Remote Attestation Procedures Architecture
draft-birkholz-rats-architecture-02

Abstract

The Remote ATtestation proceduresS (RATS) architecture facilitates interoperability of attestation mechanisms by defining a set of participant roles and interactions that reveal information about the trustworthiness attributes of an attester’s computing environment. By making trustworthiness attributes explicit, they can be evaluated dynamically and within an operational context where risk mitigation depends on having a more complete understanding of the possible vulnerabilities germane to the attester’s environment.

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 March 15, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

The long-standing Internet Threat Model [RFC3552] focuses on threats to the communication channel, as pioneered by Dolev and Yao [DOLEV-YAO] in 1983. However, threats to the endpoint [RFC5209] and system components [RFC4949] of transited communication gear (i.e. hosts) are increasingly relevant for assessing the trustworthiness properties of a communication channel. Beyond the collection and conveyance of security posture [RFC5209] about an endpoint (host), remote attestation provides believable trustworthiness claims ("Evidence") about an endpoint (host). In general, this document provides normative guidance how to use, create or adopt network protocols that facilitate RATS.
1.1. RATS in a Nutshell

The RATS architecture provides a basis to assess the trustworthiness of endpoints by other parties:

- In remote attestation workflows, trustworthiness Claims are accompanied by a proof of veracity. Typically, this proof is a cryptographic expression such as a digital signature or message digest. Trustworthiness Claims with proof is what makes attestation Evidence believable.

- A corresponding attestation provisioning workflow uses trustworthiness Claims to convey believable Endorsements and Known-Good-Values used by a Verifier to appraise Evidence.

In the RATS architecture, specific content items are identified (and described in more detail below):

- Evidence is provable Claims about a specific Computing Environment made by an Attester.

- Known-Good-Values are reference Claims used to appraise Evidence.

- Endorsements are reference Claims about the environment protecting the Attesters capabilities to create believable Evidence (e.g. the type of protection for an attestation key). It answers the question "why Evidence is believable".

- Attestation Results are the output from the appraisal of Evidence, Known-Good-Values and Endorsements.

Attestation Results are the output of RATS. Assessment of Attestation Results can be multi-faceted, but is out-of-scope for the RATS architecture. If appropriate Endorsements about the Attester are available, Known-Good-Values about the Attester are available, and if the Attester is capable of creating believable Evidence - then the Verifier is able to create Attestation Results that enable Relying Parties to establish a level of confidence in the trustworthiness of the Attester.

2. Terminology

Conveyance: a mechanism for transferring RATS Evidence, Endorsements, Known-Good-Values or Attestation Results.

Entity: a user, organization, device or computing environment.
Principal: an Entity that implements RATS Roles and creates provable Claims or Attestation Results (see [ABLP] and [Lampson2007]).

Trustworthiness: an expectation about a computing environment that it will behave in a way that is intended and nothing more.

Computing Environment: a computing context consisting of system components.


2.1. Requirements Notation

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.

3. Conceptual Overview

In network protocol exchanges, it is often the case that one entity (a Relying Party) requires an assessment of the trustworthiness of a remote entity (an Attester or specific system components [RFC4949] thereof). Remote ATtestation procedureS (RATS) enable Relying Parties to establish a level of confidence in the trustworthiness of remote system components through the creation of attestation evidence by remote system components and a processing chain of architectural constituents towards the relying party.

The corresponding trustworthiness attributes processed may not be just a finite set of values. Additionally, the system characteristics of remote components themselves have an impact on the veracity of trustworthiness attributes included in Evidence. Attester environments can vary widely ranging from those highly resistant to attacks to those having little or no resistance to attacks. Configuration options, if set poorly, can result in a highly resistant environment being operationally less resistant. Computing Environments are often malleable being constructed from re-programmable hardware, firmware, software and updatable memory. When a trustworthy environment changes, the question has to be asked whether the change transitioned the environment from a trustworthy state to an untrustworthy state. The RATS architecture provides a framework for anticipating when a relevant change with respect to a
trustworthiness attribute occurs, what changed and how relevant it is. A remote attestation framework also creates a context for enabling an appropriate response by applications, system software and protocol endpoints when changes to trustworthiness attributes do occur.

3.1. Computing Environments

In the RATS context, a Claim is a specific trustworthiness attribute that pertains to a particular Computing Environment of an Attester. The set of possible Claims is expected to follow the possible computing environments that support attestation. In other words, identical (i.e. same type, model, versions, components and composition) Attesting Computing Environments can create different Claim values that still compose valid Evidence due to different computing contexts. Exemplary Claims include flight vectors or learned configuration.

Likely, there are a set of Claims that is widely applicable across most, if not all environments. Conversely, there are Claims that are unique to specific environments. Consequently, the RATS architecture incorporates extensible mechanisms for representing Claims.

Computing Environments can be complex structurally. In general, every Attester consists of multiple components (e.g. memory, CPU, storage, networking, firmware, software). Components are computational elements that can be linked and composed to form computational pipelines, arrays and networks (e.g. a BIOS, a bootloader, or a trusted execution environment).

An Attester includes at least one Computing Environment that is able to create attestation Evidence (the Attesting Computing Environment) about other Computing Environments (the Attested Computing Environments). Not every computational element of an Attester is expected to be a Computing Environment capable of remote attestation. Analogously, remote attestation capable Computing Environments may not be capable of attesting to (creating evidence about) every computational element that interacts with the Computing Environment. A Computing Environment with an attestation capability can only be endorsed by an external entity and cannot create believable evidence about itself by its own.

A Computing Environment with the capability of remote attestation:

- is separate from other Attested Computing Environments (about which attestation evidence is created), and
- is enabling the role of an Attester in the RATS architecture.
A Computing Environment with the capability of remote attestation and taking on the role of an Attester has the following duties in order to create Evidence:

- monitoring trustworthiness attributes of other Computing Environments,
- collecting trustworthiness attributes and create Claims about them,
- serialize Claims using interoperable representations,
- provide integrity protection for the sets of Claims, and
- add appropriate attestation provenance attributes about the sets of Claims.

### 3.2. Trustworthiness

The trustworthiness of remote attestation capabilities is also a consideration for the RATS architecture. It should be possible to understand the trustworthiness properties of the remote attestation capability for any set of claims of a remote attestation flow via verification operations. The RATS architecture anticipates recursive trustworthiness properties and the need for termination. Ultimately, a portion of a computing environment’s trustworthiness is established via non-automated means. For example, design reviews, manufacturing process audits and physical security. For this reason, trustworthy RATS depend on trustworthy manufacturing and supply chain practices.

### 3.3. RATS Workflow

The basic function of RATS is creation, conveyance and appraisal of attestation Evidence. An Attester creates attestation Evidence that are conveyed to a Verifier for appraisal. The appraisals compare Evidence with expected Known-Good-Values called obtained from Asserter (e.g. Principals that are Supply Chain Entities). There can be multiple forms of appraisal (e.g., software integrity verification, device composition and configuration verification, device identity and provenance verification). Attestation Results are the output of appraisals. Attestation Results are signed and conveyed to Relying Parties. Attestation Results provide the basis by which the Relying Party may determine a level of confidence to place in the application data or operations that follow.

RATS architecture defines attestation Roles (i.e., Attester, Verifier, Asserter and Relying Party), the messages they exchange, their structure and the various legal ways in which Roles may be
hosted, combined and divided (see Principals below). RATS messages are defined by an information model that defines Claims, environment and protocol semantics. Information Model representations are realized as data structure and conveyance protocol binding specifications.

3.4. Interoperability between RATS

The RATS architecture anticipates use of information modeling techniques that describe computing environment structures - their components/computational elements and corresponding capabilities - so that verification operations may rely on the information model as an interoperable way to navigate the structural complexity.

4. RATS Architecture

4.1. Goals

RATS architecture has the following goals:

- Enable semantic interoperability of attestation semantics through information models about computing environments and trustworthiness.

- Enable data structure interoperability related to claims, endpoint composition / structure, and end-to-end integrity and confidentiality protection mechanisms.

- Enable programmatic assessment of trustworthiness. (Note: Mechanisms that manage risk, justify a level of confidence, or determine a consequence of an attestation result are out of scope).

- Provide the building blocks, including Roles and Principals that enable the composition of service-chains/hierarchies and workflows that can create and appraise evidence about the trustworthiness of devices and services.

- Use-case driven architecture and design (RATS use cases are summarized in [I-D.richardson-rats-usecases]).

- Terminology conventions that are consistently applied across RATS specifications.

- Reinforce trusted computing principles that include attestation.
4.2. Attestation Principles

Specifications developed by the RATS working group apply the following principles:

- **Freshness** - replay of previously asserted Claims about an Attested Computing Environment can be detected.
- **Identity** - the Attesting Computing Environment is identifiable (non-anonymous).
- **Context** - the Attested Computing Environment is well-defined (unambiguous).
- **Provenance** - the origin of Claims with respect to the Attested and Attesting Computing Environments are known.
- **Validity** - the expected lifetime of Claims about an Attested Computing Environment is known.
- **Relevance** - the Claims associated with the Attested Computing Environment pertain to trustworthiness metrics.
- **Veracity** - the believability (level of confidence) of Claims is based on verifiable proofs.

4.3. RATS Roles and Messages

The RATS Roles (roles) are performed by RATS Principals.

The RATS Architecture provides the building blocks to compose various RATS roles by leveraging existing and new protocols. It defines architecture for composing RATS roles with principals and models their interactions.

Figure 1 provides an overview of the relationships between RATS Roles and the messages they exchange.
4.3.1. Roles

RATS roles are implemented by principals that possess cryptographic keys used to protect and authenticate Claims or Results.

Attester: An Attestation Function that creates Evidence by collecting, formatting and protecting (e.g., signing) Claims. It presents Evidence to a Verifier using a conveyance mechanism or protocol.

Verifier: An Attestation Function that accepts Evidence from an Attester using a conveyance mechanism or protocol. It also accepts Known-Good-Values and Endorsments from an Asserter using a conveyance mechanism or protocol. It verifies the protection mechanisms, parses and appraises Evidence according to good-known valid (or known-invalid) Claims and Endorsments. It produces Attestation Results that are formatted and protected (e.g., signed). It presents Attestation Results to a Relying Party using a conveyance mechanism or protocol.

Asserter: An Attestation Function that generates reference Claims about both the Attesting Computing Environment and the Attested Computing Environment. The manufacturing and development processes are presumed to be trustworthy processes. In other words the Asserter is presumed, by a Verifier, to produce valid Claims. The function collects, formats and protects (e.g. signs)
valid Claims known as Endorsements and Known-Good-Values. It presents provable Claims to a Verifier using a conveyance mechanism or protocol.

Relying Party: An Attestation Function that accepts Attestation Results from a Verifier using a conveyance mechanism or protocol. It assesses Attestation Results protections, parses and assesses Attestation Results according to an assessent context (Note: definition of the assessment context is out-of-scope).

4.3.2. Role Messages

Claims: Statements about trustworthiness characteristics of an Attested Computing Environment.

The veracity of a Claim is determined by the reputation of the entity making the Claim. (Note: Reputation may involve identifying, authenticating and tracking transactions associated with an entity. RATS may be used to establish entity reputation, but not exclusively. Other reputation mechanisms are out-of-scope).

Evidence: Claims that are formatted and protected by an Attester.

Evidence SHOULD satisfy Verifier expectations for freshness, identity, context, provenance, validity, relevance and veracity.

Known-Good-Values: Claims about the Attested Computing Environment. Typically, KGV Claims are message digests of firmware, software or configuration data supplied by various vendors. If an Attesting Computing Environment implements cryptography, they include Claims about key material.

Like Claims, Known-Good-Values SHOULD satisfy a Verifier’s expectations for freshness, identity, context, provenance, validity, relevance and veracity. Known-Good-Values are reference Claims that are - like Evidence - well formatted and protected (e.g. signed).

Endorsements: Claims about immutable and implicit characteristics of the Attesting Computing Environment. Typically, endorsement Claims are created by manufacturing or supply chain entities.

Endorsements are intended to increase the level of confidence with respect to Evidence created by an Attester.

Attestation Results: Statements about the output of an appraisal of Evidence that are created, formatted and protected by a Verifier.
Attestation Results provide the basis for a Relying Party to establish a level of confidence in the trustworthiness of an Attester. Attestation Results SHOULD satisfy Relying Party expectations for freshness, identity, context, provenance, validity, relevance and veracity.

4.4. RATS Principals

RATS Principals are entities, users, organizations, devices and computing environments (e.g., devices, platforms, services, peripherals).

RATS Principals may implement one or more RATS Roles. Role interactions occurring within the same RATS Principal are out-of-scope.

The methods whereby RATS Principals may be identified, discovered, authenticated, connected and trusted, though important, are out-of-scope.

Principal operations that apply resiliency, scaling, load balancing or replication are generally believed to be out-of-scope.

```
+------------------+   +------------------+
|  Principal 1     |   |  Principal 2     |
|  +------------+  |   |  +------------+  |
|  |            |  |   |  |            |  |
|  |    Role 1  |<-|---|->|    Role 2  |  |
|  |            |  |   |  |            |  |
|  +------------+  |   |  +------------+  |
|                  |   |                  |
|  +-----+------+  |   |  +-----+------+  |
|  |            |  |   |  |            |  |
|  |    Role 2  |<-|---|->|    Role 3  |  |
|  |            |  |   |  |            |  |
|  +------------+  |   |  +------------+  |
|                  |   |                  |
+------------------+   +------------------+
```

Figure 2: RATS Principals-Role Composition

RATS Principals have the following properties:

- Multiplicity - Multiple instances of RATS Principals that possess the same RATS Roles can exist.

- Composition - RATS Principals possessing different RATS Roles can be combined into a singleton RATS Principal possessing the union
of RATS Roles. RATS Interactions between combined RATS Principals is uninteresting.

- Decomposition - A singleton RATS Principal possessing multiple RATS Roles can be divided into multiple RATS Principals.

RATS Interactions may occur between them.

5. Security Considerations

RATS Evidence, Verifiable Assertions and Results SHOULD use formats that support end-to-end integrity protection and MAY support end-to-end confidentiality protection. Replay attack prevention MAY be supported if a Nonce Claim is included. Nonce Claims often piggy-back other information and can convey attestation semantics that are of essence to RATS, e.g. the last four bytes of a challenge nonce could be replaced by the IPv4 address-value of the Attester in its response.

All other attacks involving RATS structures are not explicitly addressed by RATS architecture. Additional security protections MAY be required of conveyance mechanisms. For example, additional means of authentication, confidentiality, integrity, replay, denial of service and privacy protection of RATS payloads and Principals may be needed.

6. References

6.1. Normative References


6.2. Informative References


Authors' Addresses

Henk Birkholz
Fraunhofer SIT
Rheinstrasse 75
Darmstadt  64295
Germany
Email: henk.birkholz@sit.fraunhofer.de

Monty Wiseman
GE Global Research
USA
Email: monty.wiseman@ge.com
Hannes Tschofenig
ARM Ltd.
110 Fulbourn Rd
Cambridge  CB1 9NJ
UK

Email: hannes.tschofenig@gmx.net

Ned Smith
Intel Corporation
USA

Email: ned.smith@intel.com