Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 29 November 2026

L. Melegassi
Catellix

28 May 2026

MVPS Proof Envelope: Tamper-Evident Binding of Theorem Catalogues, Validators, and Numerical Receipts, with an Optional Post-Quantum Profile

draft-melegassi-ippm-mvps-proof-envelope-00

Abstract

The Multi-Vantage Path Snapshot (MVPS) family specifies coherence detection algebra [MVPS-V4] and trust profiles that authenticate vantage reports [I-D.melegassi-santos-ippm-mvps-cwt]. The mathematical truth of the MVPS theorems is established off the wire, in companion proof documents and machine validators that emit numerical receipts. Without a normative binding layer, a consumer cannot verify WHICH theorem catalogue, proof revision, validator script, or receipt artifact a given deployment claim relies on, nor detect silent substitution of those artifacts.

This document specifies the MVPS Proof Envelope: a canonicalized [RFC8785] manifest that lists theorem identifiers and the SHA-256 digests of the companion proof documents, validator scripts, and numerical receipts that support them, aggregated under a Merkle root [RFC9162] and anchored by the operator-epoch and witness-cosignature machinery of [I-D.melegassi-santos-ippm-mvps-cwt]. Verification of the envelope is tamper-evident under standard hash and signature assumptions (Theorem T-BIND-1) and traceable (Theorem T-TRACE-1).

The envelope does NOT cryptographically prove the listed theorems; a proved theorem is a mathematical fact, not subject to cryptanalysis. The envelope makes the CHOICE OF PROOF ARTIFACTS tamper-evident. An OPTIONAL Post-Quantum Protection profile MAY replace the Ed25519 anchor signatures with ML-DSA-65 [FIPS204] while leaving the per-snapshot HMAC-SHA256 hot path unchanged (Theorem T-PQ-MIG-1). Security Considerations document a finite Grover-halved floor and explicitly defer any claim of perpetual security.

Numerical receipts are regenerated by scripts/validate_proof_envelope.py (11/11 PASS) and stored in evidence/proof_envelope_receipt.json. Companion proofs are in [PROOF-ENVELOPE-PROOF].

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 29 November 2026.

Copyright Notice

Table of Contents

1. Introduction

The MVPS family [MVPS-V4] couples a coherence-detection algebra with a set of mathematical companion documents (existence proofs and lemmas) and machine validators that regenerate numerical receipts. The trust profiles [I-D.melegassi-santos-ippm-mvps-trust] and [I-D.melegassi-santos-ippm-mvps-cwt] authenticate the MEASUREMENT bundles. No existing profile authenticates the PROOF ARTIFACTS themselves.

A consumer auditing an MVPS deployment is therefore exposed to a gap: it can verify that a bundle was produced by an admitted vantage, but it cannot verify that the proof document, validator, and receipt it was handed are the genuine, unmodified artifacts the author published. An adversary -- or an honest mistake -- could substitute a weakened proof, a tampered validator that always exits 0, or a doctored receipt, and the consumer would have no normative way to

detect it.

This document closes that gap with a single canonical, tamper-evident object: the Proof Envelope.

1.1. Motivation: Binding Artifacts, Not Proving Theorems

It is essential to state precisely what cryptography can and cannot do here.

A mathematical theorem, once proved, is a fact. It is not "broken" by any computer, classical or quantum; cryptanalysis has no purchase on a proof. What CAN be attacked is the integrity of the artifacts that carry the proof to a reader: the files can be altered, swapped, or forged in transit or at rest.

The Proof Envelope binds those artifacts. It guarantees, under standard hash and signature assumptions, that the proof, validator, and receipt a consumer verifies are exactly the ones the author published. This is the honest, defensible, and IETF-verifiable statement, and it is the only one this document makes.

1.2. What This Document Does and Does Not Claim

DOES:

- Bind theorem catalogues, proof documents, validators, and receipts into one Merkle-rooted, signed manifest (T-BIND-1).
- Guarantee that every transported theorem identifier resolves to a named catalogue document (T-TRACE-1).
- Permit migration of the anchor signature to a post-quantum scheme without altering the per-snapshot hot path (T-PQ-MIG-1).

DOES NOT:

- Cryptographically prove any MVPS theorem. Proofs live in their companion documents and are validated by their own scripts.
- Claim "unbreakable" or "perpetual" security. The guarantee is conditional on SHA-256 collision resistance and anchor EUF-CMA, with a documented FINITE security floor (Section 8).
- Make Ed25519 quantum-safe. The optional PQ profile (Section 6) RAISES the floor; it does not remove it.

1.3. Relationship to the MVPS Family

The Proof Envelope is a Profile-of-Profiles, layered above:

- the bundle format [I-D.melegassi-ippm-mvps-bundle],
- the Coherent-Witness Trust profile [I-D.melegassi-santos-ippm-mvps-cwt] (preferred anchor), and
- optionally the Performance-Security Coupling profile [I-D.melegassi-mvps-perfsec-coupling].

It reuses the JSON canonicalization and per-file digest manifest pattern of the SNAP archival profile [I-D.melegassi-dispatch-mvps-snap-backup].

2. Terminology and Conventions

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.

Artifact:
: A file that carries an MVPS proof, validator, or receipt, identified by a repository-relative name and a SHA-256 digest.

Proof Manifest:
: A canonical JSON document listing a catalogue, a set of theorem identifiers, and the name-sorted artifact list.

Manifest root:
: The Merkle tree root [RFC9162] over the manifest's artifact leaves.

Envelope:
: A Proof Manifest, its manifest root, and an anchor signature over that root (plus, when CWT is present, witness cosignatures).

Catalogue:
: A named, versioned theorem set (e.g., "mvps-v4.0", "cwt-v1").

Registry:
: The map from theorem identifier to the document that proves it, realized by [MVPS-FOUNDATIONS].

3. Proof Manifest Format (Normative)

3.1. Artifact Object

Each artifact is a JSON object:

```
{
  "kind":   "proof" | "validator" | "receipt",
  "name":   "<repository-relative path, forward slashes>",
  "sha256": "<lowercase hex, 64 chars>"
}
```

The sha256 field is the SHA-256 digest of the artifact's raw bytes.

3.2. Canonical Order and Merkle Root

The artifact list MUST be sorted in ascending order by the UTF-8 bytes of the "name" field. This canonical order is REQUIRED: the Merkle root is position-sensitive (Theorem T-BIND-2), so verifiers MUST sort before hashing to obtain a reproducible root.

Each leaf is computed as:

```
leaf_i = SHA-256( 0x00 || JCS({ "name": name_i,
                                "sha256": sha256_i }) )
```

where JCS is the JSON Canonicalization Scheme [RFC8785]. Interior nodes use the RFC 9162 separator:

```
node = SHA-256( 0x01 || left || right )
```

An odd trailing node is promoted unchanged to the next level. The

manifest root is the root of this tree.

3.3. Theorem Identifiers and Catalogue

The manifest carries:

```
{
  "manifest_version": 1,
  "catalog": ["mvps-v4.0", "cwt-v1", "perfsec-v1",
              "proof-envelope-v1"],
  "theorem_ids": ["T-BIND-1", "T-COAL-1", "T1", ...],
  "artifacts": [ ... name-sorted ... ],
  "issued_at": "<RFC3339Z>"
}
```

Every identifier in "theorem_ids" MUST resolve to a document in the registry [MVPS-FOUNDATIONS]. An envelope carrying an unresolved identifier MUST be rejected (Theorem T-TRACE-1).

4. Envelope and Verification Procedure

4.1. Envelope Object

```
{
  "manifest": { ... Section 3 ... },
  "merkle_root_sha256": "<hex>",
  "anchor": {
    "scheme":    "ed25519" | "ml-dsa-65",
    "public_key":"<base64url>",
    "signature": "<base64url over the 32-octet root>"
  },
  "cwt_checkpoint": "<optional; CWT Section 8 checkpoint whose
                      'extra' field carries merkle_root_sha256>"
}
```

4.2. Verify-Envelope

A consumer accepts an envelope E iff ALL of the following hold:

1. The manifest artifact list is in canonical name-sorted order (Section 3.2).

2. For every listed artifact the consumer can access, the recomputed SHA-256 of the file equals the listed "sha256".

3. The Merkle root recomputed from the artifact list equals "merkle_root_sha256".

4. The anchor "signature" verifies over the 32-octet root under "public_key" and "scheme".

5. Every identifier in "theorem_ids" resolves in the registry.

6. When CWT integration is REQUIRED by campaign policy, the "cwt_checkpoint" carrying the root is q-witnessed per [I-D.melegassi-santos-ippm-mvps-cwt] Section 8.4.

Failure of any step: the envelope MUST be rejected and the failure logged. A rejected envelope MUST NOT be used to attest a deployment claim.

5. Integration with Coherent-Witness Trust

When deployed with CWT, the manifest root is placed in the "extra" field of the bundle checkpoint ([I-D.melegassi-santos-ippm-mvps-cwt] Section 8.2) and is therefore covered by the witness cosignatures. The envelope then inherits:

- Split-view resistance (CWT T-SPLIT-1): no two distinct accepted manifest roots for the same (bundle_id, bundle_seq).
- Coalition resistance (CWT T-COAL-1): substituting a manifest without corrupting a witness quorum has probability at most (b/w)^q.

This composition is Theorem T-COMP-PE in [PROOF-ENVELOPE-PROOF].

6. Optional Post-Quantum Protection Profile

The base profile signs the manifest root with Ed25519. The OPTIONAL Post-Quantum Protection profile substitutes ML-DSA-65 [FIPS204] at the anchor layer (operator epoch key and witness keys).

By Theorem T-PQ-MIG-1, the per-snapshot HMAC-SHA256 tag of [I-D.melegassi-santos-ippm-mvps-cwt] Section 6 is a function of the session key and the snapshot message only; it does NOT depend on the anchor signature scheme. A campaign therefore migrates to post-quantum protection by reissuing only the operator-epoch and witness keys under ML-DSA-65; vantages, session keys, and the per-snapshot wire format are untouched (Corollary T-PQ-MIG-1.1).

This profile RAISES the long-horizon security floor of the anchor layer. It does not, and cannot, claim perpetual security; see Section 8.

Implementations advertising this profile MUST use the capability flag "pq-protection-v1" (Section 10) and MUST document the ML-DSA parameter set in use.

7. Relationship to v4.0, CWT, PerfSec, SNAP

- [MVPS-V4]: the envelope binds, but does not re-prove, the v4.0 catalogue (T1..T9).
- CWT: provides the anchor signature and (optionally) the witness cosignatures that carry the manifest root.
- PerfSec [I-D.melegassi-mvps-perfsec-coupling]: when present, its proof and receipt artifacts are simply additional manifest entries.
- SNAP [I-D.melegassi-dispatch-mvps-snap-backup]: the per-file SHA-256 manifest and JCS canonicalization pattern are reused.

8. Security Considerations

This entire document is security-focused. Specific notes:

- Tamper-evidence (T-BIND-1) reduces to SHA-256 collision resistance [RFC6234] and anchor EUF-CMA [RFC8032] / [FIPS204].
- The guarantee is CONDITIONAL and FINITE. Under Grover’s algorithm the SHA-256 preimage floor is 2^128 and the collision floor is approximately 2^85; both remain infeasible but are explicitly bounded. This document makes NO claim of unconditional

or perpetual ("unbreakable forever") security.

- The mathematical theorems bound by the envelope are facts established by proof; they are not within the reach of cryptanalysis at all. Cryptanalysis can only target the binding, and the PQ-Protection profile (Section 6) is the mechanism by which the binding's floor is raised against future quantum adversaries.

- Anchor-key compromise allows manifest re-signing for the compromised key only; when CWT witness cosignatures are present, a quorum of independent witnesses still binds consumers to a single root (CWT T-SPLIT-1).

- A verifier that cannot access a listed artifact verifies only the artifacts it holds; absence is not substitution, but a complete audit REQUIRES access to every listed artifact.

9. Privacy Considerations

The manifest lists file names and digests of proof artifacts, which are intended to be public. Operators MUST NOT list artifacts whose names or contents reveal private deployment topology; the redaction guidance of [I-D.melegassi-ippm-mvps-bundle] applies.

10. IANA Considerations

This document requests registration of two MVPS Bundle Capability Flags in the registry defined by [I-D.melegassi-ippm-mvps-bundle]:

Flag name: proof-envelope-v1
Semantics: Bundle carries a Proof Envelope (Section 4) whose manifest root is anchored per Section 4.1. Consumers MUST run Verify-Envelope (Section 4.2) before relying on any transported theorem claim.

Flag name: pq-protection-v1
Semantics: The envelope anchor signature uses ML-DSA-65 [FIPS204] per Section 6.

If the base registry is not yet established, implementations SHOULD use these strings in a bundle-level "capability_flags" array until IANA assignment.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, May 2011.

[RFC8032] Josefsson, S. and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)", RFC 8032, January 2017.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, May 2017.

[RFC8785] Rundgren, A., Jordan, B., and S. Erdtman, "JSON Canonicalization Scheme (JCS)", RFC 8785, June 2020.

[RFC9162] Laurie, B., Messeri, E., and R. Stradling, "Certificate Transparency Version 2.0", RFC 9162, December 2021.

[FIPS204] National Institute of Standards and Technology, "Module-Lattice-Based Digital Signature Standard (ML-DSA)", FIPS 204, August 2024.

[I-D.melegassi-ippm-mvps-bundle] Melegassi, L., "Multi-Vantage Path Snapshot (MVPS): A Canonical Bundle Format for Coordinated Traceroute Measurements", draft-melegassi-ippm-mvps-bundle-00, May 2026.

[I-D.melegassi-santos-ippm-mvps-cwt] Melegassi, L. and J. A. Santos, "MVPS Trust Profile: Coherent-Witness Trust", draft-melegassi-santos-ippm-mvps-cwt-00, May 2026.

11.2. Informative References

[MVPS-V4] Melegassi, L., "MVPS Mathematical Existence Proof -- Version 4.0", Catellix technical note, May 2026.

[PROOF-ENVELOPE-PROOF] Melegassi, L., "MVPS Proof Envelope -- Mathematical Existence Proof", Catellix technical note, May 2026.

[MVPS-FOUNDATIONS] Melegassi, L., "MVPS IETF Foundations: Theorem Traceability Table", Catellix technical note, May 2026.

[I-D.melegassi-santos-ippm-mvps-trust] Melegassi, L. and J. A. Santos, "MVPS Trust Profile", draft-melegassi-santos-ippm-mvps-trust-00, May 2026.

[I-D.melegassi-mvps-perfsec-coupling] Melegassi, L., "MVPS Performance-Security Coupling Profile", draft-melegassi-mvps-perfsec-coupling-00, May 2026.

[I-D.melegassi-dispatch-mvps-snap-backup] Melegassi, L., "MVPS SNAP: Atomic JSON Backup", draft-melegassi-dispatch-mvps-snap-backup-01, May 2026.

Appendix A. JSON Manifest Example (Informative)

```
{
  "manifest_version": 1,
  "catalog": ["mvps-v4.0", "cwt-v1", "proof-envelope-v1"],
  "theorem_ids": ["T-AUTH-CWT-1", "T-BIND-1", "T-COAL-1",
                  "T-TRACE-1", "T1", "T9"],
  "artifacts": [
    { "kind": "proof",
      "name": "docs/MVPS_CWT_MATHEMATICAL_PROOF.txt",
      "sha256": "..." },
    { "kind": "validator",
      "name": "scripts/validate_cwt_theorems.py",
      "sha256": "..." },
    { "kind": "receipt",
      "name": "evidence/mvps_cwt_overhead_receipt.json",
      "sha256": "..." }
  ],
  "issued_at": "2026-05-28T00:00:00Z"
}
```

Appendix B. Mathematical Core (Normative)

This appendix states the envelope theorems. Full proofs appear in [PROOF-ENVELOPE-PROOF]; constructive witnesses are validated by scripts/validate_proof_envelope.py (11/11 PASS), which regenerates evidence/proof_envelope_receipt.json.

THEOREM T-BIND-1 (Tamper-evidence).
Under SHA-256 collision resistance and anchor EUF-CMA, any content alteration of a bound artifact accepted by Verify-Envelope occurs only with negligible probability.

THEOREM T-BIND-2 (Canonical order necessity).
The Merkle root is position-sensitive; the name-sorted order yields a reproducible root across verifiers.

THEOREM T-TRACE-1 (Traceability).
Every theorem identifier in an accepted envelope resolves to a named catalogue document; no accepted envelope asserts an orphan theorem.

THEOREM T-PQ-MIG-1 (Hot-path invariance under anchor migration).
The per-snapshot HMAC-SHA256 tag is invariant under replacement of the anchor signature scheme (Ed25519 -> ML-DSA-65); migration is local to the anchor layer.

THEOREM T-PQ-MIG-2 (Finite documented floor).
The SHA-256 / HMAC primitives retain a Grover-halved preimage floor of 2^128; the guarantee is finite and conditional, never perpetual.

NON-CLAIM (normative). No theorem in this appendix proves any MVPS detection theorem. The envelope binds artifacts; the proofs remain in their companion documents.

Author’s Address

Leonardo Melegassi
Catellix
Andradina, SP
Brazil

Email: melegassi@catellix.com
URI: https://catellix.com/