

Remote ATtestation ProcedureS
Internet-Draft
Intended status: Standards Track
Expires: August 30, 2020

L. Xia
W. Pan
Huawei
February 27, 2020

Using NETCONF Pub/Sub Model for RATS Interaction Procedures
draft-xia-rats-pubsub-model-02

Abstract

This draft defines a new method of using the netconf pub/sub model in the RATS interaction procedure, to increase its flexibility, efficiency and scalability.

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 August 30, 2020.

Copyright Notice

Copyright (c) 2020 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.

Internet-Draft

RATS Push

February 2020

Table of Contents

1.	Introduction	2
2.	Conventions Used in This Document	3
3.	Pub/sub Model for Remote Attestation Procedure	4
3.1.	Solution Overview	4
3.2.	Remote Attestation Event Stream Definition	7
3.3.	Remote Attestation Subscription Definition	8
3.4.	Remote Attestation Selection Filters Definition	9
3.5.	Remote Attestation Subscription Parameters Handling	9
3.6.	Remote Attestation Notification Distribution	10
3.7.	Summary	10
4.	The YANG Module for Sub/pub Model Remote Attestation Procedures	13
4.1.	Tree Format	13
4.2.	Raw Format	13
5.	Security Considerations	13
6.	IANA Considerations	13
7.	References	13
7.1.	Normative References	13
7.2.	Informative References	14
	Acknowledgements	15
	Authors' Addresses	15

[1.](#) Introduction

Remote attestation is for acquiring the evidence about various integrity information from remote endpoints to assess its trustworthiness (aka, behave in the expected manner). These evidence should be about: system component identity, composition of system components, roots of trust, system component integrity, system component configuration, operational state and so on.

[\[I-D.richardson-rats-usecases\]](#) describes possible use cases which remote attestation are using for different industries, like: network devices, FIDO authentication for online transaction, Cryptographic Key Attestation for mobile devices, and so on.

[\[I-D.ietf-rats-architecture\]](#) lays a foundation of RATS architecture about the key RATS roles (i.e., Relying Party, Verifier, Attester and asseter) and the messages they exchange, as well as some key concepts. Based on it,

[\[I-D.birkholz-rats-reference-interaction-model\]](#) specifies a basic challenge-response-based interaction model for the remote attestation

procedure, which a complete remote attestation procedure is triggered by a challenge message originated from the verifier, and finished when the attester sends its response message back. This is a very generic interaction model with wide adoption. This document proposes an alternative interaction model for Remote attestation procedure, by

customizing the NETCONF pub/sub model [[RFC8639](#)][RFC8640][[RFC8641](#)]. YANG push [[RFC8641](#)] is basically an extensive NETCONF pub/sub model mainly for the YANG datastore. With the nature of asynchronous communication, the pub/sub model for remote attestation procedure is optimal for large-scale and loosely coupled distributed systems, especially for the network devices, which has the advantages as: loose coupling, scalability, time delivery sensitivity, supporting filtering capability, event-driven and so on. The pub/sub model can be used independently, or together with the challenge-response model to complement each other as a whole. Note that in which way these models are combined together are currently out of the scope of this draft.

In summary, by utilizing the pub/sub model in remote attestation procedure, the gained benefits are as below:

- o Flexibility: the verifier does not need to send the challenge message every time. The whole thing of the verifier is to subscribe a topic to the attester and then to anticipate the period or timely on-change notification from the attester about its integrity evidence.
- o Efficiency: once the verifier has subscribed its interested topics related with its triggering condition to the attester, it will get all the condition triggered notifications on time, which are the integrity related evidence for remote attestation in fact. It will ensure any integrity change/deviation of the remote endpoint to be detected with the minimum latency.
- o Scalability: it will save a lot of challenge messages by replacing with single subscription message for one topic stream, and decrease the total number of stateful connection between the verifier and attester, especially for a very large scale network. Thus, the scalability of the solution will increase.

This document is organized as follows. [Section 2](#) defines conventions

and acronyms used. [Section 3](#) discusses pub/sub model of remote attestation procedure.

[2.](#) Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [BCP 14](#) [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

Xia & Pan

Expires August 30, 2020

[Page 3]

Internet-Draft

RATS Push

February 2020

This document uses terminology defined in [[I-D.ietf-rats-architecture](#)] and [[I-D.birkholz-rats-reference-interaction-model](#)] for security related and RATS scoped terminology.

[3.](#) Pub/sub Model for Remote Attestation Procedure

[3.1.](#) Solution Overview

The following sequence diagram illustrates the reference remote attestation procedure by utilizing the NETCONF pub/sub model defined by this document.

Internet-Draft

RATS Push

February 2020

```
[Attester]                                                     [Verifier]
|
| <--Sub(nonce,authSecID,assertionSelection, event/period)
|
collectAssertions(assertionSelection)
| => assertions
|
signAttestationEvidence(authSecID, assertions, nonce)
| => signedAttestationEvidence
|
| signedAttestationEvidence ----->
|
| verifyAttestationEvidence(signedAttestationEvidence, refAssertions)
|                                     attestationResult <=
|
| .....
|
collectAssertions(assertionSelection)
| => assertions
|
signAttestationEvidence(authSecID, assertions, nonce)
| => signedAttestationEvidence
```

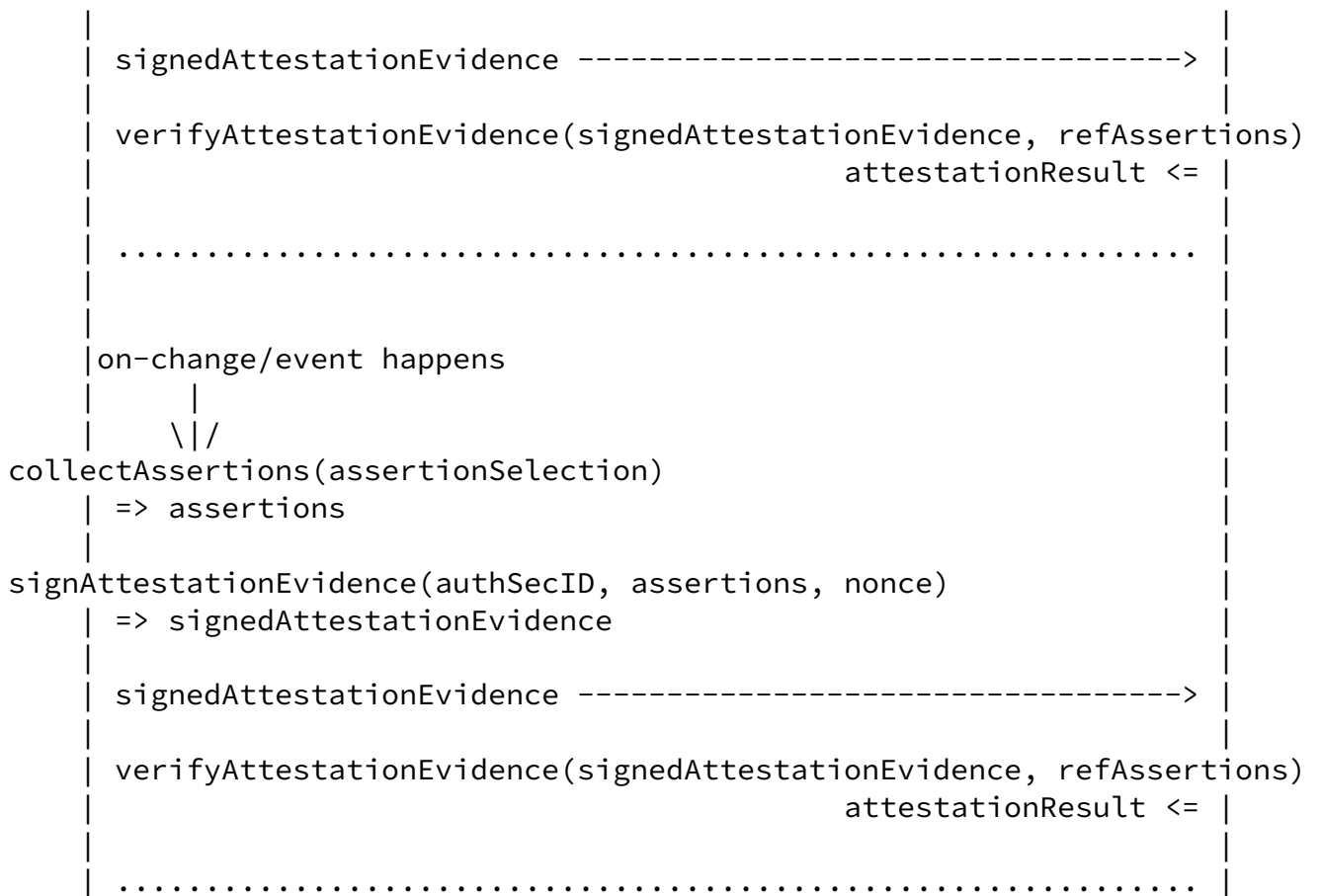


Figure 1: Pub/sub model of Remote Attestation

In short, the basic idea of pub/sub model for remote attestation is the verifier subscribes its interested event streams about the integrity evidence to the attester. The event streams can be in the YANG datastore, or not. After the subscription succeeds, the attester sends the subscribed integrity evidence back to the verifier. During subscription, the verifier may also specify how the attester returns the subscribed information, that is, the update trigger as periodic subscription or on-change subscription. And when the selection filters are applied to the subscription, only the information that pass the filter will be distributed out.

More detailed, the key steps of the remote attestation workflow with this model can be summarized as below (using the network devices as the example):

- o The verifier subscribes its interested event streams about the integrity evidence to the attester. More specifically:
 - * The event stream here refers to various integrity evidence information related to device trustworthiness, which can be in YANG datastore, or not. The basic event streams may include: software integrity information (including PCR values and system boot logs) of each layer of the trust chain recorded during device booting time; device identity certificates & Attestation Key certificate; operating system, application dynamic integrity information (e.g., IMA logs) and the device configuration information recorded during device running time.
 - * Periodic subscription is mainly used by the verifier for the general and non-critical information collection, which are not strictly time sensitive and aims for collecting the latest integrity evidence and checking the possible deviation. In contrary, on-change subscription is basically used to monitor the critical integrity evidence (e.g., integrity values and log files during device booting time, or integrity values of some key service processes). If these integrity values change, notifications are sent immediately.
 - * The selection filters may be applied to the subscription, so that only the event records that pass the filter will be distributed out. Some specific examples include: event records of a component (e.g., line card) in the composite device, the event records in a specific time period that includes a start time and an end time, and so on.
- o Consider how to send the existing parameters (i.e., nonce, hash signature algorithm, and specified TPM name, etc.) carried in the challenge message of the previous challenge-response model to the

attester through the subscription message of the new sub/pub model in advance, and the follow-up usage of them. A very important point is how to ensure that the nonce carried in every notification message is different, and both the attester and the verifier know the correct value in advance.

- o Both configuration subscription and dynamic subscription are considered. More specifically:

- * Configure subscription is for the important security event stream. For example, it enables the monitoring the important integrity information by using the on-change mode.
 - * Dynamic subscription is for the normal integrity information, that is, periodically receive those related information during NETCONF Session. The corresponding subscription RPC needs to be established dynamically. This way can reduce unnecessary NETCONF sessions.
- o In addition to the update trigger of on-change, the other possible update trigger may be certain pre-defined events according to [[I-D.bryskin-netconf-automation-yang](#)], that is: When these events occur, the specified integrity information is triggered to be sent, which is the relevant event stream plus optional selection filter. The events may include: device startup completion, device upgrade completion, specific attack event, active/standby switchover, line card insertion/removal/switchover, certificate life cycle event (expiration), etc.
 - o The attester notification delivery mechanisms thus vary as the above subscription mechanisms of verifier vary.

The following sections describes the above key steps one by one.

[3.2.](#) Remote Attestation Event Stream Definition

The event streams here refers to various integrity evidence information related to device trustworthiness. By definition, evidence is typically a set of claims about device's software and hardware platform. So, the remote attestation event stream is composed by the claims. For remote attestation, the basic event streams may generally include: system integrity information (including PCR values and system boot logs) of each layer of the trust chain recorded during device booting time, device credentials and their change, operating system and files integrity information (e.g., IMA logs) recorded during device running time, and so on.

The event streams are created and managed by the attester. And their

formal definition should be conformed to the information model definition like Attestation Evidence or others in [[I-D.birkholz-rats-information-model](#)], and the claim data model definition in [[I-D.ietf-rats-yang-tpm-charra](#)] with YANG data format, and [[I-D.ietf-rats-eat](#)] with COSE data format.

More specific, for current RATS claims YANG data model in [[I-D.ietf-rats-yang-tpm-charra](#)] , the following event streams may be defined if necessary:

- o the rats-support-structures datastore node, or its subtree nodes like: tpms, compute-nodes. All these nodes can be subscribed for pushing their values periodically or on-change;
- o For all the YANG RPCs, whether their output are the YANG datastore nodes or information stored in the device with other way, the event streams can be defined for all of them, such as: tpm12-attestation-response, tpm20-attestation-response, attestation-certificates and system-event-logs. If needed, the more fine-grained event stream can be defined for the substructure of the above, like: endorsement-cert or attestation-cer of the attestation-certificates, bios-event-logs or ima-event-logs of the system-event-logs.

[3.3](#). Remote Attestation Subscription Definition

NETCONF pub/sub model provides several subscription types in which appropriate one or more types are chosen and possibly used together to meet the service requirements.

Particularly, periodic subscription is mainly used by the verifier for the general and non-critical information collection, which are not strictly time sensitive and aims for collecting the latest integrity information and checking the possible deviation. In contrary, on-change subscription is basically used to monitor the critical integrity evidence (e.g., integrity values and log files during device booting time, or integrity values of some important files). If these integrity values change, notifications are sent immediately.

Besides, both configuration subscription and dynamic subscription are considered. In which, configure subscription is for the important security event stream as it lasts even the NETCONF session is closed. For example, it enables the monitoring of the status of important security event stream by using the on-change mode. On the other hand, dynamic subscription is for the general security event stream, that is, periodically receive those related information during

NETCONF Session. The corresponding subscription RPC needs to be established dynamically. This way can reduce unnecessary NETCONF sessions.

For the remote attestation event streams described in the previous section, some relatively critical and not frequently changed ones can be subscribed as the configuration and on-change subscription, so that the verifier can always receive them very timely. Some examples are: tpms, compute-nodes and attestation-certificates event streams. In contrary, some normal and frequently changed event streams can be the dynamic and/or periodic subscription, the verifier just want to receive and monitor them occasionally and reduce the processing. One example is ima-event-logs event stream.

Furthermore, certain pre-defined events according to [\[I-D.bryskin-netconf-automation-yang\]](#), can be the update trigger too, that is: When these events occur, the specified integrity information is triggered to be sent, which is the relevant event stream with optional selection filter. The events may include: device startup completion, device upgrade completion, specific attack event, active/standby switchover, line card insertion/removal/switchover, certificate life cycle event (expiration), etc.

[3.4.](#) Remote Attestation Selection Filters Definition

The selection filters may be applied to the subscription, so that only the event that pass the filter will be distributed out. Both the pub/sub and the YANG push selection filters can be considered.

A concrete example of selection filter is limiting the delivered event stream to those originated from a specific component with id ("xxxxxxxxxx") of a designated vendor ("xxx-vendor-device").

The other example is filtering the event records in a specific time period that has a start time and an end time.

[3.5.](#) Remote Attestation Subscription Parameters Handling

Most of the parameters carried in the subscription message are not changed during the remote attestation procedure, like: hash signature algorithm, specified TPM name and so on. Their main goal is to enable the dynamic negotiation with the attester about what information the verifier needs and how to construct them together. A very important point is how to ensure that the nonce carried in every notification message is different, and both the attester and the verifier know the correct value in advance. For this purpose, the

basic idea is to ensure that the nonce in two sides of the communication is synchronously changed, and the randomness of the

nonce is maintained. Specifically, there may be several ways to do it:

- o Verifier sends a seed with hash algorithm to the attester in the subscription message, and then perform the synchronization operation on both sides.
- o In fact, the nonce does not need to be random every time. As long as the receive endpoint (here for verifier) can identify duplicated packets, other means may be used. For example: The timestamp and increasing count.
- o The RATS TUDA mechanism [[I-D.birkholz-rats-tuda](#)] can also be used here to ensure the freshness of information.

[3.6.](#) Remote Attestation Notification Distribution

Basically, the remote attestation notification is the event stream in the YANG notification structure, and the event stream is defined above with the same YANG structure as the corresponding the YANG datastore node or RPC's output.

More details are to be added.

[3.7.](#) Summary

Based on the above discussion, this section gives some examples to illustrate the overall application of sub/pub model to remote attestation procedure.

Below is a configured subscription example with on-change update trigger, with specific contents as:

- o There are 3 integrity evidence related event streams as follows: pcr-trust-evidence, bios-log-trust-evidence and ima-log-trust-evidence. The subscribed one is pcr-trust-evidence.
- o The other parameters of the subscription include: pcr-list: {{1, 3, 7}}, tcg-hash-algo-id: TPM_ALG_SHA256, nonce-value: 0x564ac291,

TPM_ALG_ID-value: TPM_ALG_ECDSA, pub-key-id: 0x784a22bf, tpms: {"tpm1"}.

- o The selection filter is set as follows: a specific component with id ("xxxxxxxxxx") of a designated vendor ("xxx-vendor-device").

```
<edit-config>
  <subscriptions
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <subscription>
      <id>100</id>
      <stream>pcr-trust-evidence</stream>
      <stream-subtree-filter>
        <xxx-vendor-device
          xmlns="urn:xxx:params:xml:ns:yang:xxx-vendor-device ">
          <device-id>xxxxxxxxxx</device-id>
        </xxx-vendor-device>
      </stream-subtree-filter>
      <pcr-list>
        <pcr>
          <pcr-indices>1</pcr-indices>
          <pcr-indices>3</pcr-indices>
          <pcr-indices>7</pcr-indices>
          <hash-algo>
            <tcg-hash-algo-id>TPM_ALG_SHA256</tcg-hash-algo-id>
          </hash-algo>
        </pcr>
      </pcr-list>
      <nonce-value>0x564ac291</nonce-value>
      <TPM_ALG_ID-value>TPM_ALG_ECDSA</TPM_ALG_ID-value>
      <pub-key-id>0x784a22bf</pub-key-id>
      <tpms>
        <tpm-name>tpm1</tpm-name>
      </tpms>
      <yp:on-change>
        <yp:dampening-period>100</yp:dampening-period>
      </yp:on-change>
    </subscription>
```

```
</subscriptions>
</edit-config>
```

Figure 2: Configured On-change Subscription Message

Below is a dynamic subscription RPC example with periodic update trigger, with specific contents as:

- o There are 3 integrity evidence related event streams as follows: pcr-trust-evidence, bios-log-trust-evidence and ima-log-trust-evidence. The subscribed one is bios-log-trust-evidence.
- o The other parameters of the dynamic subscription include: tpms: {"tpm1"}, last-entry-value: 0xa34568baac79, log-type: bios, pcr-list: {{2, 4, 8}}, tcg-hash-algo-id: TPM_ALG_SHA256.

- o The selection filter is set as follows: a specific component with id ("xxxxxxxxxx") of a designated vendor ("xxx-vendor-device").
- o Subscription period: 500 centiseconds.

```
<rpc netconf:message-id="101"
  xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
  <establish-subscription
    xmlns="urn:ietf:params:xml:ns:yang:ietf-subscribed-notifications">
    <stream>bios-log-trust-evidence</stream>
    <stream-subtree-filter>
      <xxx-vendor-device
        xmlns="urn:xxx:params:xml:ns:yang:xxx-vendor-device ">
        <device-id>xxxxxxxxxx</device-id>
      </xxx-vendor-device>
    </stream-subtree-filter>
    <tpms>
      <tpm-name>tpm1</tpm-name>
    </tpms>
    <last-entry-value>0xa34568baac79</last-entry-value>
    <log-type>bios</log-type>
    <pcr-list>
      <pcr>
        <pcr-indices>2</pcr-indices>
        <pcr-indices>4</pcr-indices>
```

```

        <pcr-indices>8</pcr-indices>
        <hash-algo>
            <tcg-hash-algo-id>TPM_ALG_SHA256</tcg-hash-algo-id>
        </hash-algo>
    </pcr>
</pcr-list>
<yp:periodic>
    <yp:period>500</yp:period>
</yp:periodic>
</establish-subscription>
</rpc>

```

Figure 3: Dynamic Periodic Subscription Message

Below is a configured subscription RPC example with pre-defined events as the update trigger, with specific contents as:

- o There are 3 integrity evidence related event streams as follows: pcr-trust-evidence, bios-log-trust-evidence and ima-log-trust-evidence. The subscribed one is pcr-trust-evidence.
- o The other parameters of the subscription include: pcr-list: {{1, 3, 7}}, tcg-hash-algo-id: TPM_ALG_SHA256, nonce-value: 0x564ac291,

TPM_ALG_ID-value: TPM_ALG_ECDSA, pub-key-id: 0x784a22bf, tpms: {"tpm1"}.

- o The selection filter is set as follows: a specific component with id ("xxxxxxxxxx") of a designated vendor ("xxx-vendor-device").
- o The event which triggers the integrity evidence delivery is defined as: id: 1001, type: master-slave-switchover

NO FIGURE YET

Figure 4: Configured Event-triggered Subscription Message

[4.](#) The YANG Module for Sub/pub Model Remote Attestation Procedures

[4.1.](#) Tree Format

To be written.

[4.2.](#) Raw Format

To be written.

[5.](#) Security Considerations

To be written.

[6.](#) IANA Considerations

To be written, possibly.

[7.](#) References

[7.1.](#) 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>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in [RFC 2119](#) Key Words", [BCP 14](#), [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8639] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Subscription to YANG Notifications", [RFC 8639](#), DOI 10.17487/RFC8639, September 2019, <<https://www.rfc-editor.org/info/rfc8639>>.

Xia & Pan

Expires August 30, 2020

[Page 13]

Internet-Draft

RATS Push

February 2020

- [RFC8640] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Dynamic Subscription to YANG Events and Datastores over NETCONF", [RFC 8640](#), DOI 10.17487/RFC8640, September 2019, <<https://www.rfc-editor.org/info/rfc8640>>.
- [RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", [RFC 8641](#), DOI 10.17487/RFC8641, September 2019, <<https://www.rfc-editor.org/info/rfc8641>>.

[7.2.](#) Informative References

[I-D.birkholz-rats-information-model]

Birkholz, H. and M. Eckel, "An Information Model for Claims used in RATS", [draft-birkholz-rats-information-model-01](#) (work in progress), January 2020.

[I-D.birkholz-rats-reference-interaction-model]

Birkholz, H. and M. Eckel, "Reference Interaction Models for Remote Attestation Procedures", [draft-birkholz-rats-reference-interaction-model-02](#) (work in progress), January 2020.

[I-D.birkholz-rats-tuda]

Fuchs, A., Birkholz, H., McDonald, I., and C. Bormann, "Time-Based Uni-Directional Attestation", [draft-birkholz-rats-tuda-01](#) (work in progress), September 2019.

[I-D.bryskin-netconf-automation-yang]

Bryskin, I., Liu, X., Clemm, A., Birkholz, H., and T. Zhou, "Generalized Network Control Automation YANG Model", [draft-bryskin-netconf-automation-yang-03](#) (work in progress), July 2019.

[I-D.ietf-rats-architecture]

Birkholz, H., Thaler, D., Richardson, M., and N. Smith, "Remote Attestation Procedures Architecture", [draft-ietf-rats-architecture-01](#) (work in progress), February 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-00](#)
(work in progress), January 2020.

[I-D.richardson-rats-usecases]

Richardson, M., Wallace, C., and W. Pan, "Use cases for
Remote Attestation common encodings", [draft-richardson-
rats-usecases-06](#) (work in progress), November 2019.

Acknowledgements

Thanks to ...

Authors' Addresses

Liang Xia (Frank)
Huawei
101 Software Avenue, Yuhuatai District,
Nanjing, Jiangsu 210012
China

Email: frank.xialiang@huawei.com

Wei Pan
Huawei
101 Software Avenue, Yuhuatai District
Nanjing, Jiangsu 210012
China

Email: william.panwei@huawei.com