RATS
Internet-Draft
Intended status: Standards Track
Expires: 24 November 2026

R. Krishnan
JPMorgan Chase & Co.
N. Smith
Intel
D. Lopez
Telefonica
A. Prasad
Oracle
S. Addepalli
Aryaka
23 May 2026

Verifiable Proof of Environment Attestation Profile
draft-lkspa-rats-verifiable-geo-fence-02

Abstract

Operators of regulated, sovereign, and high-assurance deployments require hardware-rooted, machine-verifiable proof that a workload executes in its approved environment. Current remote attestation mechanisms address two relevant properties in isolation: platform integrity — that the hardware and software stack are in an approved, untampered state — and physical residency — that the hardware resides within an approved geographic boundary. Neither property alone is sufficient: integrity without residency permits a valid platform to operate outside approved boundaries; residency without integrity permits a compromised platform to claim valid placement.

This document defines the *Verifiable Proof of Environment Attestation Profile (V-PEA)*, a profile of the RATS Architecture {{!RFC9334}} that fuses both properties into a single TPM-sealed Evidence structure (lah-bundle). V-PEA defines two Evidence dimensions: *WHAT* — hardware provenance (TPM Attestation Key registered and manufacturer-endorsed), platform integrity (firmware and OS state matching reference values), and workload agent software integrity (binary digest matching an approved value); and *WHERE* — physical residency within an approved geographic boundary. A TPM quote seal binds WHAT and WHERE into a single unforgeable statement: neither dimension can be forged or transplanted without invalidating the other.

For the WHERE dimension, V-PEA supports Transparent Zero-Knowledge Proofs (ZKPs), enabling an Attester to prove geographic compliance without disclosing precise coordinates. A positive V-PEA Attestation Result enables a Relying Party to issue hardware-rooted credentials or authorize operations — combining verified execution environment (WHAT) with verified physical placement (WHERE) — before releasing sensitive assets or granting access. Integration with workload identity systems is described in the WIMSE Integration appendix.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 24 November 2026.

Copyright Notice

Table of Contents

1. Introduction

Operators of sovereign and high-assurance workloads need cryptographic assurance that sensitive computation occurs only on approved, untampered hardware within approved geographic boundaries. Current attestation and location mechanisms address these concerns in isolation and incompletely. The gaps fall into two categories:

* *WHAT — incomplete execution environment verification:* Platform attestation typically confirms hardware presence (e.g., "a valid TPM is present") without verifying the workload agent binary itself or binding platform state (firmware, OS boot chain) explicitly to the issued credential. A compromised or substituted agent binary may still pass basic node attestation.

* *WHERE — unverifiable physical residency:* Geographic placement is recorded as an administrative label or inferred from network signals (IP geolocation) that are trivially spoofed via VPNs or proxies. There is no mechanism to prove "inside the approved zone" with a hardware root of trust, without disclosing precise coordinates, or without trusting an intermediary.

This document defines the Verifiable Proof of Environment Attestation Profile (V-PEA), a profile of the RATS Architecture {{!RFC9334}} that provides hardware-rooted Evidence for both dimensions. V-PEA enables a Verifier to appraise Evidence that:

1. *WHAT:* the Workload Identity Agent (Target Environment) is running on an approved, manufacturer-endorsed TPM whose platform state (PCRs) and agent binary digest match Reference Values — establishing hardware provenance, platform integrity, and software integrity; and

2. *WHERE:* that platform is physically resident within an approved geographic boundary, optionally without revealing precise coordinates (privacy-preserving residency via ZKP).

WHAT and WHERE together provide the cryptographic basis for a Relying Party to issue credentials or authorize operations. For integration with specific workload identity systems, see the WIMSE Integration appendix.

1.1. Scope and Layered Attestation

V-PEA profiles a two-layer Attester, following the layered attestation model defined in {{!RFC9334}}, Section 3.2. Using the terminology of RFC 9334 Figure 3:

Layer	RFC 9334 Role	V-PEA Entity	Responsibility
Layer A (immutable root)	Attesting Environment	TPM + Location Sensor (Claim source)	Establishes WHAT (hardware provenance): manufacturer-endorsed TPM identity (EK cert). Seals all Claims — platform state (PCRs), agent binary digest, and geolocation — into a single TPM quote, fusing WHAT and WHERE. MNO location statements (mno-location), when present, are integrated as a signed Claim source for WHERE.
Layer B (measured agent)	Target Environment + Evidence assembler	Workload Identity Agent	Subject of WHAT verification: its binary digest (target-environment-image-digest) is measured by the TPM. Collects Claims from the TPM and location sensor(s), constructs the lah-bundle, and conveys Evidence to the Verifier.

Table 1

The binding of individual workloads to the local Workload Identity Agent — and the credential issuance that follows a positive Attestation Result — are out of scope for this profile. Those concerns are addressed by complementary work such as {{I-D.mw-wimse-transitive-attestation}} and the WIMSE Architecture {{I-D.ietf-wimse-architecture}}. See the WIMSE Integration appendix for a mapping.

1.2. Relationship to Related Work

defines how a Verifier encodes geographic location conclusions — jurisdiction-level results such as country, subdivision, and city — as EAT Attestation Result Claims for consumption by a Relying Party. That draft addresses the *output encoding* side of the attestation pipeline.

V-PEA addresses the complementary *input side*: the Evidence profile the Attester produces, the hardware-binding mechanism (TPM quote) that makes location evidence verifiable, and the verification procedure the Verifier applies to produce an Attestation Result. V-PEA Evidence is what a Verifier appraises to yield the kind of geographic result claims that {{I-D.richardson-rats-geographic-results}} encodes.

The two documents are intended to compose: a Verifier that appraises a V-PEA lah-bundle could express its conclusion as an Attestation Result using the geographic Claims defined in {{I-D.richardson-rats-geographic-results}}. V-PEA is self-contained; use of that encoding is OPTIONAL and is one possible way a Verifier may express its conclusions — consumers MAY enforce geofence policy directly from the Attestation Result, use the V-PEA X.509 extension (OID 1.3.6.1.4.1.65284.1.1) as the trust signal, or adopt any other result encoding their deployment requires.

One gap in the combined stack is not addressed by either document: the mapping from a raw location fix or geofence proof to a named legal jurisdiction (for example, from "inside polygon P" to "in jurisdiction X"). This mapping raises its own trust questions — who maintains the polygon-to-jurisdiction database, under what authority, and how that mapping is kept current — and is deferred to future work.

1.2.1. Platform Ownership and Confidential Computing

Intel’s Platform Ownership Endorsement (POE) architecture enables remote parties to establish who is in physical possession of the hardware running workloads, using CoRIM-formatted endorsements tied to Platform Instance Identities (PIIDs). POE addresses a complementary concern to V-PEA: POE establishes platform *ownership* (who controls the hardware), while V-PEA establishes platform *residency* (where the hardware is). In deployments using Intel SGX or Intel TDX, a POE could serve as an additional Endorsement consumed by the V-PEA Verifier, strengthening confidence that the Attester is both owned by an approved party and resident within an approved boundary.

1.2.2. Geolocation Methods as Composable Claim Sources

V-PEA is designed so that different geolocation methods (GNSS, MNO/CAMARA, timing-based, provider region attestation, latency-bounded anchor networks such as SovCert, and emerging quantum-derived location proofs) can independently produce Claims that feed into a common Evidence structure and ultimately into a common Attestation Result format. Each method has distinct security properties and threat models (see Security Considerations). Implementations SHOULD treat geolocation methods as composable and independently appraised Claim sources rather than requiring a single method. This design allows the security considerations for each method to be evaluated independently while the Attestation Result remains uniform.

1.2.3. Other Potentially Related Work

The "Sovereign Certificates" initiative (sovcert.org) proposes latency-bounded location inference using dedicated network anchors. The initiative appears to be a single-entity effort without broad community review or standardization process participation. Nonetheless, the SovCert proposal is structurally compatible with V-PEA: SovCert "anchors" produce signed location measurements that are functionally equivalent to other geolocation Claim sources (GNSS, MNO/CAMARA, timing-based). A SovCert location statement could be integrated into the V-PEA Evidence structure as an additional signed Claim source, following the same composable model used for MNO location evidence (see mno-location). Future revisions of this document will assess formal integration if the initiative matures or publishes through a recognized standards body.

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
* *MNO*: Mobile Network Operator
* *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-PEA*: Verifiable Proof of Environment 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 agents and services execute only on approved hosts within an approved geographic boundary.

Attesting Environment: An environment capable of collecting Claims about a Target Environment and producing Evidence. In V-PEA, the TPM serves as the Attesting Environment.

Target Environment: An environment about which Claims are collected

by an Attesting Environment. In V-PEA, the Workload Identity Agent is the Target Environment measured by the TPM.

Workload Identity Agent: On-host component (Target Environment) whose binary integrity is measured by the TPM. Once verified, it assembles the V-PEA Evidence structure.

Location Anchor Host (LAH): Host or device that acts as the Attester ({{!RFC9334}}). Contains the TPM (Attesting Environment), the Workload Identity Agent (Target Environment), and one or more geolocation Claim sources (for example, GNSS receiver or MNO-connected modem).

Claim Source (Geolocation): A sensor or service that provides geolocation Claims (for example, coordinates, ZKP proof) to the Evidence assembler. The Claim source is not an Attesting Environment; its output is sealed into the TPM quote alongside platform integrity Claims.

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

Proof of Residency (PoR): Conclusion that a platform resides within an approved geofence boundary for a specific attestation interval, as determined by a Verifier appraising V-PEA Evidence.

Silicon Root of Trust: Hardware trust anchor (for example, TPM) that supports measured boot, protects attestation keys, and acts as the Attesting Environment.

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

V-PEA (Verifiable Proof of Environment Attestation Profile): RATS Evidence profile defined in this document. Fuses WHAT (hardware provenance, platform integrity, workload agent software integrity) and WHERE (verified physical residency, optionally via ZKP) into a single TPM-sealed Evidence structure (lah-bundle), providing the cryptographic basis for issuing a verified workload identity credential (WHO).

N_fusion: Fresh nonce issued by the Relying Party for each attestation interval. Corresponds to the nonce field in the lah-bundle. Provides freshness per {{!RFC9334}}, Section 10.

4. Use Cases

This profile supports hardware-rooted attestation of execution environments for platforms running workload agents and (optionally) user devices. Use cases span server-centric enforcement, user-centric enforcement, and compliance and risk reduction.

4.1. Server-centric Enforcement

Enterprises need cryptographic proof that identity agents run only on approved hosts within an approved geographic boundary, and that credentials are issued only from verified platforms.

* *Platform-to-platform (general):* Relying Parties accept credentials only when the issuing Attester’s Evidence demonstrates platform integrity and "in-zone" residency, preventing credentials from being used outside the approved boundary.

* *Agentic AI platforms:* An AI agent platform may issue credentials for sensitive operations only when its Attester 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 Attester demonstrates integrity and residency within the required boundary. This is useful for intermittently connected sites.

* *Server verification:* Clients validate that a server endpoint is operating within an approved boundary (e.g., by policy tied to the Attestation Result for that server’s platform).

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

V-PEA 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 Proof of Environment Attestation Profile (V-PEA)

V-PEA is a profile of the RATS Architecture {{!RFC9334}} that fuses hardware-rooted Evidence of WHAT and WHERE into a single TPM-sealed attestation structure. WHAT — the Workload Identity Agent (Target Environment) is the approved, unmodified binary running on an approved, untampered TPM platform. WHERE — the platform physically resides within an approved geographic boundary. The Attester (Location Anchor Host) produces a lah-bundle that a Verifier appraises; a positive Attestation Result authorizes the Relying Party to issue credentials or authorize operations — but only when both WHAT and WHERE pass. For integration with specific workload identity systems, see the WIMSE Integration appendix.

5.1. RATS Role Mapping

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

RATS Role ({{!RFC9334}})	V-PEA Entity	Function
Attester	Location Anchor Host (LAH)	Contains the Attesting Environment (TPM) and Target Environment (Workload Identity Agent). Produces V-PEA Evidence (the lah-bundle), including TPM quotes and geolocation Claims.
Verifier	Verifier (for example, Keylime Verifier or	Appraises V-PEA Evidence — validates TPM quotes, checks PCRs against Reference Values, verifies geolocation proofs — and produces

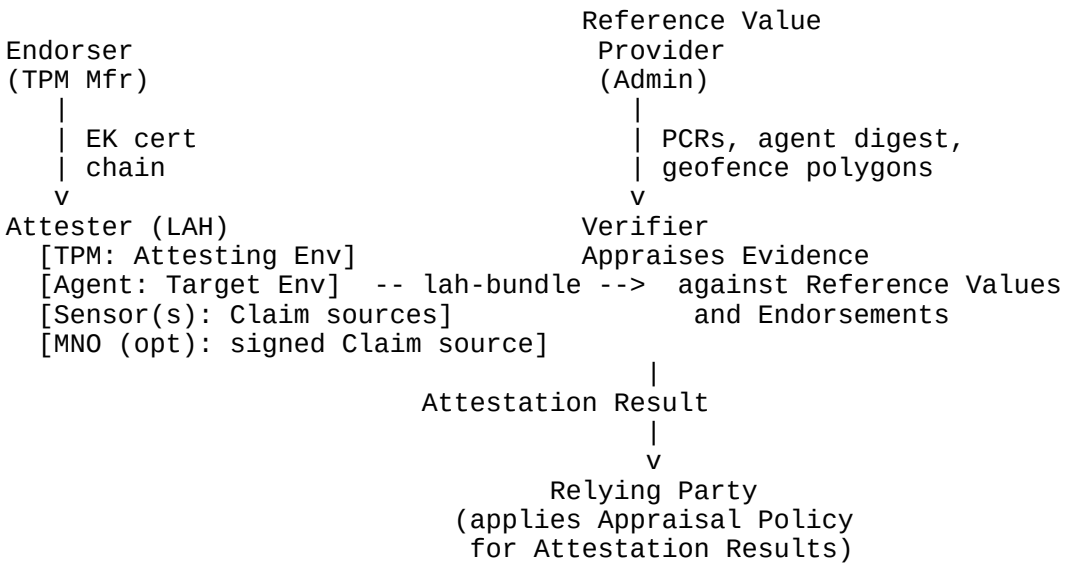
	HPE OneView)	an Attestation Result.
Endorser	TPM Manufacturer	Provides Endorsements (EK certificate chain) vouching for the TPM’s identity and signing capability.
Reference Value Provider	Platform administrator or supply chain entity	Supplies Reference Values: known-good PCR values, approved agent binary digests (target-environment-image-digest), and geofence boundary definitions.
Relying Party	Credential issuer or policy decision point	Consumes the Attestation Result and applies its Appraisal Policy for Attestation Results to decide whether to issue credentials, release keys, or authorize operations.
Verifier Owner	Security administrator	Configures the Appraisal Policy for Evidence: freshness windows, required trust levels, approved PCR sets, and geofence policy.
Relying Party Owner	Policy administrator	Configures the Appraisal Policy for Attestation Results: which Verifiers are trusted, minimum result freshness, and required Claims in the Attestation Result.

Table 2

Note: The Mobile Network Operator (MNO), when present, provides a signed location statement (mno-location) that the Attester integrates into its Evidence as a signed Claim source. This is attester-collected Evidence — not a RATS Endorsement — because the MNO asserts network-observed location, not the Attester’s identity or characteristics. The same model applies to other external signed location sources (such as SovCert anchors or quantum-derived proofs), which MAY similarly be integrated as signed Claim sources within the lah-bundle.

5.2. Evidence Flow

The V-PEA Evidence flow follows the RATS background-check model ({{!RFC9334}}, Section 5.2): the Attester conveys Evidence to the Verifier (possibly via a Relying Party), the Verifier appraises it using Reference Values and Endorsements, and conveys the Attestation Result to the Relying Party.



The Attestation Result produced by the Verifier is the output of V-PEA's RATS pipeline. What the Relying Party does with the Attestation Result — issue a credential, release a key, authorize an operation — is a Relying Party policy decision outside the scope of this profile. See the WIMSE Integration appendix for one such consumption pattern.

5.3. V-PEA Evidence Structure

The lah-bundle is the RATS Evidence structure defined by this profile. It is a hardware-sealed object produced by the Attester (LAH) and conveyed to the Verifier for appraisal. It encodes both Evidence dimensions: WHAT (hardware provenance via tpm-ak and manufacturer endorsement; platform integrity via PCRs in the TPM quote; software integrity via target-environment-image-digest) and WHERE (geofence residency via geolocation-proof-hash and geolocation-payload, optionally as a privacy-preserving ZKP). All fields are fused by the tpm-quote-seal into a single TPM-signed statement — neither dimension can be selectively forged or transplanted without invalidating the seal.

5.3.1. Top-Level Structure

```
{
  "lah-bundle": { },
  "mno-location": { }
}
```

When present, the lah-bundle fields are serialized using JSON Canonicalization Scheme (JCS) for hash computation. In the bundle, tpm-ak is carried as a PEM-encoded public key string, but hash inputs such as tpm-ak-bytes are derived from the raw DER bytes of the same public key.

5.3.2. lah-bundle Fields

Field	Type	Required	Description
tpm-ak	string (PEM)	Yes	TPM Attestation Key public key (PEM-encoded, -----BEGIN PUBLIC KEY----- format). Hardware identity anchor. The TPM enforces that only this key can produce tpm-quote-seal — proving the quote was produced by the same physical hardware as the geolocation sensor.
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 via an out-of-band channel (for example, hardware inventory or supply

			chain attestation).
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-location.mno-sig.
nonce	string (Base64URL)	Yes	Freshness nonce (N_fusion) issued by the Relying Party for each attestation interval, per {{!RFC9334}}, Section 10.2. 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, target-environment-image-digest})). Fuses WHAT (hardware identity: tpm-ak; platform state: PCRs; software integrity: target-environment-image-digest) and WHERE (geolocation-id-hash, geolocation-proof-hash) into a single hardware-sealed statement. Neither dimension can be forged or transplanted without invalidating this seal.
target-environment-image-digest	string (hex SHA-256)	Yes	SHA-256 digest of the Target Environment (Workload Identity Agent) binary, measured at attestation time. This digest is computed over the measured binary image bytes or artifact bytes that the TPM records. Compared by the Verifier against Reference Values to detect agent binary compromise.

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 Location Evidence (Signed Claim Source)

The mno-location element carries a signed location statement from a Mobile Network Operator (MNO). In RATS terms, this is attester-collected Evidence — a signed Claim source — rather than a RATS Endorsement: the MNO asserts network-observed device location within carrier visibility but does not vouch for the Attester's identity or platform characteristics. 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 location statement.
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

5.4. Attestation Result

Upon successful appraisal of the lah-bundle, the Verifier produces an Attestation Result ({{!RFC9334}}, Section 8.4). This profile does not mandate a specific encoding for the Attestation Result. Implementations MAY express results using:

* EAT Attestation Result Claims, including geographic Claims per {{I-D.richardson-rats-geographic-results}};

* an X.509 extension (OID 1.3.6.1.4.1.65284.1.1) embedded in a credential issued by a Relying Party acting as CA; or

* any other result encoding that satisfies the Relying Party's Appraisal Policy for Attestation Results.

The Attestation Result MUST convey at minimum:

Claim	Description
Platform integrity	Whether the TPM quote was valid and PCR values matched Reference Values.
Agent integrity	Whether target-environment-image-digest matched a known-good Reference Value.
Residency	Whether the geolocation proof (raw or ZKP) satisfied the configured geofence policy.
Freshness	The attestation interval (nonce and timestamp) for which the result is valid.
Trust level	The location trust level achieved (see Location Trust Levels).

Table 7

When the Attestation Result is embedded in an X.509 extension and marked CRITICAL, any downstream consumer that does not understand the extension MUST reject the credential, enforcing fail-closed behavior.

5.5. 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, target-environment-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.

5.6. 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 appraisal event (for example, by cryptographically binding freshness values used for platform quotes within the Attestation Result).

* The nonce field in the lah-bundle MUST be a freshness value issued by the Relying Party for each attestation interval, per the nonce-based freshness model in {{!RFC9334}}, Section 10.2.

* Verifiers MUST reject Evidence where the timestamp falls outside the configured freshness window.

Where policy requires it, the Verifier can additionally require that the Target Environment measurement (target-environment-image-digest) matches an approved Reference Value, reducing the risk that a modified or unauthorized agent produces accepted Evidence.

6. Security Considerations

V-PEA provides hardware-rooted assurance of both WHAT (approved, untampered execution environment) and WHERE (approved physical residency), enabling a Relying Party to issue credentials or authorize operations only when both dimensions pass appraisal. The security of issued credentials is only as strong as the weakest of these two dimensions. Implementers must 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*: GNSS and mobile network signals must be treated as adversarial inputs; per-source threats and mitigations are detailed in the subsections below.

* *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 controller compromise*: OOB paths reduce dependence on the host OS but introduce dependence on the management controller and its network. Protect this component 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 Reference Value stores and Appraisal Policy configurations 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. See the Privacy Applicability note below.

6.1. Privacy Applicability

The relevance of location privacy varies significantly by deployment context:

* *Datacenter and server environments*: When an Attester is a server in a known datacenter, the physical location of the hardware is typically not sensitive — it may be a matter of public record or contractual documentation. In such deployments, privacy-technique = "none" (raw coordinates) is appropriate and the ZKP overhead is unnecessary.

* *User-facing and edge environments*: When an Attester is a user device or edge node, precise coordinates may constitute Personally Identifiable Information (PII). In such deployments, privacy-technique = "zkp" SHOULD be used to prove geofence compliance without disclosing exact location.

* *Mixed deployments*: Appraisal Policy for Evidence SHOULD allow the Verifier Owner to configure which privacy technique is acceptable per Attester class or per geofence policy.

Implementers SHOULD select the privacy technique appropriate to their deployment context rather than applying ZKP uniformly.

6.2. Location Spoofing

6.2.1. Fundamental Limitation: Sensor Data Provenance

It is important to acknowledge that binding location Claims to a TPM quote (hardware provenance) does NOT, by itself, guarantee that the underlying sensor data is correct. A TPM can faithfully seal whatever data the sensor provides — including spoofed data. The TPM proves that the sealed data came from the measured platform; it does not prove that the sensor's input signals were authentic.

Therefore, the security of V-PEA's geolocation Claims depends on BOTH:

1. *Hardware provenance* (addressed by the TPM quote and target-environment-image-digest): ensuring the data was processed by an approved, untampered platform.

2. *Sensor input integrity* (addressed by the mitigations below): ensuring the sensor received authentic signals rather than spoofed or replayed inputs.

Implementers MUST NOT rely solely on TPM binding as evidence of correct location. The Appraisal Policy for Evidence SHOULD require corroborating evidence from independent channels and SHOULD specify minimum signal authentication requirements commensurate with the geofence policy's sensitivity.

6.2.2. GNSS Spoofing

GNSS signals are unauthenticated by default and can be spoofed via synthetic signal generators (e.g., software-defined radio replay of valid signals) or multipath injection. Implementers SHOULD apply mitigations proportional to the required assurance level:

* *Signal authentication*: Galileo OSNMA (Open Service Navigation Message Authentication) provides cryptographic authentication of navigation messages and is the strongest available civilian countermeasure. GPS GAIA offers equivalent protection for GPS III signals. Implementations SHOULD prefer authenticated GNSS signals where available.

* *Multi-constellation cross-validation*: Cross-checking fixes across independent constellations (GPS, Galileo, GLONASS, BeiDou) substantially raises the cost of spoofing; consistent simultaneous spoofing of all constellations requires significantly more attacker capability.

* *Anomaly detection*: Sudden position jumps, implausible velocity changes, and anomalous signal-to-noise ratios are indicators of spoofing or jamming. Evidence that fails these checks SHOULD be rejected.

6.2.3. Mobile Network Spoofing

Mobile network location evidence is subject to distinct threats:

* *IMSI catchers and rogue base stations*: Attacker-controlled base stations can force a device onto a fake cell, yielding attacker-controlled location if evidence derives from device-reported cell identity.

* *SS7/Diameter abuse*: Attackers with access to legacy carrier signaling can issue location queries that yield false or manipulated carrier-side location data.

* *MNO root key compromise*: The mno-location element is only as trustworthy as the MNO signing root. Verifiers MUST validate the mno-key-cert certificate chain to a known MNO root and SHOULD treat a root compromise as requiring immediate policy revocation.

The CAMARA API model — where location is derived from carrier network infrastructure rather than device-reported cell identifiers — is more resistant to IMSI catcher attacks and is the RECOMMENDED approach when MNO corroboration is used.

Notwithstanding these mitigations, MNO-derived location is ultimately under the control of the carrier infrastructure. A compromised or coerced MNO can produce false location statements. Verifiers SHOULD treat MNO location statements as corroborating evidence rather than sole proof of residency, and Appraisal Policies SHOULD require independent corroboration (for example, GNSS + MNO) for high-assurance geofence policies.

6.2.4. Location Trust Levels

The quality indicator defined in Composite Geolocation SHOULD be mapped to a location trust level enforced by the Verifier as a precondition for a positive Attestation Result. The following non-normative tiers illustrate a conformant policy:

Trust Level	Evidence Basis
Low	Single unauthenticated GNSS fix, no corroboration
Medium	Multi-constellation GNSS with anomaly detection, or network-side MNO corroboration (CAMARA) alone
High	Authenticated GNSS (OSNMA or GAIA), or Medium GNSS + MNO corroboration
Highest	Authenticated GNSS + independent network-side MNO corroboration (CAMARA) + anomaly detection

Table 8

The security value of multi-source corroboration derives from *channel independence*: GNSS and MNO evidence travel over different physical and logical channels. Requiring consistent evidence from both simultaneously raises the bar for spoofing. Verifiers SHOULD require a minimum trust level commensurate with the sensitivity of the enforced geofence policy, and SHOULD apply conservative policy (downgrade or reject the Attestation Result) when evidence quality degrades.

6.3. Zero-Knowledge Proof Security

V-PEA's privacy-technique = "zkp" mode uses Plonky2 proofs (a STARK-based proof system using FRI commitments). The following properties and threats apply:

* *Circuit correctness is the primary attack surface.* A ZKP proves only what its arithmetic circuit encodes. Errors in the geofence boundary circuit — including precision errors, off-by-one boundary conditions, or incorrect coordinate system handling — yield proofs that are cryptographically valid but semantically incorrect. The geofence circuit MUST be independently audited before deployment.

* *No trusted setup.* Plonky2 is STARK-based and requires no trusted setup phase, eliminating the class of attacks arising from compromised SNARK setup parameters. Implementations substituting a different zkp-format MUST ensure it also provides transparent setup, or MUST document the resulting trust assumptions.

* *Computational soundness.* STARK security is computational, not unconditional, and relies on the collision resistance of the underlying hash function. Implementations SHOULD target at least

128-bit security and MUST document the proof system parameters (field size, hash function, FRI parameters) to enable independent security analysis.

* *URI availability.* When privacy-technique = "zkp", the Verifier MUST reject Evidence if the zkp-proof-uri cannot be resolved or the fetched proof bytes do not match geolocation-proof-hash.

* *Proof freshness.* A valid ZKP proves location at proof-generation time. The nonce and timestamp freshness requirements that apply to tpm-quote-seal apply equally to ZKP proofs: Verifiers MUST reject proofs whose timestamp falls outside the configured freshness window.

* *Prover integrity.* A compromised prover can produce false proofs even for a correctly specified circuit. This threat is mitigated by V-PEA's TPM binding: the tpm-quote-seal covers geolocation-proof-hash, so a false proof can only be embedded in a bundle that also passes TPM quote verification for an approved platform. The ZKP privacy guarantee is only meaningful in conjunction with verified platform integrity.

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-PEA profile.

* *OID*: 1.3.6.1.4.1.65284.1.1

* *Description*: Verifiable Proof of Environment Attestation Profile (V-PEA) Evidence / Attestation Result

* *Reference*: This document.

* *PEN*: 65284 (IANA Private Enterprise Number assigned to Ram Krishnan)

8. References

8.1. Normative References

* {{!RFC9334}}

* {{!RFC2119}}

* {{!RFC8174}}

* {{I-D.richardson-rats-geographic-results}}

* {{I-D.mw-wimse-transitive-attestation}}

* {{I-D.ietf-wimse-architecture}}

8.2. Informative References

* {{!RFC7942}}

* {{I-D.ramki-ptp-hardware-rooted-attestation}}

Appendix A. Contributors

The following individuals have contributed to this document:

Bala Siva Sai Akhil Malepati
Independent
Email: saiakhil2012@yahoo.com

Ghada Arfaoui
Orange
Email: ghada.arfaoui@orange.com

Michael Epley
Red Hat
Email: mepley@redhat.com

Vijay Masilamani
Independent
Email: saanvijay20@gmail.com

Appendix B. Operational Guidance

B.1. Gating Decisions on Attestation Results

A Relying Party consumes the Attestation Result produced by the Verifier and applies its Appraisal Policy for Attestation Results to make application-specific decisions. Common decision types include:

* *Credential issuance*: Issue or renew a workload credential only when the Attestation Result satisfies policy.

* *Key release*: Release decryption keys or model weights only to Attesters with a positive, fresh Attestation Result.

* *Access authorization*: Gate access to sensitive APIs or data stores on a valid Attestation Result.

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.

B.2. Distributed Credential Issuance and Scaling

To support edge deployments and intermittent connectivity, credential issuance by a Relying Party may be distributed within a sovereign boundary.

* *Edge issuance*: Credentials may be issued by a Relying Party deployed within the same boundary as the Attesters.

* *Scoping*: Issued credentials should be scoped so they are not accepted outside the intended deployment boundary (for example, via trust bundle partitioning and policy).

* *Renewal gating*: Relying Parties should renew short-lived credentials only when the Attestation Result for integrity and residency is valid for the requested freshness window.

B.3. Mobility and Handover

When an Attester moves between anchors or boundaries, the Target Environment (Workload Identity Agent) should trigger a new V-PEA attestation cycle 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 Claims should be re-evaluated against the geofence policy for the new anchor/boundary.

B.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 C. Scalable Fleet Management

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

C.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 — Attester cannot submit without responding to the Verifier’s current state.
Merkle chain	Audit output — proves inclusion of past bundles, detects gaps, and enables regulatory audit.

Table 9

C.2. Key Registry and Synchronization

* A central Verifier 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.

C.3. Key Rotation

To prevent rogue key injection during rotation:

* The central registry should accept a new AK only if the Edge Verifier 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).

C.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"
}
```

C.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.

C.5. Revocation and Health Signals

* The Edge Verifier 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.

C.6. Disconnected Operation (Leased Attestation Result)

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

* The Verifier should revoke or cease producing positive results locally on tamper/drift signals.

* The Attester should re-attest and satisfy current policy on reconnection before the Relying Party accepts a new Attestation Result or releases high-value secrets.

Appendix D. 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 generates N_fusion and shares it with the Verifier (for example, the Keylime Verifier) over a server-to-server channel. The Verifier then delivers N_fusion to the Attester on the host, which collects TPM and geolocation Claims, 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 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 E. Policy Use

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

* *ABAC*: Residency and integrity Claims from the Attestation Result can be mandatory attributes for sensitive operations.

* *KMS gatekeeping*: Release of high-value assets (for example, decryption keys) should depend on a recent, positive Attestation Result.

* *Fail closed*: When the Attestation Result is embedded in an X.509 extension marked CRITICAL, any consumer that does not understand the extension will reject the credential.

Appendix F. V-PEA Examples and Sensor Recipes

F.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...",
    "target-environment-image-digest": "a1b2c3d4e5f6...64-char-hex-sha256"
  },
  "mno-location": {
    "mno-key-cert": "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A...",
    "mno-sig": "MEYCIQDx9z2k..."
  }
}
```

F.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 10

Appendix G. Implementation Status

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

A reference implementation of the V-PEA 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) with a custom unifiedidentity plugin that consumes V-PEA Attestation Results and embeds them as an X.509 extension (OID 1.3.6.1.4.1.65284.1.1)

* *Keylime* (Verifier) with IMA measurement of the Target Environment binary (target-environment-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 Attestation Result consumption.

Appendix H. WIMSE Integration

This appendix describes how a WIMSE deployment consumes V-PEA Attestation Results. The mapping is informative and does not constrain V-PEA's RATS profile.

H.1. Relationship to WIMSE Architecture

The WIMSE Architecture {{I-D.ietf-wimse-architecture}} defines a credential issuance and workload identity framework. V-PEA produces Attestation Results that WIMSE credential issuers consume as trust inputs. The following mapping shows how V-PEA RATS roles correspond to WIMSE entities:

V-PEA RATS Role	WIMSE Entity	Function
Attester (LAH)	Platform hosting the SPIRE Agent	Produces V-PEA Evidence about the agent and platform.
Verifier	Platform integrity service (for example, Keylime)	Appraises Evidence and produces Attestation Results.
Relying Party	SPIRE Server (Credential Issuer / CA)	Consumes the Attestation Result; issues or renews X.509-SVIDs only when the result satisfies its Appraisal Policy for Attestation Results.
(out of V-PEA scope)	Workload	Receives its credential (for example, SVID) from the SPIRE Agent via transitive attestation {{I-D.mw-wimse-transitive-attestation}}.
(out of V-PEA scope)	Downstream service consumer (mTLS peer)	Consumes the issued credential; trusts the CA signature as proxy for verified integrity and residency.

Table 11

H.2. Workload Binding Fields

In a WIMSE deployment the Relying Party (SPIRE Server) may require additional context to associate the Attestation Result with a specific credential issuance event. The following fields are carried outside the V-PEA Evidence structure, typically in the credential issuance request or as Relying Party policy inputs:

Field	Type	Description
workload-id	string (SPIFFE ID)	The workload's SPIFFE identity URI (for example, spiffe://example.org/python-app).
key-source	string	Identifier for the origin of the workload's key material (for example, "tpm-app-key"). Deployment-specific.

Table 12

These fields are not part of V-PEA Evidence. They are consumed by the Relying Party when applying its Appraisal Policy for Attestation Results.

H.3. X.509 Extension for Downstream Consumers

When the WIMSE Relying Party acts as a CA (for example, SPIRE Server issuing X.509-SVIDs), it MAY embed V-PEA Attestation Result information as an X.509 extension (OID 1.3.6.1.4.1.65284.1.1). Implementations SHOULD mark this extension as CRITICAL so that any downstream consumer that does not understand it will reject the credential, enforcing fail-closed behavior for residency-constrained workloads.

Appendix I. 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