RATS Architecture & Terminology

Henk Birkholz {henk.birkholz@sit.fraunhofer.de}
Ned Smith {ned.smith@intel.com}

IETF 104, Prague, March 28\textsuperscript{th}, RATS WG
IETF 103: An Evolution of Boxes

• At the beginning there were boxes

  There was uncertainty about the meaning of the “lines” connecting the “boxes”

  There was uncertainty about the duties each “box” performs and whether or not a 3 box model was oversimplified / overly complex

• And there was a bit confusion

  Device
  Attestation Server
  Relying Party (RP)
Post IETF 103: Recap (from the rats list) & Evolution

- **RATS Actors**: an architectural container that captures different deployment options
  - Examples:
    - **Device**, TEE, peripheral, co-processor, etc.
    - **Resource manager**, device, directory service, server, sensor, router, gateway, etc.
    - **Supply chain entity**, ODM, OEM, OSV, IHV, etc.
    - **Attestation service**, broker, orchestrator, device, etc.

- **RATS Roles**: provide a more consistent architectural structure:
  - Attester, Relying Party, Asserter & Verifier

- **RATS Interactions**: an architectural description of data in motion specifying the content required to be conveyed

- All three concepts combined enable flexible “Composability” to address different use-cases.
RATS Architecture Principles

• Information Model
  • Abstract representation of evidence, interactions and endpoints

• Data Model
  • Interoperable representation of evidence and interactions
  • Endpoint identity and definition is out-of-scope (but relevant)

• Deployment Flexibility
  • RATS solutions follow / integrate with RATS attestation use cases
  • RATS solutions integrate with IETF and other conveyance protocols
  • RATS solutions integrate with existing and emerging public key infrastructures
Evolution of RATS Architecture: Actors

1. Supply Chain Entity (SCE)
   - (I1) Provision DeviceID & Bind AtAs to Device

2. Resource Manager
   - (I2) Publish Attestation Assertions (AtAs)
   - (I3) Attestation Evidence Conveyance (EC)
     - (1a) TEE, eSE, or similar
     - (2a) Storage and Distribution
     - (2b) Remote Attestation Service

3. Enforce Trustworthiness Policy
   - (I4) Attestation-Result Conveyance (RC)
   - (Note: Only Evidence containing verifiable AtAs are conveyed)
Evolution of RATS Architecture: Roles

**Asserter**
(Note: 1-n supply chain entities per Device are possible)

(I1) Provision DeviceID & Bind AtAs to Device

**Attester**

**Verifier**
(Note: 0-n RA-Services per SC(E) are possible)

(I2) Publish Attestation Assertions (AtAs)

(I3) Attestation Evidence Conveyance (EC)
(Note: Only Evidence containing verifiable AtAs are conveyed)

(I4) Attestation-Result Conveyance (RC)

**Relying Party**

Enforce Trustworthiness Policy

1. (I1) Provision DeviceID & Bind AtAs to Device
2. (I2) Publish Attestation Assertions (AtAs)
3. (I3) Attestation Evidence Conveyance (EC)
(Note: Evidence = AtAs to device binding using a RoT key)
4. (I4) Attestation-Result Conveyance (RC)

(Note: 1-n Attestor Supply Chain entities per Device are possible)
Composability of Roles on Actors

Actor: Supply Chain Entities
Roles: Asserter & Verifier
(Note: 1-n supply chain entities per Device & 0-n RA-Services per SC(E) are possible)

Actor: Device
Role: Attester

Actor: Resource Manager
Roles: Relying Party & Verifier

An alternative Composition:
Actor: Resource Manager
Roles: Relying Party & Verifier

(I1) Provision DeviceID & Bind AtAs to Device

(I2) Evaluate Trustworthiness

(I3) Evidence Conveyance (EC)

(I3) Evidence Conveyance

(I4) Attestation-Result Conveyance (RC)
RATS WG Scoping

Out of Scope

(I1) Provision DeviceID & Bind AtAs to Device

(I2) Publish Attestation Assertions (AtAs)

(I3) Attestation Evidence Conveyance (EC)
(Note: Only Evidence containing verifiable AtAs are conveyed)

(I4) Attestation-Result Conveyance (RC)

Attester
(Note: 1-n supply chain entities per Device are possible)

Verifier
(Note: 0-n RA-Services per SC(E) are possible)

Relying Party

Enforce Trustworthiness Policy

Attestation Evidence Conveyance (EC)
(Note: Evidence = AtAs to device binding using a RoT key)

Storage and Distribution

1a

2a

2b

1b

2

1

3

Note: Actor-Role compositions can create ambiguous WG scope scenarios
Relationships to Corresponding Architectures

• **TEEP Architecture Components**
  • Trusted Application Managers (TAM) - Actor taking on the role of **Relying Party** AND/OR **Verifier**
  • Device /w TEE - Actor taking on the role of **Attester**

• **Platform Security Architecture (PSA) Components**
  • Network and App Services - Actor taking on the role of **Relying Party** AND/OR **Verifier**
  • Hardware - Actor taking on the role of **Attester**

• **EAT Overall System Components**
  • Relying Party - Actor maps to **Relying Party** role
  • Entity - Actor taking on the role of the **Attester**
  • Entity Manufacturer - Actor taking on the role of **Asserter** AND/OR **Verifier**
Overlap with other Working Groups

- **TEEP WG**
  - Trusted Execution Environments (TEE) in **Devices**
  - **Manifest Profiles**
  - TEE **Attestation Provenance** procedures

- **SUIT WG**
  - **Manifest Format & Information Model** (approach)

- **SACM WG**
  - Identity **Manifest & Information Model** (CoSWID)

- **NETCONF WG**
  - Managed **Trust Anchor** Repository (data at rest)

- **TAMP WG**
  - Protocol for configuring **Trust Anchor** policies (data in motion)
Overlapping Terminology

- RFC 4949 defines common security terminology
- Mapping of terms between different WG work efforts
  - SACM: security automation terminology
  - TEEP: attestation & trusted computing terminology
  - SUIT: evidence & measurement terminology
  - NETCONF: trust anchor terminology
- NIST, Global Platform, FIDO, and TCG defines attestation terminology.
- RATS Architecture needs to build consensus on a core vocabulary.
A suitable level of abstraction combined with thorough guidance that enables one to create interoperable solutions from it.

- E.g. the RATS Architecture avoids the term “claim” as that term is “claimed” by CWT and might create a bias towards a specific scope of solutions. The generic term used instead is “assertion”.
  - Assertions are represented as claims in CWT.
  - Assertions might be represented differently in other representation.

- The intent of the current Actor/Role/Duty/Interaction concepts that compose the RATS Architecture is to take into account, align, and consolidate current IETF WG work (& work of different SDO).
Vital Elements of RATS (next steps)

• Vital Elements of the RATS enabled by the architecture document are:
  • Attestation Assertion (AtAs) and
  • Attestation Semantics (AtSe)

• The common denominator is a compact set of (occasionally semantical grouped) assertions about the Computing Context to be attested/conveyed.

• Asserters (mostly called Claimants at this point of time) provide these assertion (data origin), but they are not necessarily the initial point where they are acquired (data source).

• Proposal: a basic set of assertions for RATS is required (e.g. via an Information Model)
  • Please take into account the lessons learned in the SACM WG

• The initial set of information elements is about “Remote Attestation” and not “Attestation Provisioning” (which is out-of-scope for now).
Reference Interaction Model for Challenge-Response-based Remote Attestation Procedures

Henk Birkholz {henk.birkholz@sit.fraunhofer.de}
Michael Eckel {michael.eckel@sit.fraunhofer.de}

IETF 104, Prague, March 28th, RATS WG
Why is this a Useful Normative Document?

• **Background**
  • Most **protocols** that require a **proof-of-freshness** use a **Challenge/Response**-based interaction
  • A **Nonce** that is provided by the challenger, processed cryptographically by the receiver and then returned to the challenger in a way that proofs that the response is a freshly composed set of information.

• **Usage**
  • This procedure is done at many places and in many protocols already 👍
  • This procedure is mostly “re-”explained and illustrated over and over again 👎

• **Contribution**
  • By describing and illustrating this essential concept in an elaborate and use-case agnostic fashion will prevent “cloning” this normative text over and over again.
The State of the Document

• Invaluable side-effect: visibility & review
  • Everyone, who is interested, can potentially find a small detail that might be missing, or wrong, or could be forked into multiple alternatives on how to do it.

• Current work
  • There are two complete (and rather thorough) sets of reviews that did not make it into the current I-D still. Stay tuned!
  • We hope for even more visibility and feedback after IETF 104.

• Current application
  • The first I-D to off-load this content is: I-D. birkholz-yang-basic-remote-attestation

• Early feedback: this seems to work pretty well, already. Please bash, if you think otherwise! Alternatively, please add the details you may find missing.
YANG Module for Basic Challenge-Response-based Remote Attestation Procedures

Henk Birkholz {henk.birkholz@sit.fraunhofer.de}
Michael Eckel {michael.eckel@sit.fraunhofer.de}
Shwetha Bhandari {shwethab@cisco.com}
Bill Sulzen {bsulzen@cisco.com}
Eric Voit {evoit@cisco.com}
Guy C. Fedorkow {gfedorkow@juniper.de}

IETF 104, Prague, March 28th, RATS WG
The Contribution of this Document

• Background
  • YANG defines a language to define data repositories for data at rest and it defines a set of operations to operate on these YANG datastores.
  • Additionally, there are ways to create RPCs, to subscribe to “hardcoded” notifications, or to changes (to parts of a) YANG datastore, i.e. creating continuous telemetry.
  • Curious? NETCONF (& NETMOD) is the place to go exploring 😊

• Usage
  • YANG is widely used and deployed, especially on network equipment and virtual services.
  • Adding Remote Attestation as procedures to existing and implemented management interfaces significantly reduces the threshold of adoption (another good example: tokbind)

• Contribution
  • This YANG module provides an RPC implementing the Reference Interaction Model for Challenge/Response based RATS (i.e. “nonce-based”).
  • The YANG module also supports multiple Roots-of-Trust for Reporting in a composite device to create remote attestation evidence about integrity and therefore trustfulness of network equipment (or VNF, respectively). I.e. enabling trustworthy continuous telemetry.
The State of the Document

• Current Work
  • The current version of the YANG module is already quite mature.
  • It defines an RFC for the Challenge/Response Procedure and a datastore for complementary information elements, such as Identity Documents, Endorsement Documents, or Device Composition – but maybe more is needed?

• The YANG statements in the I-D might require more textual description in another section (the description statement already helps, but is not enough to convey the bigger picture – probably).
Time-Based Uni-Directional Attestation (TUDA)

Henk Birkholz {henk.birkholz@sit.fraunhofer.de}
Andreas Fuchs {andreas.fuchs@sit.fraunhofer.de}
Ira McDonald {blueroofmusic@gmail.com}
Carsten Bormann {cabo@tzi.de}

IETF 104, Prague, March 28\textsuperscript{th}, RATS WG
The Contribution of this Document

• The Reference Interaction Model presented before utilizes a nonce-based procedure to provide a proof for freshness
  • The hand-shake involved in bi-directional protocols

• TUDA uses an **external** trusted time source:
  • an RFC 3161 Time Stamping Authority (TSA) that is
  • creating trusted Time Stamp Tokens (TST).

• As a result, TUDA allows for **uni-directional** unicast, broadcast, or multicast of attestation evidence – requiring **no response from the Verifier**.
  • TUDA creates secure and **trustworthy Audit Logs** of past operational states.
TUDA Methodology (in a nutshell)

• A local source of time creates a timestamp that is cryptographically bound to a timestamp created by a trusted system global source of time (the TSA).

• The result again is cryptographically bound to a second timestamp of the local source of time.

• The resulting Sync-Proof provides evidence in which period of time the association (cryptographically binding) with the trusted system global source of time (TSA) must have happened.

• Consequently, evidence signed via a Root-of-Trust of Reporting in this period of time must have been fresh [see RFC4949] and must compose provable operational state of the Attester at that given time.

• The output of this procedure are secure audit logs that constitute attestation evidence that can be conveyed and verified at any time in the future without a nonce-based proof of recentness.
The State of the Document

• All technical details, information elements and functions required by the Attester role are completed and mature (including running code).
• Structure and layout need improvement.
• A corresponding SNMP MIB & YANG module are included.
  • The YANG module is “simply” derived from the MIB and needs refactoring.
• A consolidated RATS terminology (and maybe a base set of RATS assertion/information elements) is still required for another update of this I-D.
• If there are appropriate use-cases defined, the use of CWT to convey the TUDA information elements could be taken into consideration.