RATS Working Group
Internet-Draft
Intended status: Informational
Expires: December 5, 2020

G. Fedorkow, Ed.
Juniper Networks, Inc.
E. Voit
Cisco Systems, Inc.
J. Fitzgerald-McKay
National Security Agency

June 03, 2020

TPM-based Network Device Remote Integrity Verification
draft-ietf-rats-tpm-based-network-device-attest-00

Abstract

This document describes a workflow for remote attestation of integrity of network devices.

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 5, 2020.

Copyright Notice

Table of Contents

1. Introduction

There are many aspects to consider in fielding a trusted computing device, from operating systems to applications. Mechanisms to prove that a device installed at a customer's site is authentic (i.e., not counterfeit) and has been configured with authorized software, all as part of a trusted supply chain, are just a few of the many aspects which need to be considered concurrently to have confidence that a device is truly trustworthy.

A generic architecture for remote attestation has been defined in [I-D.ietf-rats-architecture]. Additionally, the use case for remotely attesting networking devices is within Section 6 of [I-D.richardson-rats-usecases]. However, two these documents do not provide sufficient guidance for equipment vendors and network operators and to design, build, and deploy interoperable platforms.

The intent of this document is to provide such guidance. It does this by outlining the Remote Integrity Verification (RIV) problem, and then identifies elements that are necessary to get the complete, scalable attestation procedure working with commercial networking products such as Routers and Switches. An underlying assumption will be the availability within the device of a Trusted Platform Module (TPM) compliant cryptoprocessor to enable the remote trustworthy assessment of the device's software and hardware.

1.1. Terminology

A number of terms are reused from [I-D.ietf-rats-architecture]. These include: Appraisal Policy for Attestation Result, Attestation Result, Attester, Endorser, Evidence, Relying Party, Verifier, Verifier Owner.

Additionally, this document defines the following terms:

Attestation: the process of creating, conveying and appraising assertions about Platform trustworthiness characteristics, including supply chain trust, identity, platform provenance, software configuration, hardware configuration, platform composition, compliance to test suites, functional and assurance evaluations, etc.

The goal of attestation is simply to assure an administrator that the software that was launched when the device was last started is the same as the software that the device vendor initially shipped.

Within the Trusted Computing Group context, attestation is the process by which an independent Verifier can obtain cryptographic proof as to the identity of the device in question, evidence of the integrity of software loaded on that device when it started up, and then verify that what's there is what's supposed to be there. For networking equipment, a verifier capability can be embedded in a Network Management Station (NMS), a posture collection server, or other network analytics tool (such as a software asset management solution, or a threat detection and mitigation tool, etc.). While informally referred to as attestation, this document focuses on a subset defined here as Remote Integrity Verification (RIV). RIV takes a network equipment centric perspective that includes a set of protocols and procedures for determining whether a particular device

was launched with untampered software, starting from Roots of Trust. While there are many ways to accomplish attestation, RIV sets out a specific set of protocols and tools that work in environments commonly found in Networking Equipment. RIV does not cover other platform characteristics that could be attested, although it does provide evidence of a secure infrastructure to increase the level of trust in other platform characteristics attested by other means (e.g., by Entity Attestation Tokens [I-D.ietf-rats-eat].

1.2. Document Organization

The remainder of this document is organized into several sections:

- The remainder of this section covers goals and requirements, plus a top-level overview
- The Solution Overview section outlines how RIV works
- The Standards Components section links components of RIV to normative standards.
- Supporting material is in an appendix at the end.

1.3. Goals

Network operators benefit from a trustworthy attestation mechanism that provides assurance that their network comprises authentic equipment, and has loaded software free of known vulnerabilities and unauthorized tampering. In line with the overall goal of assuring integrity, attestation can be used for asset management, vulnerability and compliance assessment, plus configuration management.

As a part of a trusted supply chain, the RIV attestation workflow outlined in this document is intended to meet the following high-level goals:

- Provable Device Identity - This specification requires that an attesting device includes a cryptographic identifier unique to each device. Effectively this means that the TPM or equivalent cryptoprocessor must be so provisioned during the manufacturing cycle.
- Software Inventory - A key goal is to identify the software release installed on the attesting device, and to provide evidence that the software stored within hasn't been altered

- Verifiability - Verification of software and configuration of the device shows that the software that was authorized for installation by the administrator has actually has been launched.

In addition, RIV is designed to operate in a centralized environment, such as with a central authority that manages and configures a number of network devices, or 'peer-to-peer', where network devices independently verify one another to establish a trust relationship. (See Section 3.3 below, and also [I-D.voit-rats-trusted-path-routing])

1.4. Description of Remote Integrity Verification (RIV)

Attestation requires two interlocking services between the Attester network device and the Verifier::

- Platform Identity, the mechanism providing trusted identity, can reassure network managers that the specific devices they ordered from authorized manufacturers for attachment to their network are those that were installed, and that they continue to be present in their network. As part of the mechanism for Platform Identity, cryptographic proof of the identity of the manufacturer is also provided.

- Software Measurement is the mechanism that reports the state of mutable software components on the device, and can assure network managers that they have known, untampered software configured to run in their network.

Using these two interlocking services, RIV provides a procedure that assures a network operator that the equipment in their network can be reliably identified, and that untampered software of a known version is installed on each endpoint. Equipment in the network includes devices that make up the network itself, such as routers, switches and firewalls.

RIV includes several major processes:

1. Creation of Evidence is the process whereby an Attester generates cryptographic proof (Evidence) of claims about platform properties. In particular, the platform identity and its software configuration are both of critical importance

2. Platform Identification refers to the mechanism assuring the Relying Party (ultimately, a network administrator) of the identity of devices that make up their network, and that their manufacturers are known.

3. Software used to boot a platform can be described as a chain of measurements, started by a Root of Trust for Measurement, that normally ends when the system software is loaded. A measurement signifies the identity, integrity and version of each software component registered with an attesting device's TPM, so that the subsequent appraisal stage can determine if the software installed is authentic, up-to-date, and free of tampering.

4. Conveyance of Evidence reliably transports at least the minimum amount of Evidence from Attester to a Verifier to allow a management station to perform a meaningful appraisal in Step 5. The transport is typically carried out via a management network. The channel must provide integrity and authenticity, and, in some use cases, may also require confidentiality.

5. Finally, Appraisal of Evidence occurs. As the Verifier and Relaying Party roles are often combined within RIV, this is the process of verifying the Evidence received by a Verifier from the Attesting device, and using an Appraisal Policy to develop an Attestation Result, used to inform decision making. In practice, this means comparing the device measurements reported as Evidence with the Attester configuration expected by the Verifier. Subsequently the Appraisal Policy for Attestation Results might match what was found against Reference Integrity Measurements (aka Golden Measurements) which representing the intended configured state of the connected device.

All implementations supporting this RIV specification require the support of the following three technologies : 1. Identity: Platform identity can be based on IEEE 802.1AR Device Identity [IEEE-802-1AR], coupled with careful supply-chain management by the manufacturer. The DevID certificate contains a statement by the manufacturer that establishes the identity of the device as it left the factory. Some applications with a more-complex post-manufacture supply chain (e.g. Value Added Resellers), or with different privacy concerns, may want to use alternate mechanisms for platform authentication (for example, TCG Platform Certificates [Platform-Certificates]).

1. Platform Attestation provides evidence of configuration of software elements present in the device. This form of attestation can be implemented with TPM PCR, Quote and Log mechanisms, which provide an authenticated mechanism to report what software was started on the device through the boot cycle. Successful attestation requires an unbroken chain from a boot-time root of trust through all layers of software needed to bring the device to an operational state.

2. Reference Integrity Measurements must be conveyed from the Endorser (the entity accepted as the software authority, often the manufacturer for embedded systems) to the system in which verification will take place

1.5. Solution Requirements

Remote Integrity Verification must address the "Lying Endpoint" problem, in which malicious software on an endpoint may subvert the intended function, and also prevent the endpoint from reporting its compromised status. (See Section 5 for further Security Considerations)

RIV attestation is designed to be simple to deploy at scale. RIV should work "out of the box" as far as possible, that is, with the fewest possible provisioning steps or configuration databases needed at the end-user's site, as network equipment is often required to "self-configure", to reliably reach out without manual intervention to prove its identity and operating posture, then download its own configuration. See [RFC8572] for an example of Secure Zero Touch Provisioning.

1.6. Scope

Remote Attestation is a very general problem that could apply to most network-connected computing devices. However, this document includes several assumptions that limit the scope to Network Equipment (e.g. routers, switches and firewalls):

- This solution is for use in non-privacy-preserving applications (for example, networking, Industrial IoT), avoiding the need for a Privacy Certificate Authority for attestation keys [AIK-Enrollment] or TCG Platform Certificates [Platform-Certificates]

- This document assumes network protocols that are common in networking equipment such as YANG [RFC7950] and NETCONF [RFC6241], but not generally used in other applications.

- The approach outlined in this document mandates the use of TPM 1.2 or TPM 2.0. Other roots of trust could be used with the same information flow, although different data structures would likely be called for.

1.6.1. Out of Scope

- Run-Time Attestation: Run-time attestation of Linux or other multi-threaded operating system processes considerably expands the scope of the problem. Many researchers are working on that problem, but this document defers the run-time attestation problem.

- Multi-Vendor Embedded Systems: Additional coordination would be needed for devices that themselves comprise hardware and software from multiple vendors, integrated by the end user.

- Processor Sleep Modes: Network equipment typically does not "sleep", so sleep and hibernate modes are not considered. Although out of scope for RIV, Trusted Computing Group specifications do encompass sleep and hibernate states.

- Virtualization and Containerization: These technologies are increasingly used in Network equipment, but are not considered in this revision of the document.

2. Solution Outline

2.1. RIV Software Configuration Attestation using TPM

RIV Attestation is a process which can be used to determine the identity of software running on a specifically-identified device. Remote Attestation is broken into two phases, shown in Figure 1:

- During system startup, each distinct software object is "measured". Its identity, hash (i.e. cryptographic digest) and version information is recorded in a log. Hashes are also extended, or cryptographically folded, into the TPM, in a way that can be used to validate the log entries. The measurement process generally follows the Chain of Trust model used in Measured Boot, where each stage of the system measures the next one, and extends its measurement into the TPM, before launching it.

- Once the device is running and has operational network connectivity, a separate, trusted Verifier will interrogate the network device to retrieve the logs and a copy of the digests collected by hashing each software object, signed by an attestation private key known only to the TPM.

The result is that the Verifier can verify the device's identity by checking the certificate containing the TPM's attestation public key, and can validate the software that was launched by comparing digests

in the log with known-good values, and verifying their correctness by comparing with the signed digests from the TPM.

It should be noted that attestation and identity are inextricably linked; signed Evidence that a particular version of software was loaded is of little value without cryptographic proof of the identity of the Attester producing the Evidence.

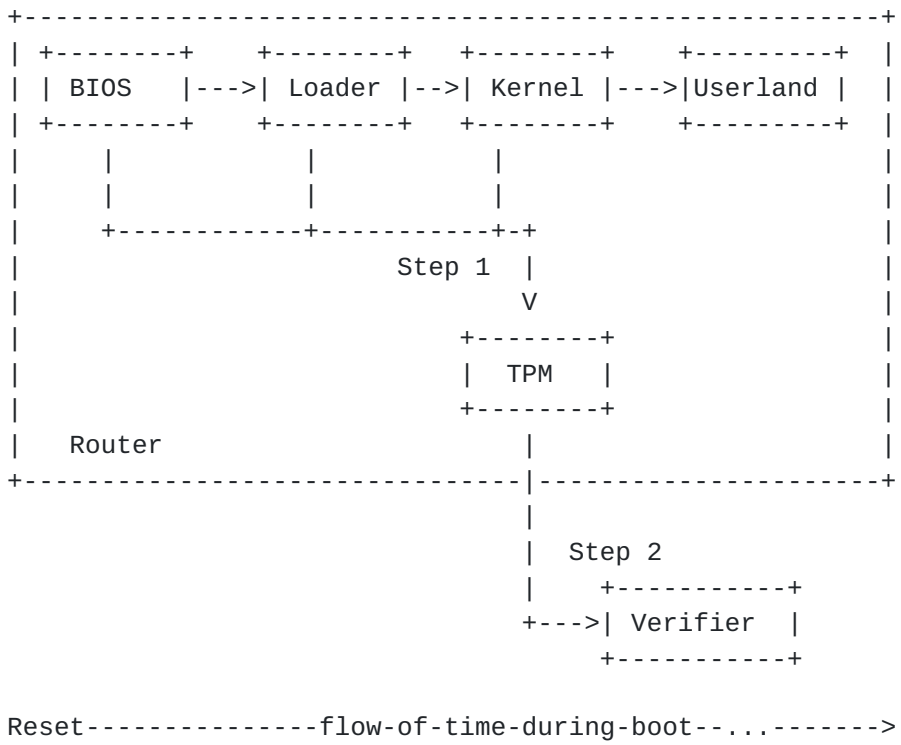

Figure 1: RIV Attestation Model

In Step 1, measurements are "extended" into the TPM as processes start. In Step 2, signed PCR digests are retrieved from the TPM for off-box analysis after the system is operational.

2.1.1. What Does RIV Attest?

TPM attestation is strongly focused around Platform Configuration Registers (PCRs), but those registers are only vehicles for certifying accompanying Evidence, conveyed in log entries. It is the hashes in log entries are extended into PCRs, where they can be retrieved in the form of a Quote signed by a key known only to the TPM (xref). The use of multiple PCRs serves only to provide some independence between different classes of object, so that one class of objects can be updated without changing the extended hash for other classes. Although PCRs can be used for any purpose, this

section outlines the objects within the scope of this document which may be extended into the TPM.

In general, PCRs are organized to independently attest three classes of object:

- Code, i.e., instructions to be executed by a CPU.
- Configuration - Many devices offer numerous options controlled by non-volatile configuration variables which can impact the device's security posture. These settings may have vendor defaults, but often can be changed by administrators, who may want to verify via attestation that the settings they intend are still in place.
- Credentials - Administrators may wish to verify via attestation that keys (and other credentials) outside the Root of Trust have not be subject to unauthorized tampering. (By definition, keys inside the root of trust can't be verified independently)

The TCG PC Client Platform Firmware Profile Specification [PC-Client-BIOS-TPM-2.0] gives considerable detail on what is to be measured during the boot phase of a platform boot using a UEFI BOIS (www.uefi.org), but the goal is simply to measure every bit of code executed in the process of starting the device, along with any configuration information related to security posture. Table XX summarizes the functions that are measured, and how this document recommends they be allocated to PCRs. It's important to note that each PCR may contain results from dozens (or even thousands) of individual measurements.

Function	Allocated PCR # Code	Allocated PCR # Configuration
BIOS Static Root of Trust, plus embedded Option ROMs and drivers	0	1
Pluggable Option ROMs to initialize and configure add-in devices	2	3
Boot Manager code and configuration (UEFI uses a separate module to implement policies for selecting among a variety of potential boot devices). This PCR records boot attempts, and identifies what resources were used to boot the OS.	4	5
Vendor Specific Measurements during boot	6	6
Secure Boot Policy. This PCR records keys and configuration used to validate the OS loader		7
OS Loader (e.g GRUB2 for Linux)	8	9
Reserved for OS (e.g. Linux IMA)	10	10

Figure 2: Attested Objects

2.2. RIV Keying

RIV attestation relies on two keys:

- An identity key is required to certify the identity of the Attester itself. RIV specifies the use of an IEEE 802.1AR DevID [IEEE-802-1AR], signed by the device manufacturer, containing the device serial number.

- An Attestation Key is required to sign the Quote generated by the TPM to report evidence of software configuration.

In TPM application, the Attestation key must be protected by the TPM, and the DevID should be as well. Depending on other TPM configuration procedures, the two keys may be different. Some of the considerations are outlined in TCG Guidance for Securing Network Equipment [NetEq].

TCG Guidance for Securing Network Equipment specifies further conventions for these keys:

- When separate Identity and Attestation keys are used, the Attestation Key (AK) and its x.509 certificate should parallel the DevID, with the same device ID information as the DevID certificate (i.e., the same Subject Name and Subject Alt Name, even though the key pairs are different). This allows a quote from the device, signed by an AK, to be linked directly to the device that provided it, by examining the corresponding AK certificate.

- Network devices that are expected to use secure zero touch provisioning as specified in [RFC8572]) must be shipped by the manufacturer with pre-provisioned keys (Initial DevID and AK, called IDevID and IAK). Inclusion of an DevID and IAK by a vendor does not preclude a mechanism whereby an Administrator can define Local Identity and Attestation Keys (LDevID and LAK) if desired.

2.3. RIV Information Flow

RIV workflow for networking equipment is organized around a simple use-case, where a network operator wishes to verify the integrity of software installed in specific, fielded devices. This use-case implies several components:

1. The Attesting Device, which the network operator wants to examine.

2. A Verifier (which might be a network management station) somewhere separate from the Device that will retrieve the information and analyze it to pass judgment on the security posture of the device.

3. A Relying Party, which can act on Attestation results. Interaction between the Relying Party and the Verifier is considered out of scope for RIV.

4. Signed Reference Integrity Manifests (RIMs), containing Reference Integrity Measurements, can either be created by the device manufacturer and shipped along with the device as part of its software image, or alternatively, could be obtained several other ways (direct to the Verifier from the manufacturer, from a third party, from the owner's observation of what's thought to be a "known good system", etc.). Retrieving RIMs from the device itself allows attestation to be done in systems which may not have access to the public internet, or by other devices that are not management stations per-se (e.g., a peer device; See

Section 3.1.3). If reference measurements are obtained from multiple sources, the Verifier may need to evaluate the relative level of trust to be placed in each source in case of a discrepancy.

These components are illustrated in Figure 2.

A more-detailed taxonomy of terms is given in [I-D.ietf-rats-architecture]

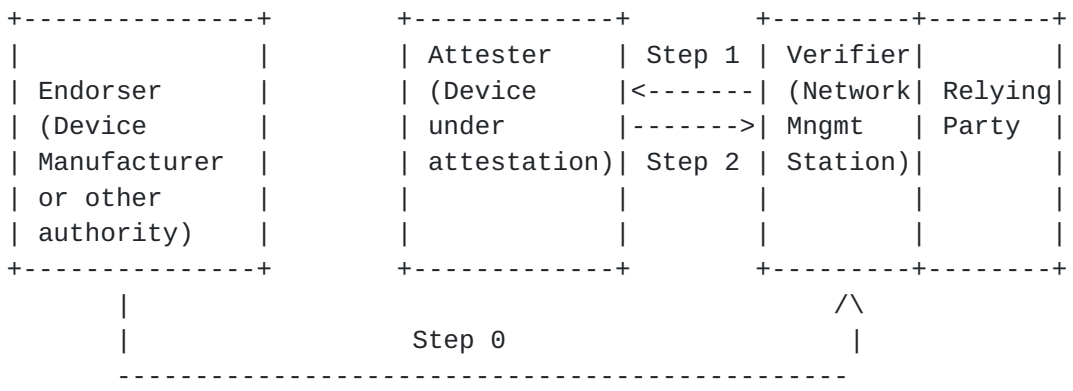

Figure 3: RIV Reference Configuration for Network Equipment

In Step 0, The Asserter (the device manufacturer) provides a Software Image accompanied by one or more Reference Integrity Manifests (RIMs) to the Attester (the device under attestation) signed by the asserter. In Step 1, the Verifier (Network Management Station), on behalf of a Relying Party, requests Identity, Measurement Values (and possibly RIMs) from the Attester. In Step 2, the Attester responds to the request by providing a DevID, quotes (measured values), and optionally RIMs, signed by the Attester.

The following standards components may be used:

1. TPM Keys are configured according to [Platform-DevID-TPM-2.0], [PC-Client-BIOS-TPM-1.2], or [Platform-ID-TPM-1.2]

2. Measurements of firmware and bootable modules may be taken according to TCG PC Client [PC-Client-BIOS-TPM-2.0] and Linux IMA [IMA]

3. Device Identity is managed by IEEE 802.1AR certificates [IEEE-802-1AR], with keys protected by TPMs.

4. Attestation logs may be formatted according to the Canonical Event Log format [Canonical-Event-Log], although other specialized formats may be used.

5. Quotes are retrieved from the TPM according to TCG TAP Information Model [TAP]. While the TAP IM gives a protocol-independent description of the data elements involved, it's important to note that quotes from the TPM are signed inside the TPM, so must be retrieved in a way that does not invalidate the signature, as specified in [I-D.ietf-rats-yang-tpm-charra], to preserve the trust model. (See Section 5 Security Considerations).

6. Reference Integrity Measurements may be encoded as CoSWID tags, as defined in the TCG RIM document [RIM], compatible with NIST IR 8060 [NIST-IR-8060] and the IETF CoSWID draft [I-D.ietf-sacm-coswid]. See Section 2.4.1.

2.4. RIV Simplifying Assumptions

This document makes the following simplifying assumptions to reduce complexity:

- The product to be attested is shipped with an IEEE 802.1AR DevID and an Initial Attestation Key (IAK) with certificate. The IAK cert contains the same identity information as the DevID (specifically, the same Subject Name and Subject Alt Name, signed by the manufacturer), but it's a type of key that can be used to sign a TPM Quote. This convention is described in TCG Guidance for Securing Network Equipment [NetEq]. For network equipment, which is generally non-privacy-sensitive, shipping a device with both an IDevID and an IAK already provisioned substantially simplifies initial startup. Privacy-sensitive applications may use the TCG Platform Certificate and additional procedures to install identity credentials on the platform after manufacture.

- The product is equipped with a Root of Trust for Measurement, Root of Trust for Storage and Root of Trust for Reporting (as defined in [GloPlaRoT]) that are capable of conforming to the TCG Trusted Attestation Protocol (TAP) Information Model [TAP].

- The vendor will ship Reference Integrity Measurements (i.e., known-good measurements) in the form of signed CoSWID tags [I-D.ietf-sacm-coswid], [SWID], as described in TCG Reference Integrity Measurement Manifest Information Model [RIM].

2.4.1. Reference Integrity Manifests (RIMs)

[I-D.ietf-rats-yang-tpm-charra] focuses on collecting and transmitting evidence in the form of PCR measurements and attestation logs. But the critical part of the process is enabling the verifier to decide whether the measurements are "the right ones" or not.

While it must be up to network administrators to decide what they want on their networks, the software supplier should supply the Reference Integrity Measurements that may be used by a verifier to determine if evidence shows known good, known bad or unknown software configurations.

In general, there are two kinds of reference measurements:

1. Measurements of early system startup (e.g., BIOS, boot loader, OS kernel) are essentially single threaded, and executed exactly once, in a known sequence, before any results could be reported. In this case, while the method for computing the hash and extending relevant PCRs may be complicated, the net result is that the software (more likely, firmware) vendor will have one known good PCR value that "should" be present in the relevant PCRs after the box has booted. In this case, the signed reference measurement could simply list the expected hashes for the given version. However, a RIM that contains the intermediate hashes can be useful in debugging cases where the expected final hash is not the one reported.

2. Measurements taken later in operation of the system, once an OS has started (for example, Linux IMA[IMA]), may be more complex, with unpredictable "final" PCR values. In this case, the Verifier must have enough information to reconstruct the expected PCR values from logs and signed reference measurements from a trusted authority.

In both cases, the expected values can be expressed as signed SWID or CoSWID tags, but the SWID structure in the second case is somewhat more complex, as reconstruction of the extended hash in a PCR may involve thousands of files and other objects.

The TCG has published an information model defining elements of reference integrity manifests under the title TCG Reference Integrity Manifest Information Model [RIM]. This information model outlines how SWID tags should be structured to allow attestation, and defines "bundles" of SWID tags that may be needed to describe a complete software release. The RIM contains some metadata relating to the software release it belongs to, plus hashes for each individual file or other object that could be attested.

TCG has also published the PC Client Reference Integrity Measurement specification [PC-Client-RIM], which focuses on a SWID-compatible format suitable for expressing expected measurement values in the specific case of a UEFI-compatible BIOS, where the SWID focus on files and file systems is not a direct fit. While the PC Client RIM is not directly applicable to network equipment, many vendors do use a conventional UEFI BIOS to launch their network OS.

2.4.2. Attestation Logs

Quotes from a TPM can provide evidence of the state of a device up to the time the evidence was recorded, but to make sense of the quote in most cases an event log that identifies which software modules contributed which values to the quote during startup must also be provided. The log must contain enough information to demonstrate its integrity by allowing exact reconstruction of the digest conveyed in the signed quote (i.e., PCR values).

There are multiple event log formats which may be supported as viable formats of Evidence between the Attester and Verifier:

- Event log exports from [I-D.ietf-rats-yang-tpm-charra]
- IMA Event log file exports [IMA]
- TCG UEFI BIOS event log (TCG EFI Platform Specification for TPM Family 1.1 or 1.2, Section 7 [EFI-TPM])
- TCG Canonical Event Log [Canonical-Event-Log]
- Legacy BIOS event log, although this document is less relevant as UEFI has largely replaced the Legacy BIOS (TCG PC Client Specific Implementation Specification for Conventional BIOS, Section 11.3[PC-Client-BIOS-TPM-1.2])

3. Standards Components

3.1. Prerequisites for RIV

The Reference Interaction Model for Challenge-Response-based Remote Attestation is based on the standard roles defined in [I-D.ietf-rats-architecture]. However additional prerequisites must be established to allow for interoperable RIV use case implementations. These prerequisites are intended to provide sufficient context information so that the Verifier can acquire and evaluate Attester measurements.

3.1.1. Unique Device Identity

A Secure device Identity (DevID) in the form of an IEEE 802.1AR certificate [IEEE-802-1AR] must be provisioned in the Attester's TPMs.

3.1.2. Keys

The Attestation Identity Key (AIK) and certificate must also be provisioned on the Attester according to [Platform-DevID-TPM-2.0], [PC-Client-BIOS-TPM-1.2], or [Platform-ID-TPM-1.2].

The Attester's TPM Keys must be associated with the DevID on the Verifier (see Section 5 Security Considerations).

3.1.3. Appraisal Policy for Evidence

The Verifier must obtain the Appraisal Policy for Evidence. This policy may be in the form of reference measurements (e.g., Known Good Values, CoSWID tags [I-D.birkholz-yang-swid]). These reference measurements will eventually be compared to signed PCR Evidence acquired from an Attester's TPM.

This document does not specify the format or contents for the Appraisal Policy for Evidence. But acquiring this policy may happen in one of two ways:

1. a Verifier obtains reference measurements directly from a Verifier Owner / device configuration authority chosen by the network administrator.

2. Signed reference measurements may be distributed by the Verifier Owner to the Attester. From there, the reference measurement may be acquired by the Verifier.

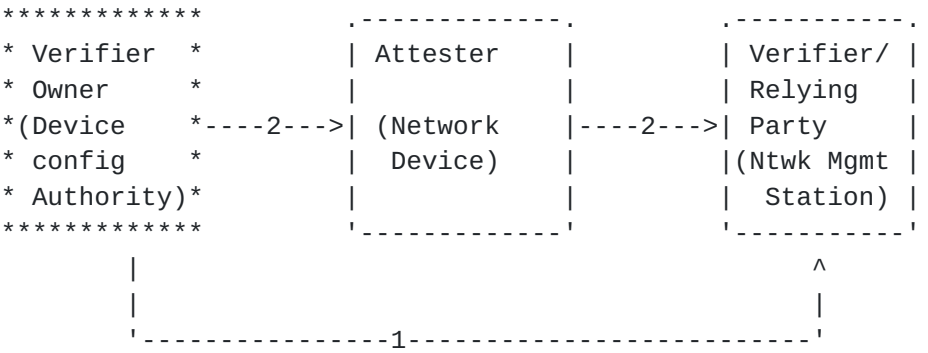

Figure 4: Appraisal Policy for Evidence Prerequisites

In either case the Appraisal Policy for Evidence must be generated, acquired and delivered in a secure way. This includes reference measurements of:

- firmware and bootable modules taken according to TCG PC Client [PC-Client-BIOS-TPM-2.0] and Linux IMA [IMA]

- encoded CoSWID tags signed by the device manufacturer, are as defined in the TCG RIM document [RIM], compatible with NIST IR 8060 [NIST-IR-8060] and the IETF CoSWID draft [I-D.ietf-sacm-coswid].

3.2. Reference Model for Challenge-Response

Once the prerequisites for RIV are met, a Verifier may acquire Evidence from an Attester. The following diagram illustrates a RIV information flow between a Verifier and an Attester. Event times shown correspond to the time types described within Appendix A of [I-D.ietf-rats-architecture]:

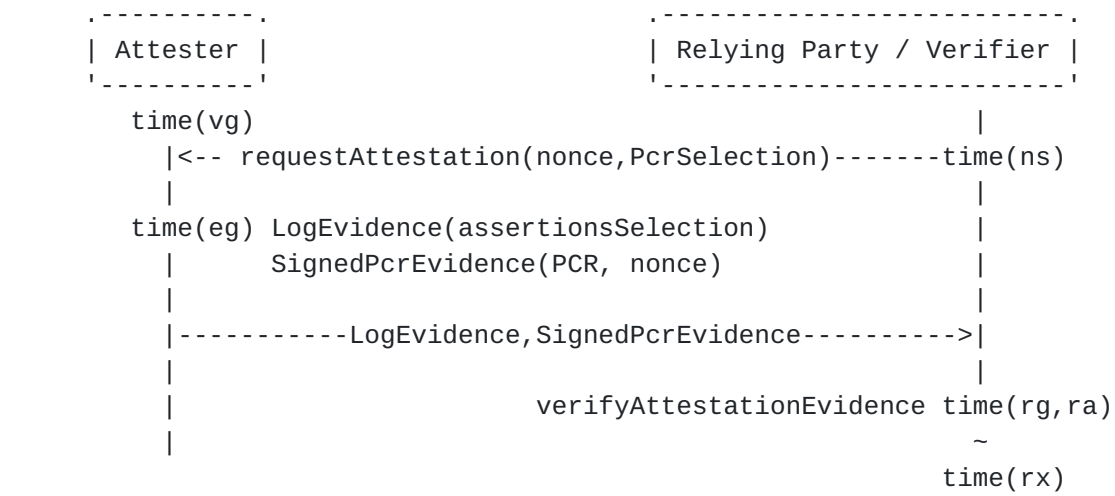

Figure 5: IETF Attestation Information Flow

- time(vg): One or more Attesting Network Device PCRs are extended with measurements.

- time(ns): The Verifier generates a nonce, and makes a request attestation data for one or more PCRs from an Attester. This can be accomplished via a YANG [RFC7950] interface that implements the TCG TAP model (e.g. YANG Module for Basic Challenge-Response-based Remote Attestation Procedures [I-D.ietf-rats-yang-tpm-charra]).

- time(eg): On the Attester, measured values are retrieved from the Attester's TPM. This requested PCR evidence is signed by the Attestation Identity Key (AIK) associated with the DevID. Quotes are retrieved according to TCG TAP Information Model [TAP]. While the TAP IM gives a protocol-independent description of the data elements involved, it's important to note that quotes from the TPM are signed inside the TPM, so must be retrieved in a way that does not invalidate the signature, as specified in [I-D.ietf-rats-yang-tpm-charra], to preserve the trust model. (See Section 5 Security Considerations).

  * At the same time, for any PCRs where known good values might not be known by the Verifier, the Attester collects log evidence showing what values have been extended into that PCR. Attestation logs are formatted according to the Canonical Event Log format [Canonical-Event-Log].

- Collected Evidence is passed from the Attester to the Verifier

- time(rg,ra): The Verifier reviews the Evidence and takes action as needed. As the Relying Party and Verifier are assumed co-resident, this can happen in one step.

  * If the signed PCR values do not match either KGVs, or the set of log entries which have extended a particular PCR, the device should not be trusted.

  * If the set of log entries are not seen as acceptable by the Appraisal Policy for Evidence, the device should not be trusted.

  * If the AIK signature is not correct, or freshness such as that provided by the nonce is not included in the response, the device should not be trusted.

- time(rx): At some point after the verification of Evidence, the Attester can no longer be considered Attested as trustworthy.

3.2.1. Transport and Encoding

Network Management systems may retrieve signed PCR based Evidence as shown in Figure 5, and can be accomplished via:

- XML, JSON, or CBOR encoded Evidence, using

- RESTCONF or NETCONF transport, over a

- TLS or SSH secure tunnel

Retrieval of Log Evidence will be via log interfaces on the network device. (For example, see [I-D.ietf-rats-yang-tpm-charra]).

3.3. Centralized vs Peer-to-Peer

Figure 5 above assumes that the Verifier is implicitly trusted, while the Attesting device is not. In a Peer-to-Peer application such as two routers negotiating a trust relationship [I-D.voit-rats-trusted-path-routing], the two peers can each ask the other to prove software integrity. In this application, the information flow is the same, but each side plays a role both as an Attester and a Verifier. Each device issues a challenge, and each device responds to the other's challenge, as shown in Figure 6. Peer-to-peer challenges, particularly if used to establish a trust relationship between routers, require devices to carry their own signed reference measurements (RIMs) so that each device has everything needed for attestation, without having to resort to a central authority.

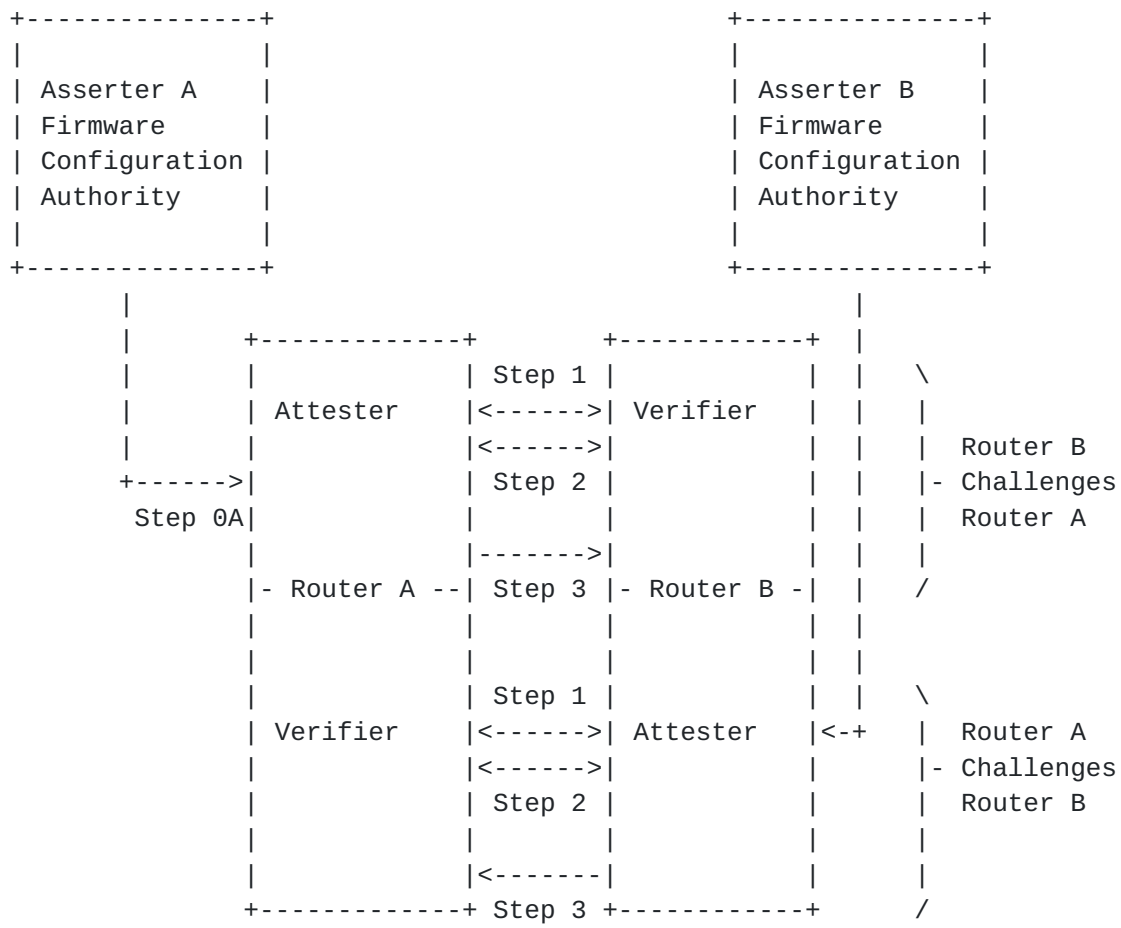

Figure 6: Peer-to-Peer Attestation Information Flow

In this application, each device may need to be equipped with signed RIMs to act as an Attester, and also a selection of trusted x.509 root certificates to allow the device to act as a Verifier. An existing link layer protocol such as 802.1x [IEEE-802.1x] or 802.1AE [IEEE-802.1ae], with Evidence being enclosed over a variant of EAP [RFC3748] or LLDP [LLDP] are suitable methods for such an exchange.

4. Privacy Considerations

Networking Equipment such as routers, switches and firewalls has a key role to play in guarding the privacy of individuals using the network:

- Packets passing through the device must not be sent to unauthorized destinations. For example

- Routers often act as Policy Enforcement Points, where individual subscribers may be checked for authorization to access a network. Subscriber login information must not be released to unauthorized parties.

- Networking Equipment is often called upon to block access to protected resources from unauthorized users.

- Routing information, such as the identity of a router's peers, must not be leaked to unauthorized neighbors.

- If configured, encryption and decryption of traffic must be carried out reliably, while protecting keys and credentials.

Functions that protect privacy are implemented as part of each layer of hardware and software that makes up the networking device. In light of these requirements for protecting the privacy of users of the network, the Network Equipment must identify itself, and its boot configuration and measured device state (for example, PCR values), to the Equipment's Administrator, so there's no uncertainty as to what function each device and configuration is configured to carry out. This allows the administrator to ensure that the network provides individual and peer privacy guarantees.

RIV specifically addresses the collection information from enterprise network devices by an enterprise network. As such, privacy is a fundamental concern for those deploying this solution, given EU GDPR, California CCPA, and many other privacy regulations. The enterprise should implement and enforce their duty of care.

See [NetEq] for more context on privacy in networking devices

5. Security Considerations

Attestation results from the RIV procedure are subject to a number of attacks:

- Keys may be compromised
- A counterfeit device may attempt to impersonate (spoof) a known authentic device
- Man-in-the-middle attacks may be used by a counterfeit device to attempt to deliver responses that originate in an actual authentic device
- Replay attacks may be attempted by a compromised device

Trustworthiness of RIV attestation depends strongly on the validity of keys used for identity and attestation reports. RIV takes full advantage of TPM capabilities to ensure that results can be trusted.

Two sets of keys are relevant to RIV attestation

- A DevID key is used to certify the identity of the device in which the TPM is installed.
- An Attestation Key (AK) key signs attestation reports, (called 'quotes' in TCG documents), used to provide evidence for integrity of the software on the device.

TPM practices usually require that these keys be different, as a way of ensuring that a general-purpose signing key cannot be used to spoof an attestation quote.

In each case, the private half of the key is known only to the TPM, and cannot be retrieved externally, even by a trusted party. To ensure that's the case, specification-compliant private/public key-pairs are generated inside the TPM, where they're never exposed, and cannot be extracted (See [Platform-DevID-TPM-2.0]).

Keeping keys safe is just part of attestation security; knowing which keys are bound to the device in question is just as important.

While there are many ways to manage keys in a TPM (See [Platform-DevID-TPM-2.0]), RIV includes support for "zero touch" provisioning (also known as zero-touch onboarding) of fielded devices (e.g. Secure ZTP, [RFC8572]), where keys which have predictable trust properties are provisioned by the device vendor.

Device identity in RIV is based on IEEE 802.1AR DevID. This specification provides several elements

- A DevID requires a unique key pair for each device, accompanied by an x.509 certificate

- The private portion of the DevID key is to be stored in the device, in a manner that provides confidentiality (Section 6.2.5 [IEEE-802-1AR])

The x.509 certificate contains several components

- The public part of the unique DevID key assigned to that device

- An identifying string that's unique to the manufacturer of the device. This is normally the serial number of the unit, which might also be printed on label on the device.

- The certificate must be signed by a key traceable to the manufacturer's root key.

With these elements, the device's manufacturer and serial number can be identified by analyzing the DevID certificate plus the chain of intermediate certs leading back to the manufacturer's root certificate. As is conventional in TLS connections, a nonce must be signed by the device in response to a challenge, proving possession of its DevID private key.

RIV uses the DevID to validate a TLS connection to the device as the attestation session begins. Security of this process derives from TLS security, with the DevID providing proof that the TLS session terminates on the intended device. [RFC8446].

Evidence of software integrity is delivered in the form of a quote signed by the TPM itself. Because the contents of the quote are signed inside the TPM, any external modification (including reformatting to a different data format) will be detected as tampering.

A critical feature of the YANG model described in [I-D.ietf-rats-yang-tpm-charra] is the ability to carry TPM data structures in their native format, without requiring any changes to the structures as they were signed and delivered by the TPM. While alternate methods of conveying TPM quotes could compress out redundant information, or add an additional layer of signing using external keys, the important part is to preserve the TPM signing, so that tampering anywhere in the path between the TPM itself and the Verifier can be detected.

Prevention of spoofing attacks against attestation systems is also important. There are two cases to consider:

- The entire device could be spoofed, that is, when the Verifier goes to verify a specific device, it might be redirected to a different device. Use of the 802.1AR identity in the TPM ensures that the Verifier's TLS session is in fact terminating on the right device.

- A compromised device could respond with a spoofed attestation result, that is, a compromised OS could return a fabricated quote.

Protection against spoofed quotes from a device with valid identity is a bit more complex. An identity key must be available to sign any kind of nonce or hash offered by the verifier, and consequently, could be used to sign a fabricated quote. To block spoofed attestation result, the quote generated inside the TPM must by signed by a key that's different from the DevID, called an Attestation Key (AK).

Given separate Attestation and DevID keys, the binding between the AK and the same device must also be proven to prevent a man-in-the-middle attack (e.g. the 'Asokan Attack' [RFC6813]).

This is accomplished in RIV through use of an AK certificate with the same elements as the DevID (i.e., same manufacturer's serial number, signed by the same manufacturer's key), but containing the device's unique AK public key instead of the DevID public key. [this will require an OID that says the key is known by the CA to be an Attestation key]

These two keys and certificates are used together:

- The DevID is used to validate a TLS connection terminating on the device with a known serial number.

- The AK is used to sign attestation quotes, providing proof that the attestation evidence comes from the same device.

Replay attacks, where results of a previous attestation are submitted in response to subsequent requests, are usually prevented by inclusion of a nonce in the request to the TPM for a quote. Each request from the Verifier includes a new random number (a nonce). The resulting quote signed by the TPM contains the same nonce, allowing the verifier to determine freshness, i.e., that the resulting quote was generated in response to the verifier's specific request. Time-Based Uni-directional Attestation

[I-D.birkholz-rats-tuda] provides an alternate mechanism to verify freshness without requiring a request/response cycle.

Requiring results of attestation of the operating software to be signed by a key known only to the TPM also removes the need to trust the device's operating software (beyond the first measurement; see below); any changes to the quote, generated and signed by the TPM itself, made by malicious device software, or in the path back to the verifier, will invalidate the signature on the quote.

Although RIV recommends that device manufacturers pre-provision devices with easily-verified DevID and AK certs, use of those credentials is not mandatory. IEEE 802.1AR incorporates the idea of an Initial Device ID (IDevID), provisioned by the manufacturer, and a Local Device ID (LDevID) provisioned by the owner of the device. RIV extends that concept by defining an Initial Attestation Key (IAK) and Local Attestation Key (LAK) with the same properties.

Device owners can use any method to provision the Local credentials.

- TCG document [Platform-DevID-TPM-2.0] shows how the initial Attestation keys can be used to certify LDevID and LAK keys. Use of the LDevID and LAK allows the device owner to use a uniform identity structure across device types from multiple manufacturers (in the same way that an "Asset Tag" is used by many enterprises use to identify devices they own). TCG doc [Provisioning-TPM-2.0] also contains guidance on provisioning identity keys in TPM 2.0.

- But device owners can use any other mechanism they want to assure themselves that Local identity certificates are inserted into the intended device, including physical inspection and programming in a secure location, if they prefer to avoid placing trust in the manufacturer-provided keys.

Clearly, Local keys can't be used for secure Zero Touch provisioning; installation of the Local keys can only be done by some process that runs before the device is configured for network operation.

On the other end of the device life cycle, provision should be made to wipe Local keys when a device is decommissioned, to indicate that the device is no longer owned by the enterprise. The manufacturer's Initial identity keys must be preserved, as they contain no information that's not already printed on the device's serial number plate.

In addition to trustworthy provisioning of keys, RIV depends on other trust anchors. (See [GloPlaRoT] for definitions of Roots of Trust.)

- Secure identity depends on mechanisms to prevent per-device secret keys from being compromised. The TPM provides this capability as a Root of Trust for Storage

- Attestation depends on an unbroken chain of measurements, starting from the very first measurement. That first measurement is made by code called the Root of Trust for Measurement, typically done by trusted firmware stored in boot flash. Mechanisms for maintaining the trustworthiness of the RTM are out of scope for RIV, but could include immutable firmware, signed updates, or a vendor-specific hardware verification technique.

- RIV assumes some level of physical defense for the device. If a TPM that has already been programmed with an authentic DevID is stolen and inserted into a counterfeit device, attestation of that counterfeit device may become indistinguishable from an authentic device.

RIV also depends on reliable reference measurements, as expressed by the RIM [RIM]. The definition of trust procedures for RIMs is out of scope for RIV, and the device owner is free to use any policy to validate a set of reference measurements. RIMs may be conveyed out-of-band or in-band, as part of the attestation process (see Section 3.1.3). But for embedded devices, where software is usually shipped as a self-contained package, RIMs signed by the manufacturer and delivered in-band may be more convenient for the device owner.

6. Conclusion

TCG technologies can play an important part in the implementation of Remote Integrity Verification. Standards for many of the components needed for implementation of RIV already exist:

- Platform identity can be based on IEEE 802.1AR Device identity, coupled with careful supply-chain management by the manufacturer.

- Complex supply chains can be certified using TCG Platform Certificates [Platform-Certificates]

- The TCG TAP mechanism can be used to retrieve attestation evidence. Work is needed on a YANG model for this protocol.

- Reference Measurements must be conveyed from the software authority (e.g., the manufacturer) to the system in which verification will take place. IETF CoSWID work forms the basis for this, but new work is needed to create an information model and YANG implementation.

7. IANA Considerations

This memo includes no request to IANA.

8. Appendix

8.1. Layering Model for Network Equipment Attester and Verifier

Retrieval of identity and attestation state uses one protocol stack, while retrieval of Reference Measurements uses a different set of protocols. Figure 5 shows the components involved.

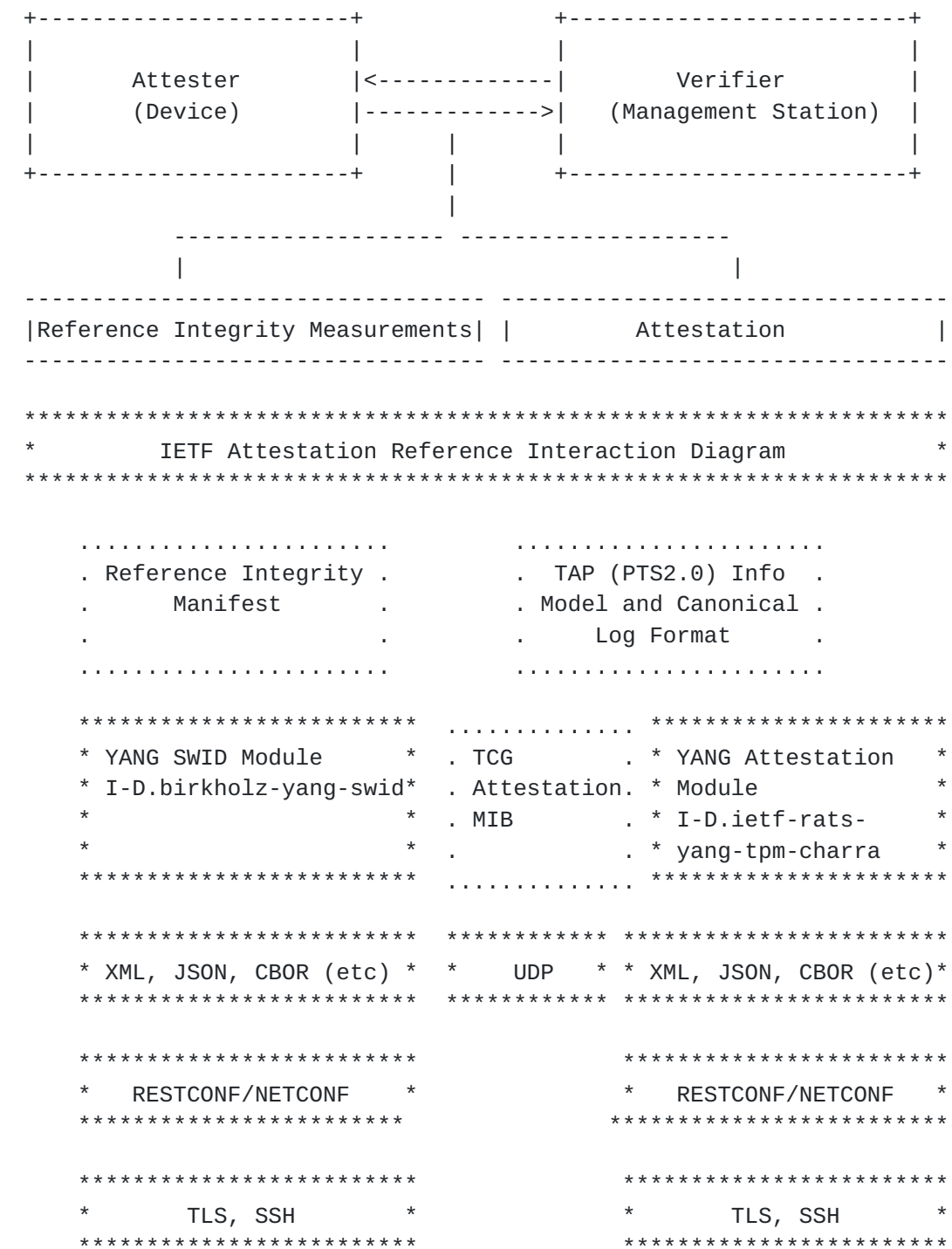

Figure 7: RIV Protocol Stacks

IETF documents are captured in boxes surrounded by asterisks. TCG documents are shown in boxes surrounded by dots. The IETF Attestation Reference Interaction Diagram, Reference Integrity

Manifest, TAP Information Model and Canonical Log Format, and both YANG modules are works in progress. Information Model layers describe abstract data objects that can be requested, and the corresponding response SNMP is still widely used, but the industry is transitioning to YANG, so in some cases, both will be required. TLS Authentication with TPM has been shown to work; SSH authentication using TPM-protected keys is not as easily done [as of 2019]

8.1.1. Why is OS Attestation Different?

Even in embedded systems, adding Attestation at the OS level (e.g. Linux IMA, Integrity Measurement Architecture [IMA]) increases the number of objects to be attested by one or two orders of magnitude, involves software that's updated and changed frequently, and introduces processes that begin in unpredictable order.

TCG and others (including the Linux community) are working on methods and procedures for attesting the operating system and application software, but standardization is still in process.

8.2. Implementation Notes

Table 1 summarizes many of the actions needed to complete an Attestation system, with links to relevant documents. While documents are controlled by several standards organizations, the implied actions required for implementation are all the responsibility of the manufacturer of the device, unless otherwise noted.

Component	Controlling Specification
Make a Secure execution environment o Attestation depends on a secure root of trust for measurement outside the TPM, as well as roots for storage amd reporting inside the TPM. o Refer to TCG Root of Trust for Measurement. o NIST SP 800-193 also provides guidelines on Roots of Trust	TCG RoT UEFI.org
Provision the TPM as described in TCG documents.	TCG TPM DevID TCG Platform Certificate
Put a DevID or Platform Cert in the TPM o Install an Initial Attestation Key at the	TCG TPM DevID TCG Platform

same time so that Attestation can work out of the box	Certificate
o Equipment suppliers and owners may want to implement Local Device ID as well as Initial Device ID	IEEE 802.1AR
Connect the TPM to the TLS stack o Use the DevID in the TPM to authenticate TAP connections, identifying the device	Vendor TLS stack (This action is simply configuring TLS to use the DevID as its trust anchor.)
Make CoSWID tags for BIOS/LoaderLKernel objects o Add reference measurements into SWID tags o Manufacturer should sign the SWID tags o The TCG RIM-IM identifies further procedures to create signed RIM documents that provide the necessary reference information	IETF CoSWID ISO/IEC 19770-2 NIST IR 8060
Package the SWID tags with a vendor software release o A tag-generator plugin such as https://github.com/Labs64/swid-maven-plugin can be used	Retrieve tags with {{I-D.birkholz-yang-swid}} TCG PC Client RIM
Use PC Client measurement definitions to define the use of PCRs (although Windows OS is rare on Networking Equipment, UEFI BIOS is not)	TCG PC Client BIOS
Use TAP to retrieve measurements o Map TAP to SNMP o Map to YANG Use Canonical Log Format	TCG SNMP MIB YANG Module for Basic Attestation TCG Canonical Log Format
Posture Collection Server (as described in IETF SACMs ECP) should request the attestation and analyze the result The Management application might be broken down	

to several more components:		
o A Posture Manager Server which collects reports and stores them in a database		
o One or more Analyzers that can look at the results and figure out what it means.		

Figure 8: Component Status

8.3. Root of Trust for Measurement

The measurements needed for attestation require that the device being attested is equipped with a Root of Trust for Measurement, i.e., some trustworthy mechanism that can compute the first measurement in the chain of trust required to attest that each stage of system startup is verified, and a Root of Trust for Reporting to report the results [TCGRoT], [GloPlaRoT].

While there are many complex aspects of a Root of Trust, two aspects that are important in the case of attestation are:

- The first measurement computed by the Root of Trust for Measurement, and stored in the TPM's Root of Trust for Storage, is presumed to be correct.

- There must not be a way to reset the RTS without re-entering the RTM code.

The first measurement must be computed by code that is implicitly trusted; if that first measurement can be subverted, none of the remaining measurements can be trusted. (See [NIST-SP-800-155])

9. Informative References

[AIK-Enrollment]
Trusted Computing Group, "TCG Infrastructure Working GroupA CMC Profile for AIK Certificate Enrollment Version 1.0, Revision 7", March 2011, <https://trustedcomputinggroup.org/wp-content/uploads/IWG_CMC_Profile_Cert_Enrollment_v1_r7.pdf>.

[Canonical-Event-Log]
Trusted Computing Group, "DRAFT Canonical Event Log Format Version: 1.0, Revision: .12", October 2018.

[EFI-TPM] Trusted Computing Group, "TCG EFI Platform Specification for TPM Family 1.1 or 1.2, Specification Version 1.22, Revision 15", January 2014, <https://trustedcomputinggroup.org/wp-content/uploads/EFI-Protocol-Specification-rev13-160330final.pdf>.

[GloPlaRoT] GlobalPlatform Technology, "Root of Trust Definitions and Requirements Version 1.1", June 2018, <https://globalplatform.org/specs-library/globalplatform-root-of-trust-definitions-and-requirements/>.

[I-D.birkholz-rats-tuda] Fuchs, A., Birkholz, H., McDonald, I., and C. Bormann, "Time-Based Uni-Directional Attestation", draft-birkholz-rats-tuda-02 (work in progress), March 2020.

[I-D.birkholz-yang-swid] Birkholz, H., "Software Inventory YANG module based on Software Identifiers", draft-birkholz-yang-swid-02 (work in progress), October 2018.

[I-D.ietf-rats-architecture] Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote Attestation Procedures Architecture", draft-ietf-rats-architecture-04 (work in progress), May 2020.

[I-D.ietf-rats-eat] Mandyam, G., Lundblade, L., Ballesteros, M., and J. O'Donoghue, "The Entity Attestation Token (EAT)", draft-ietf-rats-eat-03 (work in progress), February 2020.

[I-D.ietf-rats-yang-tpm-charra] Birkholz, H., Eckel, M., Bhandari, S., Sulzen, B., Voit, E., Xia, L., Laffey, T., and G. Fedorkow, "A YANG Data Model for Challenge-Response-based Remote Attestation Procedures using TPMs", draft-ietf-rats-yang-tpm-charra-01 (work in progress), March 2020.

[I-D.ietf-sacm-coswid] Birkholz, H., Fitzgerald-McKay, J., Schmidt, C., and D. Waltermire, "Concise Software Identification Tags", draft-ietf-sacm-coswid-15 (work in progress), May 2020.

[I-D.richardson-rats-usecases]
Richardson, M., Wallace, C., and W. Pan, "Use cases for Remote Attestation common encodings", draft-richardson-rats-usecases-07 (work in progress), March 2020.

[I-D.voit-rats-trusted-path-routing]
Voit, E., "Trusted Path Routing using Remote Attestation", draft-voit-rats-trusted-path-routing-01 (work in progress), March 2020.

[IEEE-802-1AR]
Seaman, M., "802.1AR-2018 - IEEE Standard for Local and Metropolitan Area Networks - Secure Device Identity, IEEE Computer Society", August 2018.

[IEEE-802.1ae]
Seaman, M., "802.1AE MAC Security (MACsec)", 2018, <https://1.ieee802.org/security/802-1ae/>.

[IEEE-802.1x]
IEEE Computer Society, "802.1X-2020 - IEEE Standard for Local and Metropolitan Area Networks--Port-Based Network Access Control", February 2020, <https://standards.ieee.org/standard/802_1X-2020.html>.

[IMA] and , "Integrity Measurement Architecture", June 2019, <https://sourceforge.net/p/linux-ima/wiki/Home/>.

[LLDP] IEEE Computer Society, "802.1AB-2016 - IEEE Standard for Local and metropolitan area networks - Station and Media Access Control Connectivity Discovery", March 2016, <https://standards.ieee.org/standard/802_1AB-2016.html>.

[NetEq] Trusted Computing Group, "TCG Guidance for Securing Network Equipment", January 2018, <https://trustedcomputinggroup.org/wp-content/uploads/TCG_Guidance_for_Securing_NetEq_1_0r29.pdf>.

[NIST-IR-8060]
National Institute for Standards and Technology, "Guidelines for the Creation of Interoperable Software Identification (SWID) Tags", April 2016, <https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8060.pdf>.

[NIST-SP-800-155]
National Institute for Standards and Technology, "BIOS Integrity Measurement Guidelines (Draft)", December 2011, <https://csrc.nist.gov/csrc/media/publications/sp/800-155/draft/documents/draft-sp800-155_dec2011.pdf>.

[PC-Client-BIOS-TPM-1.2]
Trusted Computing Group, "TCG PC Client Specific Implementation Specification for Conventional BIOS, Specification Version 1.21 Errata, Revision 1.00", February 2012, <https://www.trustedcomputinggroup.org/wp-content/uploads/TCG_PCClientImplementation_1-21_1_00.pdf>.

[PC-Client-BIOS-TPM-2.0]
Trusted Computing Group, "PC Client Specific Platform Firmware Profile Specification Family "2.0", Level 00 Revision 1.04", June 2019, <https://trustedcomputinggroup.org/pc-client-specific-platform-firmware-profile-specification>.

[PC-Client-RIM]
Trusted Computing Group, "DRAFT: TCG PC Client Reference Integrity Manifest Specification, v.09", December 2019, <https://trustedcomputinggroup.org/xx>.

[Platform-Certificates]
Trusted Computing Group, "DRAFT: TCG Platform Attribute Credential Profile, Specification Version 1.0, Revision 15, 07 December 2017", December 2017.

[Platform-DevID-TPM-2.0]
Trusted Computing Group, "DRAFT: TPM Keys for Platform DevID for TPM2, Specification Version 0.7, Revision 0", October 2018.

[Platform-ID-TPM-1.2]
Trusted Computing Group, "TPM Keys for Platform Identity for TPM 1.2, Specification Version 1.0, Revision 3", August 2015, <https://trustedcomputinggroup.org/wp-content/uploads/TPM_Keys_for_Platform_Identity_v1_0_r3_Final.pdf>.

[Provisioning-TPM-2.0]
Trusted Computing Group, "TCG TPM v2.0 Provisioning Guidance", March 2015, <https://trustedcomputinggroup.org/wp-content/uploads/TCG-TPM-v2.0-Provisioning-Guidance-Published-v1r1.pdf>.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, Ed., "Extensible Authentication Protocol (EAP)", RFC 3748, DOI 10.17487/RFC3748, June 2004, <https://www.rfc-editor.org/info/rfc3748>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6813] Salowey, J. and S. Hanna, "The Network Endpoint Assessment (NEA) Asokan Attack Analysis", RFC 6813, DOI 10.17487/RFC6813, December 2012, <https://www.rfc-editor.org/info/rfc6813>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8572] Watsen, K., Farrer, I., and M. Abrahamsson, "Secure Zero Touch Provisioning (SZTP)", RFC 8572, DOI 10.17487/RFC8572, April 2019, <https://www.rfc-editor.org/info/rfc8572>.

[RIM] Trusted Computing Group, "DRAFT: TCG Reference Integrity Manifest Information Model", June 2019, <https://trustedcomputinggroup.org/wp-content/uploads/TCG_RIM_Model_v1-r13_2feb20.pdf>.

[SWID] The International Organization for Standardization/ International Electrotechnical Commission, "Information Technology Software Asset Management Part 2: Software Identification Tag, ISO/IEC 19770-2", October 2015, <https://www.iso.org/standard/65666.html>.

[TAP] Trusted Computing Group, "DRAFT: TCG Trusted Attestation Protocol (TAP) Information Model for TPM Families 1.2 and 2.0 and DICE Family 1.0, Version 1.0, Revision 0.29", October 2018.

[TCGRoT] Trusted Computing Group, "TCG Roots of Trust Specification", October 2018, <https://trustedcomputinggroup.org/wp-content/uploads/TCG_Roots_of_Trust_Specification_v0p20_PUBLIC_REVIEW.pdf>.

Authors' Addresses

Guy Fedorkow (editor)
Juniper Networks, Inc.
US

Email: gfedorkow@juniper.net

Eric Voit
Cisco Systems, Inc.
US

Email: evoit@cisco.com

Jessica Fitzgerald-McKay
National Security Agency
US

Email: jmfitz2@nsa.gov