Workgroup: Remote ATtestation ProcedureS
Internet-Draft:
draft-frost-rats-eat-collection-01
Published: 27 June 2022
Intended Status: Standards Track
Expires: 29 December 2022

Authors: S. Frost
Arm

Entity Attestation Token (EAT) Collection Type

Abstract

The default top-level definitions for an EAT [I-D.ietf-rats-eat] assume a hierarchy involving a leading signer within the Attester. Some token use cases do not match that model. This specification defines an extension to EAT allowing the top-level of the token to consist of a collection of otherwise defined tokens.

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-frost-rats-eat-collection/.

Discussion of this document takes place on the Remote ATtestation ProcedureS Working Group mailing list (mailto:rats@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/rats/.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

An Attestation Token conforming to EAT [I-D.ietf-rats-eat] has a default top level definition for a token to be constructed principally as a claim set within a CBOR Web Token (CWT) [RFC8392] with the associated COSE envelope [RFC8152] providing at least integrity and authentication. An equivalent JSON encoding for a JWT [RFC7519] in a JWS envelope [RFC7515] is supported as an alternative at the top-level definition. The top level token can be augmented with related claims in a Detached Bundle (DEB).

For the use case of transmitting a claim set through a secure channel, the top-level definition can be extended to use an Unprotected CWT Claim Set (UCCS) [I-D.ietf-rats-uccs].

This document outlines an additional top-level extension for which neither of the above top level definitions match exactly: the attestation token consists of a collection of objects, each with their own integrity and some internally defined relationship through which the integrity of the whole collection can be determined. i.e. there is no top-level signer for the set. The objects may all share the same logical hierarchy in a device or have a hierarchy which is internally defined within the object set.

2. Design Considerations / Use Cases

Take a device with an attestation system consisting of a platform claim set and a workload claim set, each controlled by different components and with an underlying hardware Root of Trust. The two claim sets are delivered together to make up the overall attestation token. Depending upon the implementation and deployment use case, the signing system can either be entirely centric to the platform RoT or can have separate signers for the two claim sets. In either case, a cryptographic binding is established between the two parts of the token.

A specific manifestation of such a device is one incorporating the Arm Confidential Compute Architecture (CCA) attestation token [Arm-CCA]. In trying to prepare the attestation token using EAT, there were no issues constructing the claim sets or incorporating them into individual CWTs where appropriate. However, in trying to design an 'envelope structure' to convey the two parts as a single report it was found that maintaining EAT compatibility would require very different shaped compound tokens for different models, for example one based on a submod arrangement and another based on a DEB, though with different 'leading' objects. This would create extra code and explanation in areas where keeping things simple is desirable. There was an alternative approach considered, which stays close to existing thinking on EAT, which would be to create the wrapper from the UCCS EAT extension containing only submods for the respective components. This however stretches the current use case for UCCS beyond its existing description. The RATS WG approach of separating UCCS from the core EAT specification to be an extension also encourages proposing this further extension.

To support the CCA use case, it is also relevant to consider current attestation technologies which are based on certificate chains (e.g. SPDM, DICE, several key attestation systems). Here also are multiple objects with their own integrity and an internally defined relationship. If attempting to move such a technology to the EAT world, the same challenges apply.

3. Token Collection

The proposed extension for the top-level definition is to add a 'Token Collection' type. The contents of the type are a map of CWTs (JWTs). The DEB top-level entry for EAT is included for completeness, and the UCCS extension can also be embraced, though the use cases for these have not been explored. The identification of collection members and the intra collection integrity mechanism is considered usage specific. A verifier will be expected to extract each of the members of the collection and check their validity both individually and as a set.

A map was chosen rather than an unbounded array to give the opportunity to add identifying map tags to each entry. The interpretation of the tags will be usage specific, but may correspond to registered identities of specific token types. To assist a verifier correlate the expected contents a profile entry can be added as the 'profile-label' identity in the map.

See Appendix A for a CDDL [RFC8610] description of the proposed extension.

While most of the use cases for collections are for scenarios where there will be at least two entries in a collection, the CDDL allows for >= 1 entries in a collection to allow for the scenario where only one entry is currently available even though the normal set is larger.

4. Security Considerations

A verifier for an attestation token must apply a verification process for the full set of entries contained within the Token Collection. This process will be custom to the relevant profile for the Token Collection and take into account any individual verification per entry and/or verification for the objects considered collectively, including the intra token integrity scheme. As there is no overall signature for the Collection, protection against malicious modification must be contained within the entries. It is expected that there exists a cryptographic binding between entries, this can for example be one to many or one to one in a (chain) series. Depending upon the use case and associated threat model, the freshness of entries may need extra consideration.

5. IANA Considerations

In the registry [IANA.cbor-tags], IANA is requested to allocate the tag in Table 1 from the FCFS space, with the present document as the specification reference.

Tag	Data Item	Semantics
TBD399	map	EAT Collection RFCthis

Table 1: EAT Collection

6. References

6.1. Normative References

[I-D.ietf-rats-eat] Lundblade, L., Mandyam, G., and J. O'Donoghue, "The Entity Attestation Token (EAT)", Work in Progress, Internet-Draft, draft-ietf-rats-eat-13, 20 May 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-eat-13>.

[IANA.cbor-tags] IANA, "Concise Binary Object Representation (CBOR) Tags", <https://www.iana.org/assignments/cbor-tags>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/rfc/rfc8610>.

6.2. Informative References

[Arm-CCA] Arm Ltd, "Confidential Compute Architecture", n.d..

[I-D.ietf-rats-uccs] Birkholz, H., O'Donoghue, J., Cam-Winget, N., and C. Bormann, "A CBOR Tag for Unprotected CWT Claims Sets", Work in Progress, Internet-Draft, draft-ietf-rats-uccs-02, 12 January 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-uccs-02>.

[RFC7515] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Signature (JWS)", RFC 7515, DOI 10.17487/RFC7515, May 2015, <https://www.rfc-editor.org/rfc/rfc7515>.

[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/rfc/rfc7519>.

[RFC8152] Schaad, J., "CBOR Object Signing and Encryption (COSE)", RFC 8152, DOI 10.17487/RFC8152, July 2017, <https://www.rfc-editor.org/rfc/rfc8152>.

[RFC8392] Jones, M., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "CBOR Web Token (CWT)", RFC 8392, DOI 10.17487/RFC8392, May 2018, <https://www.rfc-editor.org/rfc/rfc8392>.

Appendix A. CDDL

```
$$EAT-CBOR-Tagged-Token /= Tagged-Collection
$$EAT-CBOR-Untagged-Token /= TL-Collection

Tagged-Collection =  #6.TBD399(TL-Collection)

; Note that although the common use cases for collections are for at least two entries in a c
; the CDDL below allows for >= 1 entry to allow the scenario where only one entry is currentl
; though the normal set is larger
TL-Collection = {
    ? eat-collection-identifier,
    + cwt-collection-entries // jwt-collection-entries // DEB-collection-entries
}

eat-collection-identifier = (
    profile-label => general-uri / general-oid
)

cwt-collection-entries = (
    collection-entry-label => CWT-Messages
)

jwt-collection-entries = (
    collection-entry-label => JWT-Messages
)

DEB-collection-entries = (
    collection-entry-label => DEB-Messages
)

collection-entry-label = JC<text, int>
```

Acknowledgments

Thomas Fossati and Yogesh Deshpande provided insightful comments and review for this proposal.

Author's Address

Simon Frost
Arm

Email: Simon.Frost@arm.com