Reference Interaction Model for Challenge-Response-based Remote Attestation
draft-birkholz-rats-reference-interaction-model-00

Abstract

   This document defines an interaction model for a basic remote
   attestation procedure. Additionally, the required information
   elements are illustrated.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   Remote attestation procedures (RATS) are a combination of activities,
   in which a Verifier creates assertions about claims of integrity and
   about the characteristics of other system entities by the appraisal
   of corresponding signed claims (evidence). In this document, a
   reference interaction model for a generic challenge-response-based
   remote attestation procedure is provided. The minimum set of
   components, roles and information elements that have to be conveyed
   between Verifier and Attester are defined as a standard reference to
   derive more complex RATS from.

1.1. 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 RFC
   2119, BCP 14 [RFC2119].

2. Disambiguation

   The term "Remote Attestation" is a common expression and often
   associated with certain properties. The term "Remote" in this
   context does not necessarily refer to a remote system entity in the
   scope of network topologies or the Internet. It rather refers to a

decoupled system or different computing context, which also could be
present locally as components of a composite device. Examples
include: a Trusted Execution Environment (TEE), Baseboard Management
Controllers (BMCs), as well as other physical or logical protected/
isolated execution environments.

3. Scope

This document focuses on a generic interaction model between
Verifiers and Attesters. Complementary processes, functions and
activities that are required for a complete semantic binding of RATS
are not in scope. Examples include: identity establishment, key
enrollment, and certificate revocation. Furthermore, any processes
and activities that go beyond carrying out the remote attestation
process are out of scope. For instance, using the result of a remote
attestation that is emitted by the Verifier, such as triggering
remediation actions and recovery processes, as well as the
remediation actions and recovery processes themselves, are out of
scope.

4. Component Roles

The Reference Interaction Model for Challenge-Response-based Remote
Attestation is based on the standard roles defined in
[I-D.birkholz-rats-architecture]:

Attester: The role that designates the subject of the remote
   attestation. A system entity that is the provider of evidence
   takes on the role of an Attester.

Verifier: The role that designates the system entity and that is the
   appraiser of evidence provided by the Attester. A system entity
   that is the consumer of evidence takes on the role of a Verifier.

5. Prerequisites

Identity: An Attester must have a unique Identity in such a way that
   a Verifier can uniquely identify an Attester. This Identity MUST
   be part of the signed claims (attestation evidence) that the
   Attester conveys to the Verifier.

Secret: A Secret that is present on the Attester and that a Verifier
   can identify by its Secret ID, e.g. a public key. This Secret
   MUST be established before a remote attestation procedure can take
   place. How this Secret is established is out of scope for this
   reference model.

6. Remote Attestation Interaction Model

This section defines the information elements that have to be conveyed via a protocol, enabling the conveyance of Evidence between Verifier and Attester, as well as the interaction model for a generic challenge-response scheme.

6.1. Information Elements

Nonce: mandatory

   The Nonce (number used once) is a number intended to be unique and intended to be effectively infeasible to guess. In this reference interaction model it MUST be provided by the Verifier and MUST be used as a proof of freshness, with respect to conveyed evidence ensuring that the result of an attestation activity was created recently (i. e. triggered by the challenge emitted by the Verifier). As such, the Nonce MUST be part of the signed evidence sent by the Attester to the Verifier.

Secret ID: mandatory

   An identifier that MUST be associated with the Secret which is used to sign the evidence.

Evidence: mandatory

   The signed claims that are required to enable integrity proving of the corresponding characteristics of the Attester. Examples of claims included in attestation evidence are claims about sensor data, policies that are active on the system entity, versions of composite firmware of a platform, running software, routing tables, or information about a local time source. Attestation evidence must be cryptographically bound to the Verifier-provided Nonce, the Identity of the Attester, as well as to the Secret identified by the Secret ID.

Claim Selection: optional

   An Attester MAY provide a selection of claims that are relevant to the appraisal conducted by the Verifier in order to prove correctness of the (integrity) claims created by the Attester. Usually, all available claims that are available to the Attester SHOULD be conveyed. This claim selection can be composed as complementary signed claims or can be encapsulated claims in the signed evidence that composes the evidence about integrity. This information element is optional in order to enable a Verifier to narrow down or increase the amount of received evidence. An
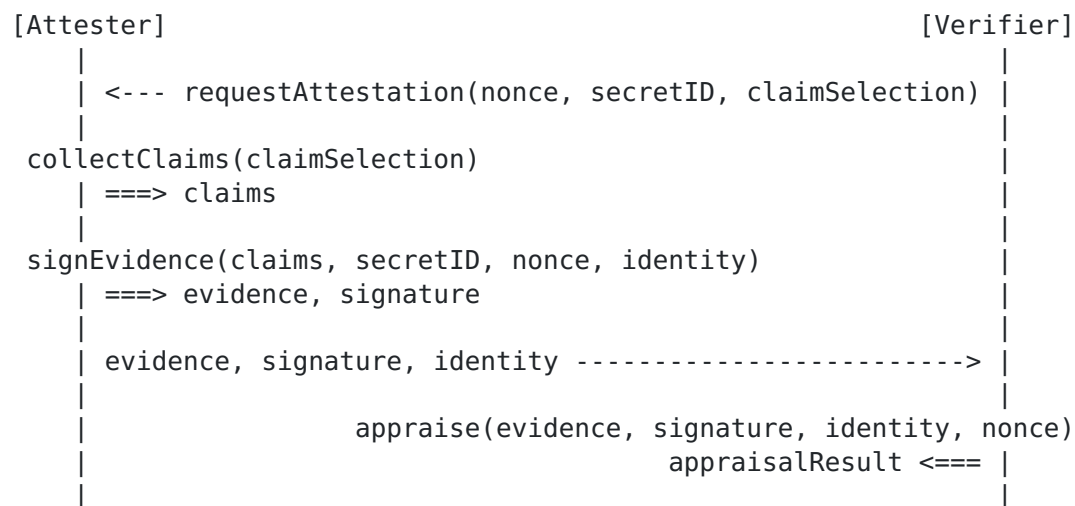
Attester MAY decide whether or not to provide the requested claims
or not. In either case, the claim selection MUST be
cryptographically bound to the signed claim set. An example for a
claim selection is that a Verifier can request from an Attester
(signed) Reference Integrity Measurements (RIMs), which represent
a claim about the intended platform operational state of the
Attester.

Identity: mandatory

A statement about a distinguishable Attester made by an entity
without accompanying evidence of its validity, used as proof of
identity.

7. Interaction Model

The following sequence diagram illustrates the reference remote
attestation procedure defined by this document.

```
[Attester]                                              [Verifier]
    |                                                        |
    | <--- requestAttestation(nonce, secretID, claimSelection) |
    |                                                        |
 collectClaims(claimSelection)                              |
    | ===> claims                                           |
    |                                                        |
 signEvidence(claims, secretID, nonce, identity)           |
    | ===> evidence, signature                              |
    |                                                        |
    | evidence, signature, identity ------------------------> |
    |                                                        |
    |                   appraise(evidence, signature, identity, nonce)
    |                                     appraisalResult <=== |
    |                                                        |
```

The remote attestation procedure is initiated by the Verifier,
sending an attestation request to the Attester. The attestation
request consists of a Nonce, a Secret ID, and a Claim Selection. The
Nonce guarantees attestation freshness. The Secret ID selects the
secret the Attester is requested to sign the Evidence with. The
Claim Selection narrows down or increases the amount of received
Evidence, if required. If the Claim Selection is empty, then by
default all claims that are available on the system of the Attester
SHOULD be signed and returned as Evidence. For example, the Verifier
is only interested in particular information about the Attester, such
as whether the device booted up in a known state, and not include
information about all currently running software.

The Attester, after receiving the attestation request, collects the
corresponding claims to compose the evidence the Verifier requested--
or, in case the Verifier did not provide a claim selection, the
Attester collects all information that can be used as complementary
claims in the scope of the semantics of the remote attestation
procedure. After that, the Attester signs the evidence with the
secret identified by the secret ID, including the nonce and the
identity information. Then the Attester sends the output back to the
Verifier. Important at this point is that the nonce as well as the
identity information must be cryptographically bound to the
signature, i. e. it is not required for them to be present in plain
text. For instance, those information can be part of the signature
after a one-way function (e. g. a hash function) was applied to them.
There is also a possibility to scramble the nonce or identity with
other information that is known to both the Verifier and Attester. A
prominent example is the IP address of the Attester that usually is
known by the Attester as well as the Verifier. This extra
information can be used to scramble the Nonce in order to counter
certain types of relay attacks. As soon as the Verifier receives the
evidence, it appraises it, including the verification of the
signature, the identity, the nonce, and the claims included in the
evidence. This process is application-specific and can be done by e.
g. comparing the claims to known (good), expected reference claims,
such as Reference Integrity Measurements (RIMs), or evaluating it in
other ways. The final output, the appraisal result (also referred to
as attestation result), is a new claim about properties of the
Attester, i. e. whether or not it is compliant to policies, or even
can be "trusted".

8. Further Context

Depending on the use cases to cover there may be additional
requirements.

8.1. Confidentiality

Use confidential communication to exchange attestation information.
This requirement usually is present when communication happens over
insecure channels, such as the public Internet. Speaking of a
suitable communication protocol, TLS is a good candidate. In private
networks, such as carrier management networks, it must be evaluated
whether or not the transport medium is considered confidential.

8.2. Mutual Authentication

In particular use cases mutual authentication may be desirable in
such a way that a Verifier also needs to prove its identity to the

Attester instead of only the Attester proving its identity to the Verifier.

8.3. Hardware-Enforcement/Support

In particular use cases hardware support can be desirable. Depending on the requirements those can be secure storage of cryptographic keys, crypto accelerators, or protected or isolated execution environments. Well-known technologies are Hardware Security Modules (HSM), Physical Unclonable Functions (PUFs), Shielded Secrets, Trusted Executions Environments (TEEs), etc.

9. Security Considerations

There are always some.

10. Acknowledgements

Very likely.

11. Change Log

Initial draft -00

Changes from version 00 to version 01:

Added details to the flow diagram

12. References

12.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
           Requirement Levels", BCP 14, RFC 2119,
           DOI 10.17487/RFC2119, March 1997,
           <https://www.rfc-editor.org/info/rfc2119>.

12.2. Informative References

[I-D.birkholz-rats-architecture]
           Birkholz, H., Wiseman, M., Tschofenig, H., and N. Smith,
           "Architecture and Reference Terminology for Remote
           Attestation Procedures", draft-birkholz-rats-
           architecture-00 (work in progress), October 2018.

Authors' Addresses

      Henk Birkholz
      Fraunhofer SIT
      Rheinstrasse 75
      Darmstadt 64295
      Germany

      Email: henk.birkholz@sit.fraunhofer.de

      Michael Eckel
      Huawei Technologies
      Feldbergstrasse 78
      Darmstadt 64293
      Germany

      Email: michael.eckel@huawei.com