds the TPM identity anchor to the geolocation sensor identity. Sensor integrity is assumed to be established by the host management plane via an out-of-band channel.
geolocation-proof-hash	string (Base64URL)	Yes	SHA-256 commitment over geolocation-payload. Required in both privacy modes. When privacy-technique=zkp: SHA-256(zkp-proof-bytes). When privacy-technique=none: SHA-256(JCS({lat, lon, accuracy})).
privacy-technique	string enum	Yes	"none" = raw lat/lon/accuracy in payload. "zkp" = zero-knowledge proof URI in payload. Controls location privacy only; device identity privacy is

			always protected via geolocation-id-hash.
geolocation-payload	object	Yes	Inner location data. Structure depends on privacy-technique (see Payload Variants below). Committed to by geolocation-proof-hash and optionally signed by mno-endorsement.mno-sig.
nonce	string (Base64URL)	Yes	Freshness nonce (N_fusion) issued by the Relying Party (Workload Identity Management Plane) for each attestation interval. Implementations may use chained nonce constructions for additional audit guarantees (see Nonce Chain and Merkle Audit Log appendix).
timestamp	integer (int64)	Yes	Unix epoch seconds. Set by the Attester (LAH) at bundle construction time.
tpm-quote-seal	string (Base64URL)	Yes	TPM2_Quote produced by the AK in tpm-ak. Qualifying data = SHA-256(JCS({tpm-ak, geolocation-id-hash, geolocation-proof-hash, privacy-technique, nonce, timestamp, workload-identity-agent-image-digest})). Binds all fields into a single hardware-sealed statement.
workload-	string (hex	Yes	SHA-256 digest of the

identity-agent-image-digest	SHA-256)		Workload Identity Agent binary, measured at attestation time by the Verifier (Host Identity Management Plane). Detects agent binary compromise on every renewal cycle.

Table 3

5.3.3. geolocation-payload Variants

When privacy-technique = "none" (raw coordinates):

Field	Type	Required	Description
lat	number (float64)	Yes	Latitude, WGS-84 decimal degrees
lon	number (float64)	Yes	Longitude, WGS-84 decimal degrees
accuracy	number (float64)	Yes	Accuracy radius in meters

Table 4

geolocation-proof-hash = Base64URL(SHA-256(JCS({lat, lon, accuracy})))

When privacy-technique = "zkp" (zero-knowledge proof):

Field	Type	Required	Description
zkp-proof-uri	string (URI)	Yes	URI to fetch full ZKP proof bytes from the proof depository. Verifier fetches bytes, computes SHA-256(bytes), checks against geolocation-proof-hash.
zkp-format	string enum	Yes	ZKP proof system. Currently: "plonky2".

Table 5

geolocation-proof-hash = Base64URL(SHA-256(zkp-proof-bytes))

5.3.4. MNO Endorsement (RATS Endorsement)

The mno-endorsement is a RATS Endorsement {{!RFC9334}}: a signed statement from a third party (the Mobile Network Operator) about the Attester’s location. The MNO attests device location within carrier visibility but does not sign host-level fields. This element is OPTIONAL at the top level; when present, its fields are REQUIRED.

Field	Type	Required	Description
mno-key-cert	string (Base64URL DER)	Yes	MNO signing certificate. Verifiers SHOULD validate this certificate chains to a known MNO root before accepting the endorsement.
mno-sig	string (Base64URL)	Yes	ECDSA/EdDSA signature over JCS(geolocation-payload) only. The MNO attests location within carrier visibility — does not sign host fields (tpm-ak, nonce, tpm-quote-seal).

Table 6

5.3.5. Workload Identity Binding

The workload object binds the V-GAP Evidence to a specific workload identity, enabling the Verifier to associate platform and residency claims with the credential being issued.

Field	Type	Required	Description
workload-id	string (SPIFFE ID)	Yes	The workload's SPIFFE identity URI (e.g., spiffe://example.org/python-app).
key-source	string	Yes	Identifier for the origin of the workload's key material (for example, "tpm-app-key"). Values are deployment-specific; this field is recorded for audit and policy evaluation.

Table 7

5.4. TPM Quote Verification Procedure

The Verifier MUST perform the following steps to validate the tpm-quote-seal:

1. Decode tpm-quote-seal (Base64URL → bytes)
2. Parse TPMS_ATTEST structure
3. Assert TPMS_ATTEST.type == TPM_ST_ATTEST_QUOTE
4. Compute expected_qd = SHA-256(JCS({tpm-ak, geolocation-id-hash, geolocation-proof-hash, privacy-technique, nonce, timestamp, workload-identity-agent-image-digest}))
5. Assert TPMS_ATTEST.qualifyingData == expected_qd
6. Verify signature over TPMS_ATTEST bytes using tpm-ak public key (RSASSA-PKCS1-v1_5 or ECDSA)

If any step fails, the Verifier MUST reject the Evidence and MUST NOT produce a positive Attestation Result.

5.5. Freshness and Replay Prevention

To prevent mix-and-match and replay attacks, Verifiers MUST enforce the following:

* Attestation Results MUST be fresh and MUST be bound to the credential issuance event (for example, by cryptographically binding freshness values used for platform quotes and workload credential issuance within the Verifier result).
* The nonce field in the lah-bundle MUST be a freshness value issued by the Relying Party (Workload Identity Management Plane) for each attestation interval.
* Verifiers MUST reject Evidence where the timestamp falls outside the configured freshness window.

Where policy requires it, the Verifier can additionally require that an agent software measurement (for example, image digest) is covered by validated platform Evidence, reducing the risk that a modified or unauthorized agent obtains credentials.

6. Security Considerations

V-GAP reduces reliance on spoofable location signals and stolen tokens by making integrity and residency cryptographically verifiable. Implementers still need to address the following threats:

* *Replay and mix-and-match*: Use nonces and evidence stapling so that old location evidence cannot be combined with a fresh platform quote (or vice versa).
* *Location spoofing*: Treat sensor and network inputs as adversarial. Prefer multiple, corroborating sources where feasible, and apply conservative policy when evidence quality degrades.
* *Relay and displacement*: When proximity mechanisms are introduced in future profiles, implementers should be aware that they are vulnerable to relay attacks and anchor displacement. Mitigations (such as tight RTT-based acceptance windows and anchor health attestation) are deferred to those future profiles.
* *Management plane compromise*: OOB paths reduce dependence on the host OS but introduce dependence on the management controller and its network. Protect this plane with secure boot, authenticated updates, strong access controls, network segmentation, and audit logging.
* *Time and freshness*: Verifiers MUST enforce bounded freshness windows and MUST define recovery behavior (re-attestation, quarantine, or revocation) when clocks drift or evidence becomes stale.
* *Registry and allowlist integrity*: Protect key registries and policy stores against tampering; treat them as high-value privileged assets.

* *Privacy*: Avoid unnecessary collection or retention of precise location data. Prefer "in-zone" proofs (ZKP) where policy permits.

7. IANA Considerations

IANA is requested to register the following Object Identifier (OID) in the "SMI Numbers" registry under the "SMI Private Enterprise Numbers" (1.3.6.1.4.1) branch, or as appropriate for the V-GAP profile.

* *OID*: 1.3.6.1.4.1.65284.1.1
* *Description*: Verifiable Geofencing Attestation Profile (V-GAP) Evidence Bundle
* *Reference*: This document.
* *PEN*: 65284 (IANA Private Enterprise Number assigned to Ram Krishnan)

Appendix A. Operational Guidance

A.1. Gating Credentials on Verified Evidence

This profile assumes two cooperating control planes, mapped to RATS roles:

* *Verifier (Host Identity Management Plane):* Appraises platform integrity and residency Evidence and produces an Attestation Result.
* *Relying Party + Credential Issuer (Workload Identity Management Plane):* Issues or renews workload identities (for example, SVIDs) only when the Attestation Result satisfies policy. Also acts as the CA that signs the resulting credential.

In intermittently connected edge deployments, local operation can continue during outages, while centralized policy can be enforced on renewal and on release of high-value secrets once connectivity is available.

A.2. Distributed Identity Issuance and Scaling

To support edge deployments and intermittent connectivity, identity issuance may be distributed within a sovereign boundary.

* *Edge issuance*: Workload identities (for example, SVIDs) may be issued by an issuer deployed within the same boundary as the workloads.

* *Scoping*: Issued identities should be scoped so they are not accepted outside the intended deployment boundary (for example, via trust bundle partitioning and policy).
* *Renewal gating*: Issuers should renew short-lived identities only when the Verifier result for integrity and residency is valid for the requested freshness window.

A.3. Mobility and Sovereign Handover

When a workload moves between anchors or boundaries, the Workload Identity Agent should obtain a new V-GAP bundle that reflects the new LAH and current residency.

Verifiers should treat this as a normal re-attestation event: - platform integrity continuity can remain stable, but - residency checks should be re-evaluated for the new anchor/boundary.

A.4. Location Anchor Hosts

To scale location sensing, a deployment may use dedicated anchors:

* *End-user anchors*: A user device (for example, a phone) can serve as an LAH for a nearby client device. The mechanism by which the anchor establishes its own location (and any proximity evidence it may provide) is out of scope for this document.
* *Data center anchors*: A small set of hosts can act as LAHs for a cluster. Timing-based mechanisms (for example, PTP-derived) may assist in establishing relative location; protocol details are deferred to future profiling work (see {{I-D.ramki-ptp-hardware-rooted-attestation}}).

Appendix B. Scalable Fleet Management

Large deployments need lifecycle management for the attestation keys referenced by V-GAP (for example, tpm-ak) and for the policies that authorize them.

B.1. Nonce Chain and Merkle Audit Log

One way to satisfy the freshness requirements in this profile is through a chained nonce and Merkle audit log. Where bundle[n] denotes the JCS-canonicalized lah-bundle object at attestation interval n:

```
chain[n] = SHA-256(chain[n-1] || SHA-256(JCS(bundle[n])))
nonce[n]  = HMAC(secret, n || chain[n-1])
```

Mechanism	Role
Chained nonce	Input control — agent cannot submit without responding to the management plane's current state.
Merkle chain	Audit output — proves inclusion of past bundles, detects gaps, and enables regulatory audit.

Table 8

B.2. Key Registry and Synchronization

* A Cloud (central) Verifier (Host Identity Management Plane) maintains a registry of accepted AK public keys and associated metadata (for example, EK certificate chain, hardware identity, and status).
* An Edge Verifier may maintain a local registry to support disconnected operation and periodically synchronizes updates to the central registry.

B.3. Key Rotation

To prevent rogue key injection during rotation:

* The central registry should accept a new AK only if the edge plane provides a rotation proof that chains the new AK to previously accepted state.
* A rotation proof should be a JCS-canonicalized object signed by the previously accepted AK (or, if available, validated by a fresh hardware-rooted OOB quote).

B.3.1. Example Rotation Proof

```
{
 "new-ak-pub": "Base64URL_Encoded_Public_Key",
 "serial-number": "AK_Serial_XYZ",
 "timestamp": 1708845600,
 "hardware-uuid": "Host_Hardware_UUID",
 "signature": "Base64URL_Signature_from_Previous_AK"
}
```

B.4. Credential Activation and Re-Verification

Credential activation (for example, TPM2_MakeCredential) is expensive to run on every request. Verifiers should perform it on events such as:

* Initial onboarding
* Reboot / reset detection (for example, TPM clock/reset counters)
* Policy violations or drift signals (for example, firmware or inventory changes)
* Failure of location evidence checks
* Explicit elevation to higher assurance policy

Between full activations, Verifiers may accept fresh quotes from registered AKs as proof of continued compliance, subject to policy.

B.5. Revocation and Health Signals

* The edge plane should maintain a per-node health signal (for example, tamper, firmware policy violations).
* On severe health signals, the Verifier should revoke the relevant AK(s) and reject identities derived from them according to policy.

B.6. Disconnected Operation (Leased Identity)

For intermittent connectivity, the Verifier may issue identities with extended validity (a lease) under policy. If a lease is used:

* The edge plane should revoke or refuse renewal locally on tamper/drift signals.
* The workload should re-attest and satisfy current policy on reconnection before renewal or release of high-value secrets.

Appendix C. Deployment Patterns

Implementations commonly fall into the following patterns, differing in how platform integrity Evidence and the tpm-quote-seal are collected:

* *In-band host attestation*: Evidence collected by host software (for example, Keylime-style deployments). In this pattern, the Relying Party (for example, SPIRE Server) generates N_fusion and shares it with the Verifier (for example, the Keylime Verifier) over a server-to-server channel. The Keylime Verifier then delivers N_fusion to the Keylime Agent running on the host, which collects TPM and geolocation evidence, assembles the lah-bundle, and returns it via the host-side channel. This pattern is well-suited to commodity servers and cloud VMs where a BMC path is not available or not required.

* *Out-of-band management*: Evidence collected via a management controller / BMC path (for example, iLO-class OOB management such as HPE OneView). In this pattern, the Relying Party (for example, SPIRE Server) generates N_fusion and shares it with the Verifier

(for example, HPE OneView) over a server-to-server channel. The Verifier delivers N_fusion to the host via the BMC / OOB path — bypassing the host OS entirely. The host TPM seals the lah-bundle with that nonce, and the sealed bundle is returned via the same OOB path. This pattern is recommended for high-assurance environments where the host OS is part of the threat model.

* *Cloud-hosted attestation environments*: Provider mechanisms exposing measured boot and TPM-backed claims (for example, Nitro-class enclaves or shielded VM instances). The cloud provider supplies a hardware-rooted quote that can serve as the tpm-quote-seal; the geolocation claim is typically derived from the provider's zone or region attestation. Implementations should verify that the provider's attestation scope satisfies the geofence policy.

Appendix D. Policy Use

Relying parties and credential issuers can use V-GAP Attestation Results as inputs to authorization.

* *ABAC*: Residency and integrity can be mandatory claims for sensitive operations.
* *KMS gatekeeping*: Release of high-value assets (for example, decryption keys) should depend on a recent successful verification result.
* *Fail closed*: If V-GAP Evidence is carried in an X.509 extension and marked CRITICAL, any implementation that does not understand the extension will reject the credential.

Appendix E. V-GAP Examples and Sensor Recipes

E.1. Example Instance (privacy-technique = "zkp")

```
   {
     "lah-bundle": {
       "tpm-ak": "-----BEGIN PUBLIC KEY-----\nMIIBIjANBgkqhkiG...\n-----END PUBLIC KEY---
--",
       "geolocation-id-hash": "7f4a2c1b9e3d8f0a6b5c4d2e1f0a9b8c...",
       "geolocation-proof-hash": "c8bc2ed62a7a650d99e0884197cdf345...",
       "privacy-technique": "zkp",
       "geolocation-payload": {
         "zkp-proof-uri": "https://verifier.example/v1/proof/c8bc2ed6...",
         "zkp-format": "plonky2"
       },
       "nonce": "ZmUyZjdmMzlmZGVlZWQxOTM1YjY0Mjk0...",
       "timestamp": 1740693456,
       "tpm-quote-seal": "ARoAAQALAAUACwEA...",
       "workload-identity-agent-image-digest": "a1b2c3d4e5f6...64-char-hex-sha256"
     },
     "mno-endorsement": {
       "mno-key-cert": "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A...",
       "mno-sig": "MEYCIQDx9z2k..."
     },
     "workload": {
       "workload-id": "spiffe://example.org/python-app",
       "key-source": "tpm-app-key"
     }
   }
```


E.2. Sensor Type Input Recipes

The following recipes define how geolocation-id-hash is constructed from different sensor types. The Verifier sees only the opaque hash — never the raw identifiers.

Sensor Type	geolocation-id-hash Input
Mobile (CAMARA)	SHA-256(tpm-ak-bytes || IMEI-bytes || IMSI-bytes)
GNSS receiver	SHA-256(tpm-ak-bytes || sensor-serial-bytes || sensor-class-id-bytes)

Table 9

Appendix F. Implementation Status

[Note to RFC Editor: This section may be removed before publication as per {{!RFC7942}}.]

A reference implementation of the V-GAP profile is publicly available:

* *Repository*: https://github.com/lfedgeai/AegisSovereignAI (https://github.com/lfedgeai/AegisSovereignAI)
* *Path*: hybrid-cloud-poc/
* *License*: Apache 2.0

The implementation demonstrates the *in-band host attestation* deployment pattern ({{deployment-patterns-informative}}) using:

* *TPM 2.0* hardware root of trust (AK-based quotes, PCR 15 TOCTOU protection)
* *SPIRE* (Relying Party + Credential Issuer) with a custom unifiedidentity plugin that embeds the lah-bundle as an X.509 extension (OID 1.3.6.1.4.1.65284.1.1)
* *Keylime* (Verifier) with IMA measurement of the SPIRE agent binary (workload-identity-agent-image-digest)
* *Plonky2* STARK prover for privacy-technique = "zkp" geofence proofs
* *Geolocation sensor cascade*: Mobile/CAMARA, GNSS/GPS, and config-file fallback with IMEI/IMSI binding for geolocation-id-hash

The implementation includes automated end-to-end tests (./run-demo.sh) that exercise the full attestation flow from TPM quote construction through ZKP proof generation and SVID issuance with embedded V-GAP Evidence.

Appendix G. Data Residency References

India -- Reserve Bank of India (RBI): Payment System Data Localization (2018): From RBI Circular RBI/2017-18/153 (April 6, 2018): "All system providers shall ensure that the entire data relating to payment systems operated by them are stored in a system only in India. This data should include the full end-to-end transaction details / information collected / carried / processed as part of the message / payment instruction."

South Korea’s Data Localization Regulations -- Geospatial Information Management Act (Spatial Data Act): Article 16, Paragraph 1: Prohibits the export of state-led survey data.

Authors’ Addresses

Ram Krishnan
JPMorgan Chase & Co.
Email: ramkri123@gmail.com

Ned Smith
Intel
Email: ned.smith@intel.com

Diego R. Lopez
Telefonica
Email: diego.r.lopez@telefonica.com

A Prasad
Oracle
Email: a.prasad@oracle.com

Srinivasa Addepalli
Aryaka
Email: srinivasa.addepalli@aryaka.com