

RATS Working Group
Internet-Draft
Intended status: Standards Track
Expires: 28 October 2025

M. Richardson
Sandelman Software Works
26 April 2025

Proof of Position for Auditor managed Endorsements
draft-richardson-rats-pop-endorsement-00

Abstract

Some aspects of a device can not be intuited by the device itself. For instance, a router platform may have no way to know what color the case is, where in a cabinet it is located, or which electrical circuit it is connected to. This kind of information must be provided through an Endorsement: a statement from a third party.

These statements may require human auditors to inspect the device physically. But, which device is really in front of an auditor? This document describes a mechanism by which an auditor can make physical contact with a device and collect information to identify the device in a cryptographically strong manner.

This protocol is not designed to run over Internet Protocol cabling, but rather over mechanisms such as USB cables, or serial consoles.

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 28 October 2025.

Copyright Notice

Copyright (c) 2025 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
1.1. Overview of mechanism	3
2. Terminology	3
3. Protocol	4
3.1. Initial Handshake	4
3.2. Proof of Position	4
3.3. Collection of Endorsement Claims	5
3.4. Additional Commands	6
3.5. Generation of Endorsement	6
4. Alternatives to USB/serial cables	7
5. Privacy Considerations	8
6. Security Considerations	8
7. IANA Considerations	9
8. Acknowledgements	9
9. Changelog	9
10. References	9
10.1. Normative References	9
10.2. Informative References	10
Author's Address	10

1. Introduction

In the RATS Architecture [RFC9334], an important input to the Verifier function is the Endorsement. Endorsements [I-D.ietf-rats-endorsements] provide information that the Attester can not inherently know. This includes a statement that a particular (Attester) keypair belongs to a device of a particular type. Some of this information might be placed into device identity certificate (such as an IDevID [IDevID]), but many other kinds of claims would not belong.

For instance, the physical location or connectivity of a device would be subject to change. In some cases a GPS coordinate might make sense, but in other cases GPS might not be trustworthy, or might be inadequate. The physical location of a router, as being in "Building 4, Aisle 37, Cabinet 9, Rack Unit 2-3" would be a level of precision that GPS would be unable to provide. Other claims might involve

knowledge of the color of a device ("the red car"), or the connectivity of the device ("the device plugged into the blue cable"). A relative claim might also be relevant: The house on top of the person with Ruby Slippers.

In these cases an endorsement will need to be created, often by a human inspector/auditor that will have to physically visit the device and ascertain the state of the device. There are some challenges for such an auditor: they could be led astray by malicious intent to inspect the wrong device, or they could simply not locate the device they intended to audit. This results in an endorsement linked to the wrong device.

1.1. Overview of mechanism

The auditor is equipped with a portable device (e.g., a tablet computer) containing an endorsement signing key. This is the audit device. (This could be an actual key, or it could just be a secure/tamperproof container in which endorsements will be stored until a long-term/more-secure endorsement key can be employed)

The auditor finds the device in question and then collects whatever information is relevant. For instance, the location in whatever form makes sense for the endorsement. That might include taking pictures of the device in-situ, scans of serial numbers on the outside of the case, and which cables are connected to which physical ports.

The auditor then plugs a physical cable between their audit device and the device under audit. This cable is envisioned to be either a USB console "rollover" cable [rollover]. The audit device then initiates some commands over this cable that will result in a proof of the identity of connected device.

2. Terminology

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.

Audit Device: The device that is used to collect information that will go into endorsements.

Device Under Audit: Abbreviated to DUA. The device for which endorsements are being made.

Auditor: The human who inspects the Device Under Audit.

3. Protocol

It is assumed that the console port has been designed for use by humans. This protocol is designed to interact as if it's a human. Note that more sophisticated mechanisms such as SLIP or PPP would be more vulnerable to spoofing.

3.1. Initial Handshake

The audit device sends carriage returns (octet 13) until it sees a response with a colon (":") in it. This is usually a "login:" or "username:" prompt of some kind. The audit device then sends the word "endorsementaudit" with a carriage return. This represents a user login, and for some systems this can be set as a real login with limited permissions. It could run a special audit shell that only can perform system audits. On other systems, this might just an unprivileged account with the normal prompts and commands.

The handshake is over when the word "endorsement" is seen.

3.2. Proof of Position

(Proof of Position is probably a silly name that ought to be replaced)

All commands are prefixed with "rfcXXXX" (with a trailing space, where XXXX is the number of this document). A second word indicates the command. This allows the device under audit to collect all operations under a single command or command sub-system.

The audit device then sends the word/command "position-proof", followed by a 33 byte nonce, base64URL encoded into a 45 byte string. (33 bytes are recommended because being divisible by three, they encode evenly in base64, leaving no trailing =)

The device under audit then responds with a CBOR format EAT object, encoded in base64URL, and wrapped in the strings "--- BEGIN COSE OBJECT ---" and "--- END COSE OBJECT ---" {XXX: or should we use CBOR Diagnostic Format?}

The EAT payload should be constructed as follows. Shown are a few attributes that would make sense to include. The above provide nonce becomes the eat_nonce.

```

{
  / eat_nonce /      10: h'948f8860d13a463e8e',
  / ueid /           256: h'0198f50a4ff6c05861c8860d13a638ea',
  / oemid /          258: h'894823', / IEEE OUI format OEM ID /
  / hwmodel /        259: h'549dcecc8b987c737b44e40f7c635ce8'
                        / Hash of chip model name /,
  / hwversion /      260: ["1.3.4", 1], / Multipartnumeric version /
  / swname /          271: "Acme OS",
  / swversion /       272: ["3.5.5", 1],
}

```

The EAT payload is signed using the device under audit's Attestation Key. (A TPM's Endorsement Key can not sign things) The hash of the Attestation Key is provided in the unprotected headers.

```

61( 18( [
  h'A10126', / protected headers /
  {
    / x5t / 34: [16, h'1234456781234567812344567812345678'],
  }, / unprotected headers /
  h'A20B46024A6B0978DE0A49000102030405060708', / payload /
  h'9B9B2F5E470000F6A20C8A4157B5763FC45BE759
    9A5334028517768C21AFFB845A56AB557E0C8973
    A07417391243A79C478562D285612E292C622162
    AB233787' / signature /
] ) )

```

x5t is from [RFC9360]. The hash algorithm SHOULD be SHA256, or newer. (Example to be updated.)

3.3. Collection of Endorsement Claims

The audit device then validates the received EAT object from the device. The audit device locates the public part of the device's Attestation Key. This will in most cases be part of the "work order" that the auditor has been provided, but in some cases, the auditor will have to collect it from the device.

For instance, a device might have been replaced since the audit was requested, or there might be additional devices that the auditor thinks might be relevant. An example might be when a switching device has many cables connected into an adjacent optical media converter that puts multiple signals through a single multi-frequency optical cable. Such a layer of indirection might affect the audit's ability to indicate which port is physically connected to which cable.

Some key physical information that an auditor might need to collect is which fibre patches are connected to which ports of the switch. This information would be ideally collected by scanning (and checking) labels on the fibre patches.

3.4. Additional Commands

Some additional commands can be provided by the device under audit:

"attestation-key": This command returns the public certificate for the device's Attestation Key in [RFC7468] Certificate format.

"port-flash": This command takes an interface number/name (e.g., "GigabitEthernet 3/014"), and causes the LEDs adjacent to a particular port to flash in a distinct pattern. This is to help identity which physical port is which.

"port-down": This command takes an interface number/name (as above), and causes the switch to mark the physical port as down, but not turn off the laser. Traffic SHOULD be re-routed as if the fibre has failed. This is a disruptive test! The auditor may then physically unplug the fibre as part of an audit of the fibre paths. This command might not perform the action immediately, but could signal to the network operations center that such a test is desired, and allow them to approve it.

"port-up": Indicates the end of an path-audit started by "port-down".

"endorsements": This command is followed by a base64URL encoded CBOR Sequence of Endorsements, wrapped in the same BEGIN COSE header and footer as before. To guard against failure during, the device under audit SHOULD time out this command after 120s if no END COSE OBJECT framing is seen. This command allows the auditor to load any resulting endorsements directly into the device to be passed up along with Evidence to a Verifier.

"exit": This command indicates that the audit is over, and the device under audit can return the console interface to the normal state.

3.5. Generation of Endorsement

The auditor, having collected one or more proofs, then transmits them to the endorsement agency. This may be via physical transfer, secured email, or some secured online API.

The endorsement agency then needs to do the following:

1. locate the Endorsement Certificate, if it was not collected from the device.
2. validate that the provided EAT is signed by the associated Endorsement Key.
3. validate that the claims in the EAT are consistent with that which is expected from the device.
4. validate the format of the additional information collected by the auditor. Location, type and number of connections, (colour of device even).

From this, one or more Endorsements are then created and signed by the endorsing agency.

In some cases the auditor might be entirely self-contained, producing the endorsements directly on their audit device. In that case, they would use the "endorsements" to load the resulting Endorsements directly into the device.

4. Alternatives to USB/serial cables

There are a number of cases where a USB or serial console cable might be unavailable, or it's might be undesirable. Many smaller IoT devices, home routers and consumer items do not have any kind of console available. The console access might require removal of the case, which might be impossible to do while the device is operational, or while the device is physically installed. Console access might provide access to a privileged prompt; that access might either be unsafe to give to the auditor, or might require a password to access that should not be shared.

One alternative is to use Ethernet. Most routing platforms have many ethernet ports, and usually have at least one empty ethernet port. It is also common for there to be a dedicated copper ethernet port for management even on routing platforms that otherwise only have fiber-optic ports running at multiple hundred gigabits. The use of ethernet has problems because ethernet can easily be switched; transmitting the signal to another device elsewhere. This is often the whole point of the routing platform: to switch traffic elsewhere. This would result in a false audit if the auditor is diverted.

The LLDP protocol [LLDP] is a one-hop protocol which is usually not forwarded. It can do things like tell a connected device which port of client device is connected to. While trustworthy devices will not forward LLDP frames, an untrustworthy device that has been programmed to participate in a subterfuge might well forward frames. It might

be possible to construct a challenge/response system that has the auditor plug cables in some non-deterministic order in order to defeat the subterfuge.

This process would probably need to be augmented with some other forms of feedback; perhaps flashing of status LEDs on the device in a pattern.

Note: the console/USB cable could also be redirected to another host!

5. Privacy Considerations

The ability to walk up to a device and interrogate it as to its identity is potentially privacy violating if the device is associated with a person. This would include all kinds of small devices: phones, laptops, electric bicycles, automobiles,

One countermeasure is that the device needs to be put into an audit mode before it can be interrogated. This is reasonable for some devices and some audit processes, but for other processes, the need to do random audits may countermand this need. For devices such as large routing platforms, they are often located in data centers with multiple layers of physical access control: locked buildings, locked machine rooms, and locked cabinets. For such devices, there are perhaps few privacy concerns, and the auditor needs credentials in order to access the device at all.

6. Security Considerations

There are three concerns with this protocol.

The first is the potential for unauthorized people to collect information about devices to which they have no authority to interrogate. In industrial settings, this is mitigated by physical access controls. In those settings the ability to identify devices which may be physically misbehaving or are damaged and connect them to their digital identity significantly outweighs any concerns about device identity.

In consumer and retail settings, the device SHOULD not respond to this protocol unless an operator/owner has put the device into device identification mode.

The second concern with this protocol is that it might be spoofed or confuzzled. The auditor could be misled as to whether the device in front of them is really the device that is responding to their queries. The auditor SHOULD have the device identity with them already as part of the work order. They should therefore not be

mislead as to which device they intend to audit, the issue is that it might not be the device that is physically in front of them. The device might be a mock-up designed to look right, but really it is wired secretly to the real device which is elsewhere, and is differently configured. This attack is called the "Sock-Puppet" attack.

Such attacks require physical examination to detect. Some attacks may be mitigated through careful use of the "port-flash" commands to cause signals visible to the auditor that would ideally be difficult to fake. Efforts this way are the subject of further work.

The third concern with this protocol is that it might open up the device to attacks via this console port. The Initial Handshake Section 3.1 mechanism is designed so that it can be easily implemented by typical router and operating system login mechanisms. A very limited account would be created, or even a mode within the login mechanism itself, and so no additional inquiries would be possible. Some operators prefer to never have a login process on the console/craft ports of their devices. This is usually done so that maintenance people do not need to have passwords that can then be re-used over a network, weakening the security of the device. They depend upon physical security for the console ports to provide security. Such operators might wish to rethink this policy for devices which will be subject to audit.

7. IANA Considerations

IANA is asked to allocate a CBOR Tag for this object. Details TBD.

8. Acknowledgements

Your name here.

9. Changelog

10. References

10.1. Normative References

[I-D.ietf-rats-endorsements]

Thaler, D., Birkholz, H., and T. Fossati, "RATS Endorsements", Work in Progress, Internet-Draft, draft-ietf-rats-endorsements-06, 3 March 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-rats-endorsements-06>>.

- [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>>.
- [RFC7468] Josefsson, S. and S. Leonard, "Textual Encodings of PKIX, PKCS, and CMS Structures", RFC 7468, DOI 10.17487/RFC7468, April 2015, <<https://www.rfc-editor.org/info/rfc7468>>.
- [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>>.
- [RFC9334] Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote ATtestation procedureS (RATS) Architecture", RFC 9334, DOI 10.17487/RFC9334, January 2023, <<https://www.rfc-editor.org/info/rfc9334>>.
- [RFC9360] Schaad, J., "CBOR Object Signing and Encryption (COSE): Header Parameters for Carrying and Referencing X.509 Certificates", RFC 9360, DOI 10.17487/RFC9360, February 2023, <<https://www.rfc-editor.org/info/rfc9360>>.

10.2. Informative References

- [IDevID] IEEE Standard, "IEEE 802.1AR Secure Device Identifier", 2018, <<https://1.ieee802.org/security/802-1ar/1>>.
- [LLDP] IEEE Standard, "802.1AB-REV - Station and Media Access Control Connectivity Discovery", 19 June 2009, <<https://www.ieee802.org/1/pages/802.1AB-rev.html>>.
- [rollover] Wikipedia, "Console Rollover Cable", 26 April 2025, <https://en.wikipedia.org/wiki/Rollover_cable>.

Author's Address

Michael Richardson
Sandelman Software Works
Email: mcr+ietf@sandelman.ca