RATS Working Group
Internet-Draft
Intended status: Informational
Expires: December 21, 2019

M. Richardson
Sandelman Software Works
June 19, 2019

Use cases for Remote Attestation common encodings
draft-richardson-rats-usecases-02

Abstract

This document details mechanisms created for performing Remote Attestation that have been used in a number of industries. The document initially focuses on existing industry verticals, mapping terminology used in those specifications to the more abstract terminology used by the IETF RATS Working Group.

The document aspires to describe possible future use cases that would be enabled by common formats.

Commented [DT1]: typo

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 December 21, 2019.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

The recently chartered IETF RATS WG intends to create a system of attestations that can be shared across a multitude of different users.

This document exists as a place to collect use cases for the common RATS technologies in support of the IETF RATS charter point 1. This document is not expected to be published as an RFC, but remains open as a working document. It could become an appendix to provide motivation for a protocol standards document.

This document will probably not deal with use cases from an end-user point of view, but rather on the technology verticals that wish to use RATS concepts (such as EAT) in their deployments.

End-user use cases that would either directly leverage RATS technology, or would serve to inform technology choices are welcome, however.

Commented [DT2]: Why not? If you can get text for one, why exclude it?
The next paragraph seems to indicate you would, in which case this paragraph is misleading. It implies that "you're welcome to send me text, but I'll probably ignore it", which is what I'm pushing back against.

2. Terminology

Critical to dealing with and constrasting different technologies is to collect terms with which are compatible, to distinguish those terms which are similar but used in different ways.

This section will grow to include forward and external references to terms which have been seen. When terms need to be disambiguated they will be prefixed with their source, such as "TCG(claim)" or "FIDO(relying party)"

Platform attestations generally come in two categories. This document will attempt to indicate for a particular attestation technology falls into this.

2.1. Static attestations

A static attestation says something about the platform on which the code is running.

2.2. Session attestations

A session attestation says something about how the shared session key was created.

2.3. Statements

The term "statement" is used as the generic term for the semantic content which is being attested to.

3. Requirements Language

This document is not a standards track document and does not make any normative protocol requirements using terminology described in [RFC2119].

4. Overview of Sources of Use Cases

The following specifications have been convered in this document:

- The Trusted Computing Group "Network Attestation System" (private document)
- Android Keystore
- Fast Identity Online (FIDO) Alliance attestation,

This document will be expanded to include summaries from:

Commented [DT3]: typo

Commented [DT4]: typo

Commented [DT5]: In what sense is this 'static'? It can be updated at any time, and so can a policy as to whether a platform is considered compliant or not.

Commented [DT6]: Undefined term.

- Trusted Computing Group (TCG) Trusted Platform Module (TPM)/Trusted Software Stack (TSS)

- ARM "Platform Security Architecture" [I-D.tschofenig-rats-psa-token]

And any additional sources suggested.

5. Use case summaries

5.1. Trusted Computing Group (TCG)

The TCG is trying to solve the problem of knowing if a networking device should be part of a network, if it belongs to the operator, and if it is running appropriate software.

This proposal is a work-in-progress, and is available to TCG members only. The goal is to be multi-vendor, scalable and extensible. The proposal intentionally limits itself to:

- "non-privacy-preserving applications (i.e., networking, Industrial IoT)",

- ~~that~~ the firmware is provided by the device manufacturer

- ~~that~~ there is a manufacturer installed hardware root of trust (such as a TPM and boot ~~room~~ROM)

Service providers and enterprises deploy hundreds of routers, many of them in remote locations where they're difficult to access or secure. The point of remote attestation is to:

- identify a remote box in a way that's hard to spoof

- report the inventory of software was launched on the box in a way that can not be spoofed

The use case described is to be able to monitor the authenticity of software versions and configurations running on each device. This allows owners and auditors to detect deviation from approved software and firmware versions and configurations, potentially identifying infected devices.

Attestation may be performed by network management systems. Networking Equipment is often highly interconnected, so it's also possible that attestation could be performed by neighboring devices.

Commented [DT7]: Intel SGX attestation: https://software.intel.com/en-us/sgx/attestation-services

Windows Defender System Guard attestation: https://www.microsoft.com/security/blog/2018/04/19/introducing-windows-defender-system-guard-runtime-attestation/

Windows Device Health Attestation: https://docs.microsoft.com/en-us/windows-server/security/device-health-attestation

Azure Sphere Attestation: https://azure.microsoft.com/en-us/resources/azure-sphere-device-authentication-and-attestation-service/en-us/

Also I believe the IETF NEA WG (https://datatracker.ietf.org/doc/charter-ietf-nea/) was relevant to the network attestation use case, and in particular I think referencing RFC 5209 would be appropriate.

Commented [DT8]: typo

Commented [DT9]: Add this term to the terminology section above

Commented [DT10]: I believe this is identical to the use case in RFC 5209.

Specifically listed to be out of scope includes: Linux processes, assemblies of hardware/software created by end-customers, and equipment that is sleepy (check term).

The TCG Attestation leverages the TPM to make a series of measurements during the boot process, and to have the TPM sign those measurements. The resulting "PCG" hashes are then available to an external verifier.

The TCG uses the following terminology:

- Device Manufacturter
- Attester ("device under attestation")
- Verifier (Network Management Station)
- "Explicit Attestation" is the TCG term for a static (platform) statement.
- "Implicit Attestation" is the TCG term for a session statement.
- Reference Integrity Measurements (RIM), which are signed ~~m~~by the device manufacturer and integrated into firmware.
- Quotes: measured values (having been signed), and RIMs
- Reference Integrity Values (RIV)
- devices have an Initial Attestation Key (IAK), which is provisioned at the same time as the IDevID.
- PCR - Platform Configuration Registry (deals with hash chains)

The TCG document builds upon a number of IETF technologies: SNMP (Attestation MIB), YANG, XML, JSON, CBOR, NETCONF, RESTCONF, CoAP, TLS and SSH. The TCG document leverages the 802.1AR IDevID and LDevID processes.

5.2. Android Keystore system

[keystore] describes a system used in smart phones that run the Android operation system. The system is primarily a software container to contain and control access to cryptographic keys, and therefore provides many of the same functions that a hardware Trusted Platform Module might provide.

Commented [DT11]: Do we know WHY they're declared out of scope? Since just based on the "The use case described is…" paragraph above, these would all be in scope.

Commented [DT12]: In my view, this paragraph isn't part of the use case per se, this is a partial solution to a use case. And it only addresses part of the stated use case (being the paragraph I mentioned above), i.e. the part that's available at boot time, which might be much less than the part that's available at network connection time, especially if the network connection is a user-triggered VPN connection to the network.

Commented [DT13]: The term "static statement" does not appear in the terminology section, only "static attestation". I would find the term "platform attestation" far more clear, and the fact that you had to say "(platform)" implies you would agree ☺

Commented [DT14]: Undefined term

On hardware which is supported, the Android Keystore will make use of whatever trusted hardware is available, including use of a Trusted Execution Environment (TEE) or Secure Element (SE)~~)~~. The Keystore therefore abstracts the hardware, and guarantees to applications that the same APIs can be used on both more and less capable devices.

A great deal of focus from the Android Keystore seems to be on providing fine-grained authorization of what keys can be used by which applications.

XXX - clearly there must be additional (intended?) use cases that provide some kind of attestation.

Android 9 on Pixel 2 and 3 can provided protected confirmation messages. This uses hardware access from the TPM/TEE to display a message directly to the user, and receives confirmation directly from the user. A hash of the contents of the message can provided in an attestation that the device provides.

In addition, the Android Keystore provides attestation information about itself for use by FIDO.

QUOTE: Finally, the Verified Boot state is included in key attestation certificates (provided by Keymaster/Strongbox) in the deviceLocked and verifiedBootState fields, which can be verified by apps as well as passed onto backend services to remotely verify boot integrity [**21]

5.3. Fast IDentity Online (FIDO) Alliance

The FIDO Alliance [fido] has a number of specifications aimed primarily at eliminating the need for passwords for authentication to online services. The goal is to leverage asymmetric cryptographic operations in common browser and smart-phone platforms so that users can easily authentication.

FIDO specifications extend to various hardware second factor authentication devices.

Terminology includes:

- "relying party" validates a claim
- "relying party application" makes FIDO Authn calls
- "browser" provides the Web Authentication JS API
- "platform" is the base system

- o "internal authenticator" is some credential built-in to the device

- o "external authenticator" may be connected by USB, Bbluetooth, WwiFfi, and may be an stand-alone device, USB connected key, phone or watch.

FIDO2 had a Key Attestation Format [fidoattestation], and a Signature Format [fidosignature], but these have been combined into the W3C document [fido_w3c] specification.

A FIDO use case involves a relying party that having a attestation on the biometric system that identifies a human. It is the state of the biometric system that is being attested to, not the identity of the human.

Commented [DT15]: Can't parse grammar

FIDO does provides a transport in the form of the WebAuthn and FIDO CTAP protocols.

According to [fidotechnote] FIDO uses attestation to make claims about the kind of device which is be used to enroll. Keypairs are generated on a per-device _model_ basis, with a certificate having a trust chain that leads back to a well-known root certificate. It is expected that as many as 100,000 devices in a production run would have the same public and private key pair. One assumes that this is stored in a tamper-proof TPM so it is relatively difficult to get this key out. The use of this key attests to the ~~the~~ device type, and the kind of protections for keys that the relying party may assume, not to the identity of the end user.

6. Privacy Considerations.

TBD

7. Security Considerations

TBD.

8. IANA Considerations

TBD.

9. Acknowledgements

10. References

10.1. 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>.

10.2. Informative References

[android_security] Kralevich, R., "The Android Platform Security Model", n.d., <https://arxiv.org/pdf/1904.05572.pdf>.

[fido] FIDO Alliance, ., "FIDO Specification Overview", n.d., <https://fidoalliance.org/specifications/>.

[fido_w3c] W3C, ., "Web Authentication: An API for accessing Public Key Credentials Level 1", n.d., <https://www.w3.org/TR/webauthn-1/>.

[fidoattestation] FIDO Alliance, ., "FIDO 2.0: Key Attestation", n.d., <https://fidoalliance.org/specs/fido-v2.0-ps-20150904/fido-key-attestation-v2.0-ps-20150904.html>.

[fidosignature] FIDO Alliance, ., "FIDO 2.0: Signature Format", n.d., <https://fidoalliance.org/specs/fido-v2.0-ps-20150904/fido-signature-format-v2.0-ps-20150904.html>.

[fidotechnote] FIDO Alliance, ., "FIDO TechNotes: The Truth about Attestation", n.d., <https://fidoalliance.org/fido-technotes-the-truth-about-attestation/>.

[I-D.tschofenig-rats-psa-token] Tschofenig, H., Frost, S., Brossard, M., and A. Shaw, "Arm's Platform Security Architecture (PSA) Attestation Token", draft-tschofenig-rats-psa-token-01 (work in progress), April 2019.

[keystore] Google, ., "Android Keystore System", n.d., <https://developer.android.com/training/articles/keystore>.

Appendix A. Changes

- added comments from Guy, Jessica, Henk and Ned on TCG description.

Author's Address

Michael Richardson
Sandelman Software Works

Email: mcr+ietf@sandelman.ca