Secure Evidence and Attestation Transport (SEAT) Working Group
Internet-Draft
Intended status: Informational
Expires: 23 April 2026

I. Mihalcea
Arm
M. U. Sardar
TU Dresden
T. Fossati
Linaro

20 October 2025

Use Cases and Properties for Integrating Remote Attestation with Secure Channel Protocols

draft-mihalcea-seat-use-cases-00

Abstract

This document outlines use cases and desirable properties for integrating remote attestation (RA) capabilities with secure channel establishment protocols, with an initial focus on Transport Layer Security (TLS) v1.3 Handshake. Traditional peer authentication in TLS establishes trust in a peer's network identifiers but provides no assurance regarding the integrity of its underlying software and hardware stack. Remote attestation addresses this gap by enabling a peer to provide verifiable evidence about its current state, including the state of its trusted computing base (TCB). This document explores specific use cases, such as confidential data collaboration and secure secrets provisioning, to motivate the need for this integration. From these use cases, it specifies a set of essential properties the protocol solution must have, including cryptographic binding to the TLS connection, evidence freshness, and flexibility to support different attestation models. This document is intended to serve as an input to the design of protocol solutions within the SEAT working group.

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 23 April 2026.

Table of Contents

1. Introduction

1.1. Establishing Trust in Secure Communications

Traditional secure channel protocols, such as Transport Layer Security (TLS), primarily establish trust in a peer's identity. This is typically achieved through mechanisms like a Public Key Infrastructure (PKI), where a trusted Certification Authority (CA) vouches for the binding between a public key and an identifier (e.g., a hostname).

However, this model has a core limitation: identity authentication provides no assurance about the peer's internal state or the integrity of its software stack. A compromised server, for instance, can still present a valid X.509 certificate and be considered "trusted" by a client. This gap allows compromised endpoints to maintain network access and the trust of their peers, posing a significant security risk in many environments.

1.2. The Role of Remote Attestation

Remote Attestation (RA), as described in the RATS architecture [RFC9334], is a mechanism designed to fill this gap. RA allows an entity (the "Attester") to produce verifiable "Evidence" about its current runtime state. This Evidence covers the Attester's TCB, and can thus include measurements of its firmware, operating system, and application code, as well as the configuration of its hardware and software security features (e.g., secure boot status, memory isolation). A "Relying Party" can then use this Evidence, often with the help of a trusted "Verifier", to appraise the Attester's trustworthiness.

By integrating RA into a secure channel establishment protocol, a second dimension of trust—trustworthiness—is added to complement regular peer authentication. This allows a peer to make authorization decisions based not just on who the other party is, but also on what it is (e.g., an AMD SEV-SNP-based server running in some known datacenter) and whether its state is acceptable.

1.3. Purpose and Scope

The purpose of this document is to outline the key use cases that motivate the integration of RA with secure channel protocols and to establish a set of essential properties for such an integration. The initial focus is on TLS 1.3 and its datagram-oriented variant, DTLS 1.3.

This document is intended as an input to the design of protocol solutions within the SEAT working group. It defines the "why" and the "what" (the requirements), but not the "how" (the protocol

specification itself). A key goal is to define requirements for a solution that is agnostic to any specific attestation technology (e.g., Trusted Platform Modules (TPMs), Intel TDX, AMD SEV, Arm CCA).

2. Terminology

This document uses the terminology defined in the RATS Architecture [RFC9334], including "Attester", "Relying Party", "Verifier", "Evidence", and "Attestation Results".

This document also uses the following terms:

* Trusted Computing Base (TCB) of a device: all security-relevant components: hardware, firmware, software, and their respective configurations.

* Confidential Workload: as defined in [I-D.draft-ccc-wimse-twi-extensions].

* Measurements: as defined in [I-D.draft-ietf-rats-eat-measured-component].

3. Use Cases

This section provides the concrete motivation for the WG’s work by describing specific use cases. For each case, the scenario, actors, and specific security guarantees needed from RA are described.

3.1. Confidential Data Collaboration

Goal: Enable multiple parties to collaborate on sensitive, combined datasets without exposing raw data to each other or to the infrastructure operator.

Use case: Data Clean Rooms: Multiple data providers contribute sensitive data to a confidential workload for joint analysis. Data consumers receive aggregated insights without ever accessing the raw, combined dataset.

* Requirement: Before sending data, each data provider must attest the confidential workload to verify it is running the authorized analysis code in a secure Trusted Execution Environment (TEE). Similarly, data consumers must attest the workload to trust the integrity of the results.

Use case: Secure Multi-Party Computation (MPC): Distributed parties collaboratively compute a function (e.g., train a machine learning model) without sharing their local data.

* Requirement: The central aggregator, as well as each participating client, must be able to mutually attest to ensure all parties are running the correct, untampered MPC algorithm in a trusted environment.

3.2. Secure Provisioning and High-Assurance Operations

Goal: Ensure the integrity of workloads and devices when bootstrapping their identity or receiving critical commands.

Use case: Runtime Secret Provisioning: A confidential workload starts in a generic state and needs to fetch secrets (e.g., API keys, database credentials, encryption keys) to become operational.

* Requirement: The workload must attest its runtime state (TEE genuineness, software measurements) to a secrets management service. The service will only release the secrets after successful verification, ensuring they are delivered exclusively to a trustworthy environment. This use-case also covers secure device onboarding for IoT devices that lack a pre-provisioned identity.

Use case: High-Assurance Command Execution: An operator sends a critical command to a remote system (e.g., an industrial controller, a financial transaction processor).

* Requirement: The system must provide fresh attestation Evidence to the operator to prove its integrity before the command is dispatched. This prevents commands from being executed on a compromised system.

3.3. Network Infrastructure Integrity

Goal: Verify the integrity of network devices that form the foundation of communication.

Use case: Attestation of Network Functions: A router, switch, or firewall joins a network’s management plane. A Virtualized Network Function (VNF) is instantiated on a generic server.

* Requirement: The network orchestrator must verify the device’s integrity (e.g., secure boot enabled, running signed OS and firmware) before allowing it to join the network and receive policy. This prevents a compromised router from misdirecting traffic or a malicious VNF from inspecting sensitive packets.

Use case: Securing Control and Management Planes: An administrator connects to a network device’s management interface.

* Requirement: The administrator’s client must verify the integrity of the management endpoint on the network device to ensure they are not connecting to a compromised interface that could steal credentials or manipulate the device.

4. Integration Properties

This section provides a list of desirable properties for designs that integrate RA into secure channel protocols. Proposed integration protocols should make it clear which of these properties are fulfilled, and how.

4.1. Cryptographic Binding to Communication Channel

The attestation Evidence or Attestation Result is cryptographically bound to the specific secure channel instance (e.g., the TLS connection). This prevents replay and relay attacks where an attacker presents valid, but old or unrelated Evidence from a different connection or context. This binding is paramount for all use cases.

4.2. Cryptographic Binding to Machine Identifier

Evidence should be cryptographically bound to the identifier provided to the machine by the infrastructure provider to prevent diversion attacks [Meeting-122-TLS-Slides].

4.3. Attestation Credential Freshness

The Relying Party is able to verify that the Evidence or Attestation Result it receives was freshly generated by the Attester for the current connection. State is transient, and credentials from a previous connection may no longer be valid. See Section 10 of [RFC9334] for more details about freshness in the context of RA.

4.4. Negotiation and Capability Discovery

Peers have a secure mechanism to discover each other’s support for RA, the specific attestation formats they can produce or consume, and the attestation models they support. This enables interoperability and allows for graceful fallback for endpoints that do not support RA.

4.5. Attestation Model Flexibility

The solution supports both the Background Check and Passport models as defined in the RATS architecture [RFC9334]. The Background Check model is essential for use cases requiring maximum freshness, while the Passport model is better suited for performance, scalability, and scenarios where the Verifier may be offline or unreachable by the Relying Party.

4.6. Interaction with Peer Authentication

The solution supports using RA in conjunction with traditional PKI-based authentication (e.g., X.509 certificates). This provides two independent pillars of trust: trustworthiness (from RA) and identity (from PKI). The solution may also support RA as the sole method of authentication in constrained use cases, such as device onboarding, where a device has no stable, long-term identity yet. This latter option could have a negative impact on the security of the overall design, warranting additional security considerations.

4.7. Credential Lifecycle Management

The solution can provide a mechanism for re-attesting or refreshing attestation credentials on an existing, long-lived connection without requiring a full new handshake. For long-lived connections, the initial attestation may become stale, and a lightweight refresh mechanism is beneficial towards re-evaluating the peer’s state.

4.8. Privacy Preservation

The solution does not degrade the privacy of a standard TLS connection. Evidence can contain highly specific, unique information about a device’s hardware and software, which could be used as an advanced tracking mechanism, following a user across different connections and services. The design must consider how to minimize this leakage, especially when a third-party Verifier is involved in the protocol exchange.

4.9. Performance and Efficiency

The introduction of attestation should not add prohibitive latency or overhead to the connection establishment process. To be widely adopted, the solution must be practical. While some overhead is unavoidable, multiple additional round-trips or very large payloads in the initial handshake should be minimized.

5. Security Considerations

This document describes use cases and integration properties. The security of any protocol designed to fulfill these properties will depend on its specific mechanisms. However, any solution must address the following high-level considerations:

* Replay and Relay Protection: The requirements for cryptographic binding and freshness are critical. Failure to bind attestation credentials tightly to the current connection would allow an adversary to replay or relay old or stolen, yet valid credentials from a compromised system, completely undermining the security goals.

* Verifier Trust and Privacy: In the Background Check model, the Relying Party communicates with a Verifier. This reveals to the Verifier that the Relying Party is communicating with the Attester. Depending on the scenario, this could leak sensitive information about business relationships or user activity. Solutions should consider mechanisms to minimize the data revealed to the Verifier.

* Downgrade Attacks: The negotiation of attestation capabilities must be secure. An active attacker must not be able to trick two parties that both support attestation into negotiating a connection without it.

* Evidence Semantics: This document does not define attestation appraisal policies. However, a Relying Party must be careful when interpreting Attestation Results. A "valid" attestation only means the Evidence is authentic and correctly signed; it does not automatically mean the underlying system is "secure". The Relying Party must have a clear policy for what measurements, software versions, and security configurations are acceptable.

6. IANA Considerations

This document has no IANA actions.

7. Informative References

[I-D.draft-ccc-wimse-twi-extensions]
Novak, M., Deshpande, Y., and H. Birkholz, "WIMSE Extensions for Trustworthy Workload Identity", Work in Progress, Internet-Draft, draft-ccc-wimse-twi-extensions-00, 4 July 2025, <https://datatracker.ietf.org/doc/html/draft-ccc-wimse-twi-extensions-00>.

[I-D.draft-ietf-rats-eat-measured-component]
Frost, S., Fossati, T., Tschofenig, H., and H. Birkholz, "EAT Measured Component", Work in Progress, Internet-Draft, draft-ietf-rats-eat-measured-component-05, 16 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-eat-measured-component-05>.

[Meeting-122-TLS-Slides]
Sardar, M. U., Moustafa, M., and T. Aura, "Identity Crisis in Attested TLS for Confidential Computing", March 2025, <https://datatracker.ietf.org/meeting/122/materials/slides-122-tls-identity-crisis-00>.

[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/rfc/rfc9334>.

Acknowledgments

TODO

Authors' Addresses

Ionu Mihalcea
Arm
Email: ionut.mihalcea@arm.com

Muhammad Usama Sardar
TU Dresden
Email: muhammad_usama.sardar@tu-dresden.de

Thomas Fossati
Linaro
Email: thomas.fossati@linaro.org