Remote ATtestation ProcedureS
Internet-Draft
Intended status: Informational
Expires: 12 July 2026

T. Kamimura
VeritasChain Standards Organization (VSO)

8 January 2026

On the Relationship Between Remote Attestation and Behavioral Evidence Recording

draft-kamimura-rats-behavioral-evidence-00

Abstract

This document provides an informational discussion of the conceptual relationship between remote attestation, as defined by the RATS (Remote ATtestation ProcedureS) Working Group, and behavioral evidence recording mechanisms. It observes that attestation and behavioral evidence recording address fundamentally different verification questions and can serve as complementary layers in comprehensive system accountability frameworks. This document is purely descriptive and does not propose any modifications to RATS architecture, define new attestation mechanisms, or establish normative requirements.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Remote ATtestation ProcedureS (RATS) Working Group mailing list (rats@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/rats/.

Source for this draft and an issue tracker can be found at https://github.com/veritaschain/draft-kamimura-rats-behavioral-evidence.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The IETF RATS (Remote ATtestation ProcedureS) Working Group has developed a comprehensive architecture for remote attestation, enabling Relying Parties to assess the trustworthiness of remote systems through cryptographic evidence about their state. This attestation capability addresses a fundamental question in distributed systems: "Is this system in a trustworthy state?"

A related but distinct verification need exists in many operational contexts: the ability to verify what actions a system has actually performed. This question - "What did the system actually do?" - is addressed by cryptographically verifiable audit trails, which record system behaviors and decisions in tamper-evident formats.

This document observes that these two verification capabilities address different aspects of system accountability and can conceptually complement each other without overlapping in scope or responsibility. The document provides an explanatory framework for understanding how attestation and audit trails could relate within broader accountability architectures.

1.1. Document Status

This document is purely INFORMATIONAL and NON-NORMATIVE. It:

* Does NOT propose any new RATS mechanisms or architecture changes
* Does NOT modify or extend the RATS architecture as defined in [RFC9334]
* Does NOT define new protocols, claims, tokens, or data formats
* Does NOT establish normative requirements for any implementation
* Remains fully compatible with and respectful of existing RATS concepts and design philosophy

The language in this document uses descriptive terms (MAY, COULD, CAN) exclusively to indicate possibilities and observations. This document does not use normative requirements language (MUST, SHOULD, SHALL) as there are no mandatory behaviors or requirements being specified.

This document treats verifiable audit trail systems in general terms, using VeritasChain Protocol (VCP) [VCP-SPEC] as one illustrative example among various possible approaches to cryptographic audit logging.

1.2. Motivation

Several domains increasingly require both trustworthiness assessment and behavioral verification:

* *Financial Services:* Algorithmic trading systems face regulatory requirements for both system integrity and decision audit trails

* *Artificial Intelligence:* AI governance frameworks increasingly distinguish between system certification and operational logging

* *Critical Infrastructure:* Systems controlling physical processes may require attestation of their configuration alongside records of their actions

Understanding the conceptual relationship between attestation and audit trails could help architects design accountability frameworks that leverage both capabilities appropriately, without conflating their distinct purposes.

2. Terminology

This document reuses terminology from the RATS Architecture [RFC9334] without modification. The following terms are used as defined in that document:

Attester A role performed by an entity that creates Evidence about itself and conveys it to a Verifier.

Verifier A role performed by an entity that appraises Evidence about an Attester.

Relying Party A role performed by an entity that uses Attestation Results to make authorization decisions.

Evidence Information about an Attester’s state, generated by the Attester.

Attestation Results Output produced by a Verifier, indicating the trustworthiness of an Attester.

Claims Assertions about characteristics of an Attester.

Reference Values Expected values against which Evidence is compared.

The following terms are used in this document to describe audit trail concepts in general terms:

Note on "Audit" Terminology: The term "audit" in this document refers solely to post-hoc examination of recorded system behavior. It does not imply regulatory auditing, compliance verification, or financial auditing in any jurisdiction-specific sense. This usage is consistent with general systems engineering terminology (e.g., "audit log") rather than domain-specific compliance frameworks.

Audit Event A recorded occurrence representing a discrete action, decision, or state change in a system. Not to be confused with RATS Evidence, which describes system state rather than behavior.

Audit Trail A chronologically ordered sequence of Audit Events that can be cryptographically verified for integrity and authenticity.

Behavioral Evidence Records of what a system has done, as distinct from Evidence about what state a system is in. This term is used to distinguish audit records from RATS Evidence.

3. Conceptual Layering

This section describes an observational framework for understanding how attestation and audit trails could conceptually relate as distinct layers of system accountability.

3.1. Attestation Layer (RATS)

The RATS architecture addresses trustworthiness assessment through remote attestation. At its core, attestation answers questions about system state:

* Is the system’s software configuration as expected?
* Is the system running on genuine, uncompromised hardware?
* Does the system’s current state match known-good reference values?
* Can the system’s identity and configuration be cryptographically verified?

These questions are fundamentally about the properties and characteristics of a system at a point in time or across a measurement period. The RATS architecture provides mechanisms for generating, conveying, and appraising Evidence that enables Relying Parties to make trust decisions about Attesters.

Key characteristics of attestation as defined by RATS:

* Focuses on system state and configuration

* Enables trust decisions before or during interactions

* Produces Attestation Results for Relying Party consumption

* Relies on hardware roots of trust where available

* Addresses the question: "Can I trust this system?"

3.2. Behavioral Evidence Layer (Audit Trails)

Cryptographically verifiable audit trails address a different category of verification need. Rather than assessing system state, audit trails record what a system has done:

* What decisions did an algorithm make?

* What actions did the system execute?

* What inputs led to what outputs?

* What was the sequence and timing of operations?

These questions are fundamentally about system behavior over time. Verifiable audit trails could provide mechanisms for recording, preserving, and proving the integrity of behavioral records, enabling after-the-fact verification of system actions.

Key characteristics of audit trail systems (in general terms):

* Focus on system behavior and decisions

* Enable verification after events have occurred

* Produce verifiable records for auditors, regulators, or other parties

* May rely on cryptographic structures such as hash chains, Merkle trees, or append-only logs

* Address the question: "What did this system do?"

As an illustrative example, VCP [VCP-SPEC] defines audit trails using three integrity layers: event integrity (hashing), structural integrity (Merkle trees), and external verifiability (digital signatures and anchoring). Other audit trail systems could employ different but analogous mechanisms to achieve similar goals.

3.3. Separation of Concerns

The distinction between attestation and audit trails can be understood as a separation of concerns:

Aspect	Attestation (RATS)	Audit Trails
Primary Question	Is this trustworthy?	What happened?
Focus	System state	System behavior
Temporal Scope	Point-in-time or measurement period	Historical record
Primary Consumer	Relying Party	Auditor, Regulator
Trust Question	Should I interact?	Did it behave correctly?

Table 1: Conceptual Comparison of Attestation and Audit Trails

This separation suggests that attestation and audit trails could serve as complementary layers rather than alternatives. A system could potentially be subject to both attestation (verifying its trustworthy state) and audit logging (recording its subsequent behavior), with neither capability substituting for the other.

4. Relationship Between Attestation Claims and Audit Evidence

4.1. Complementary Questions

When viewed together, attestation and audit trails could address a more complete accountability picture:

Attestation (Before/During): "This trading system’s software is the certified version, running on genuine hardware, with approved configuration."

Audit Trail (After): "This trading system generated signal X at time T1, submitted order Y at time T2, and received execution confirmation Z at time T3."

Neither layer fully substitutes for the other:

* A system could pass attestation but subsequently behave in unexpected ways that would only be visible through audit records

* A system could maintain a complete audit trail while itself being compromised in ways that attestation might detect

The combination could potentially provide stronger accountability than either mechanism alone.

4.2. Temporal Considerations

Attestation and audit trails may operate on different temporal rhythms:

Attestation Patterns:

* At system boot or initialization

* Periodically during operation

* On-demand when a Relying Party requests

Audit Trail Patterns:

* Continuously, as events occur

* At varying granularities depending on event significance

* With periodic integrity commitments (e.g., Merkle root anchoring)

A conceptual integration could involve attestation confirming system integrity at key moments, while audit trails continuously record behavior between those moments. This temporal complementarity means neither mechanism creates gaps that the other cannot fill, but together they could provide comprehensive temporal coverage.

4.3. Trust Anchors

Both attestation and audit trails ultimately rely on trust anchors, though potentially different ones:

Attestation Trust Anchors:

* Hardware roots of trust (TPM, TEE)

* Endorsement keys and certificates

* Reference value providers

Audit Trail Trust Anchors:

* Signing keys for audit records
* External anchoring mechanisms (transparency logs, timestamps)
* Verifier infrastructure for inclusion proofs

In some deployment scenarios, these trust anchors could overlap or share infrastructure. For example, a hardware security module used as an attestation root of trust could potentially also sign audit trail records. However, this document does not prescribe any particular trust anchor arrangement.

5. Example Conceptual Flow

This section provides a purely illustrative, non-normative example of how attestation and audit trails could conceptually complement each other in a hypothetical deployment. This example does not define any protocol or establish any requirements.

Consider a hypothetical automated trading system:

Phase 1: System Startup The trading system boots and undergoes remote attestation. A Verifier confirms that the system's software, configuration, and hardware environment match expected reference values. A Relying Party (such as an exchange) receives Attestation Results and permits the system to begin trading operations.

Phase 2: Operational Period During trading operations, the system generates audit events for each significant action: signal generation, order submission, execution acknowledgment, risk parameter changes. These events are recorded in a cryptographically verifiable audit trail, with periodic integrity anchoring.

Phase 3: Periodic Re-Attestation At configured intervals, the system may undergo re-attestation to confirm its state has not drifted. Audit trail entries created between attestation events provide behavioral evidence for that period.

Phase 4: Post-Hoc Verification An auditor later wishes to verify the system's behavior during a specific time window. The auditor can: (a) Check that valid Attestation Results existed for the relevant period, confirming the system was in a trustworthy state; (b) Verify the audit trail for that period, confirming what actions the system actually took; (c) Correlate both to form a comprehensive accountability view.

This example is purely conceptual and illustrative. Actual deployments would involve specific technical decisions not addressed in this document.

6. Non-Goals and Explicit Out-of-Scope Items

To maintain clarity about this document's limited scope, the following items are explicitly out of scope and are NOT addressed:

This document does NOT:

* Propose any modifications to the RATS architecture
* Define any new attestation mechanisms, Evidence formats, or Attestation Result structures
* Specify any protocol for connecting attestation and audit trails
* Define any new claims, tokens, or data structures
* Establish normative requirements for either attestation or audit trail implementations
* Mandate any particular audit trail format or mechanism
* Suggest that RATS should incorporate audit trail capabilities
* Propose work items for the RATS Working Group
* Define interoperability requirements between attestation and audit systems
* Specify trust relationships or delegation models

The purpose of this document is solely to observe and explain the conceptual relationship between attestation and audit trails as distinct but potentially complementary verification layers. Any technical integration or standardization work would require separate documents with appropriate community review.

7. Security Considerations

This document is purely informational and does not define any protocols or mechanisms. Therefore, it does not introduce new security considerations beyond those already present in the referenced specifications.

The following observations may be relevant to deployments that consider both attestation and audit trails:

Attestation Security: Security considerations for remote attestation are thoroughly addressed in [RFC9334]. These considerations remain unchanged by the conceptual observations in this document.

Audit Trail Security: Cryptographically verifiable audit trails face their own security considerations, including:

* Key management for signing audit records

* Protection against selective omission of events

* Integrity of external anchoring mechanisms

* Privacy considerations for sensitive behavioral data

Potential Interactions: If both attestation and audit trails are deployed together, architects could consider:

* Whether audit trail signing keys are themselves subject to attestation

* Whether audit trail infrastructure is included in attestation measurements

* How trust is established across both layers

Inter-Layer Trust Model: Attestation and behavioral evidence recording operate at different temporal granularities, creating distinct trust validation points:

* Attestation establishes trust at discrete moments (boot, periodic re-attestation, on-demand challenges)

* Behavioral evidence provides continuous records between attestation events

* A system could pass attestation validation yet subsequently exhibit unexpected behavior detectable only through behavioral records

* Conversely, a system could maintain complete behavioral records while being compromised in ways that attestation might detect

The combination of both mechanisms could provide defense-in-depth, but architects should consider the trust assumptions and failure modes of each layer independently.

Key Separation: Deployments using both attestation and behavioral evidence recording could benefit from separating cryptographic keys used for each purpose. Attestation keys (bound to hardware roots of trust) and behavioral evidence signing keys (potentially in separate security domains) may have different lifecycle, rotation, and compromise-response requirements.

This document does not mandate any particular approach to these considerations, which would depend on specific deployment contexts and threat models.

This document does not alter the RATS threat model, and introduces no new attack surfaces beyond those already considered by existing attestation and logging mechanisms.

8. IANA Considerations

This document has no IANA actions.

9. References

9.1. Normative References

[RFC9334] Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote ATtestation procedureS (RATS) Architecture", RFC 9334, DOI 10.17487/RFC9334, January 2023, <https://www.rfc-editor.org/info/rfc9334>.

9.2. Informative References

[I-D.ietf-scitt-architecture] Birkholz, H., Delignat-Lavaud, A., Fournet, C., Deshpande, Y., and S. Lasker, "An Architecture for Trustworthy and Transparent Digital Supply Chains", Work in Progress, Internet-Draft, draft-ietf-scitt-architecture, <https://datatracker.ietf.org/doc/html/draft-ietf-scitt-architecture>.

[I-D.ietf-rats-reference-interaction-models]
Birkholz, H., Eckel, M., Pan, W., and E. Voit, "Reference Interaction Models for Remote Attestation Procedures", Work in Progress, Internet-Draft, draft-ietf-rats-reference-interaction-models, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-reference-interaction-models>.

[RFC6962] Laurie, B., Langley, A., and E. Kasper, "Certificate Transparency", RFC 6962, DOI 10.17487/RFC6962, June 2013, <https://www.rfc-editor.org/info/rfc6962>.

[RFC9162] Laurie, B., Messeri, E., and R. Stradling, "Certificate Transparency Version 2.0", RFC 9162, DOI 10.17487/RFC9162, December 2021, <https://www.rfc-editor.org/info/rfc9162>.

[VCP-SPEC] VeritasChain Standards Organization, "VeritasChain Protocol (VCP) Specification Version 1.1", 2025, <https://veritaschain.org/spec>.

Acknowledgments

The author thanks the RATS Working Group for developing the comprehensive attestation architecture that enables discussions of complementary verification layers. This document builds upon and respects the careful design work reflected in the RATS architecture.

Author’s Address

Tokachi Kamimura
VeritasChain Standards Organization (VSO)
Email: kamimura@veritaschain.org
URI: https://veritaschain.org