INTERNET-DRAFT
Expires: April 20, 2026

Internet Engineering Task Force (IETF)
Request for Comments: 0
Category: Informational
draft-ramki-ptp-hardware-rooted-attestation-00.txt
ISSN: 2070-1721

R. Krishnan
Vishanti Systems, Inc.
M. Richardson
Sandelman Software Works Inc
D. Lopez
Telefonica
A. Prasad
Oracle
S. Addepalli
Aryaka

17 October 2025

Hardware-Rooted Attestation for Precision Time Protocol: Verifiable Residency and Proximity proofs

Abstract

This document defines an extension to Precision Time Protocol (PTP) that provides per-event cryptographic attestation using non-exportable asymmetric keys resident in TPMs or HSMs, and an optional PTP-in-HTTPS/MTLS encapsulation mode. When combined with freshness and multi-observer correlation, this provides defensible proof of proximity for timing events. PTP-in-HTTPS/MTLS adds end-to-end confidentiality for timing payloads across untrusted fabrics.

NOTE: This note is a placeholder to show the correct structure that avoids the "setup generated" error (Line 53). All paragraphs inside a note must be separated by a blank line.

setup generated

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at https://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at https://www.ietf.org/shadow.html

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are candidates for any level of Internet

Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc0. Copyright Notice

Table of Contents

1. Introduction

Precise, auditable time provenance is increasingly required by regulated systems, distributed ledgers, event forensics, and safety-critical infrastructures. Existing symmetric PTP authentication primitives provide integrity but limited non-repudiation and fragile key distribution (e.g., https://www.ietf.org/id/draft-kumarvarigonda-ptp-auth-extension-00.html).

This draft specifies an asymmetric, TPM/HSM-backed attestation extension for PTP events plus an optional PTP-in-HTTPS/MTLS encapsulation mode. Goals are per-event provenance, replay resistance, staged deployability in heterogeneous environments, and practical offload to SmartNICs or HSMs to meet performance needs. The optional HTTPS/MTLS encapsulation adds end-to-end confidentiality to the integrity and provenance provided by signing.

2. Conventions and Definitions

PHC: Packet Hardware Clock exposed by NIC or SmartNIC.

TPM: Trusted Platform Module supporting non-exportable keys and Quote operations.

HSM: Hardware Security Module on SmartNIC or separate appliance.

Verifier: Service that validates signed tokens and records audit evidence.

Registrar: PKI/registry service binding signer_id to device identity, PCR profile, and revocation state.

Monotonic Counter: Non-decreasing hardware or TPM counter used to prevent replay.

HTTPS/MTLS: HTTP over TLS 1.3 with mutual TLS (client certificates) for endpoint authentication.

SmartNIC: Programmable NIC with PHC, crypto acceleration, and optionally on-card HSM.

3. Architecture Overview

In-band signed PTP extension PTP messages carry an attached signed token for each signed event. This mode preserves end-to-end integrity and provenance of PTP payloads (signature binds payload, PHC timestamp, nonce, seq, counter) while leaving confidentiality and in-fabric correction semantics to the underlying network fabric.

Hardware-rooted signing: PTP endpoints (masters, slaves, boundary clocks) are provisioned with non-exportable asymmetric keys in TPMs or HSMs. Each PTP event is signed using a Quote operation that includes a nonce and monotonic counter to prevent replay. The signer_id (e.g., key hash or certificate serial) is included in the signed token to allow verifiers to fetch the corresponding public key and PCR profile from a registrar service.

Note: In-band attestation preserves integrity and provenance but does not provide confidentiality; PTP payloads remain visible to in-path observers.

3.1. PTP-in-HTTPS/MTLS encapsulation

Native PTP bytes are framed inside persistent HTTPS/MTLS streams between endpoints. Signed tokens are carried inside the same MTLS connection or out-of-band to a verifier. This prevents in-path modification and adds confidentiality for timing payloads and signed metadata.

3.2. Signing Mechanism

Endpoints MUST compute event_digest over the entire PTP message as transmitted, except for fields explicitly designated as mutable by IEEE 1588 (e.g., correction field). When PTP messages are encapsulated in HTTPS/MTLS, endpoints SHOULD sign the entire PTP message without exclusions, as no in-path modification is permitted.

3.3. Verifier Roles

Two deployment patterns are supported for the verifier function:

Dedicated Verifier Service * A logically separate service issues nonces, validates signed tokens, checks counters, and records audit evidence. * Advantages: clear separation of duties, centralized audit logs, simplified revocation handling, and independence for regulatory or forensic review. * Normative requirements: * The dedicated verifier MUST maintain an append-only, tamper-evident audit log of all tokens and validation results. * The verifier MUST enforce nonce freshness, monotonic counter progression, and token TTL. * The verifier MUST reject tokens from revoked or unregistered Signer_IDs.

Peer-as-Verifier * A PTP peer (master, slave, or boundary clock) may

act as verifier by issuing nonces to its counterpart and validating returned signed tokens inline with the timing exchange. * Advantages: immediate freshness check, no extra infrastructure, lower latency. * Risks: blurs separation of duties, reduces independence of audit evidence, and increases reliance on peer trustworthiness. * Normative requirements: * A peer acting as verifier MUST log all signed tokens and validation results to an append-only audit store or forward them to a registrar. * A peer acting as verifier MUST apply the same validation rules as a dedicated verifier (nonce freshness, monotonic counter, TTL, revocation). * Operators SHOULD prefer independent verifiers when regulatory or forensic requirements demand separation of duties.

4. Signed Token Structure

The signed token is a CBOR map with the following fields. CBOR encoding (RFC 8949) is be used to ensure consistent signatures.

text ; Signed Token (CBOR map)) { 1 : uint, ; version (e.g., 1) 2 : uint, ; event_type (PTP message type) 3 : uint, ; ptp_seq (SequenceID) 4 : uint, ; phc_timestamp_ns (nanoseconds) 5 : bstr, ; event_digest (SHA-256 of signed PTP fields) 6 : bstr, ; nonce (verifier-issued, 16 bytes recommended) 7 : uint, ; monotonic_counter (TPM/HSM-backed) 8 : bstr, ; signer_id (hash of TPM/HSM public key or cert fingerprint) 9 : bstr / null, ; pcr_summary (optional TPM Quote or compressed PCR set) 10: bstr ; signature (TPM/HSM non-exportable key) } # PTP Message Signing Coverage The following table indicates which PTP fields MUST be included in the event_digest computation. Fields marked as mutable by IEEE 1588 (e.g., CorrectionField) are excluded in in-band mode. In PTP-in-HTTPS/MTLS mode, the entire PTP message MUST be signed since no in-path modification is permitted.

PTP Field (IEEE 1588 header)	Signed?	Rationale
TransportSpecific + MessageType	Yes	Immutable, identifies event type
VersionPTP	Yes	Immutable
MessageLength	Yes	Integrity of framing
DomainNumber	Yes	Integrity of domain separation
FlagField	Yes	Integrity of mode bits
CorrectionField	No	Mutable by transparent clocks; excluded in in-band mode
SourcePortIdentity	Yes	Binds to originating clock
SequenceID	Yes	Prevents replay/reordering
ControlField	Yes	Immutable
LogMessageInterval	Yes	Immutable
PTP Payload (Sync, FollowUp, DelayReq, DelayResp, etc.)	Yes	Except correction sub-fields if mutable
TLVs (other than Attestation)	Yes	Integrity of extensions

Table 1

Normative rule: * In in-band TLV mode, event_digest MUST be computed over the entire PTP message excluding CorrectionField (and any other fields normatively designated as mutable by IEEE 1588). * In PTP-in-HTTPS/MTLS mode, the entire PTP message MUST be included in the digest, since no in-path modification is permitted.

5. Security Considerations

Replay and relay attacks

Endpoints MUST include a verifier-issued nonce and a monotonic counter in each token.

Verifiers MUST:

* Reject tokens with stale or missing nonces.
* Reject tokens with regressions in monotonic counters.
* Reject tokens where counters jump beyond an operator-defined threshold.

Verifiers SHOULD log round-trip times (RTT) for challenge/response exchanges and MAY apply policy thresholds to detect relays or anomalous delays.

5.1. Confidentiality

In-band attestation TLVs provide integrity and provenance but do not provide confidentiality; PTP payloads remain visible to in-path observers.

Operators requiring confidentiality MUST use PTP-in-HTTPS/MTLS encapsulation, which prevents in-path modification and protects both timing payloads and attestation metadata.

5.2. PCR privacy

PCR values and TPM quotes may reveal sensitive configuration or software state.

Registrars MUST enforce minimal disclosure policies, requiring only the PCRs necessary for attestation policy.

Verifiers MUST validate PCR summaries against registrar policy but MUST NOT require disclosure of unrelated PCRs.

5.3. Revocation

Verifiers MUST reject tokens from revoked or unregistered Signer_IDs.

Registrars MUST support rapid revocation and distribution of revocation state to verifiers.

Operators MUST ensure revocation information is available to verifiers in near-real time.

5.4. Compromise handling

In the event of TPM/HSM compromise, operators MUST support re-enrollment and key rollover.

Registrars MUST provide mechanisms to bind new keys to existing device identities and to revoke compromised keys without disrupting

unaffected devices.

Audit logs MUST record revocation and re-enrollment events for forensic traceability.

5.5. Location claims

Verifiers MUST NOT assert geographic residency or location from a single signed timestamp.

Proximity proofs require correlation across multiple observers and RTT measurements.

6. IANA Considerations

A new PTP TLV type for the signed token.

A registry for token versions and signature algorithm identifiers.

7. References

Normative: IEEE 1588 (PTP), RFC 8949 (CBOR), RFC 8446 (TLS 1.3), TPM 2.0 spec, draft-kumarvarigonda-ptp-auth-extension.

Informative: draft-ietf-ntp-over-ptp, RATS architecture (RFC 9334), COSE (RFC 8152).

Acknowledgments

TODO acknowledge.

Authors' Addresses

Ramki Krishnan
Vishanti Systems, Inc.
Email: ramkri123@gmail.com

Michael Richardson
Sandelman Software Works Inc
Email: mcr+IETF@sandelman.ca

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