Workgroup: RATS Working Group
Internet-Draft: draft-ietf-rats-uccs-09
Published: 4 March 2024
Intended Status: Standards Track
Expires: 5 September 2024
Authors: H. Birkholz J. O'Donoghue
Fraunhofer SIT Qualcomm Technologies Inc.
N. Cam-Winget C. Bormann
Cisco Systems Universität Bremen TZI

A CBOR Tag for Unprotected CWT Claims Sets

Abstract

When transported over secure channels, CBOR Web Token (CWT, RFC 8392) Claims Sets may not need the protection afforded by wrapping them into COSE, as is required for a true CWT. This specification defines a CBOR tag for such unprotected CWT Claims Sets (UCCS) and discusses conditions for its proper use.

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-ietf-rats-uccs/.

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

Source for this draft and an issue tracker can be found at https://github.com/ietf-rats-wg/draft-ietf-rats-uccs.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

A CBOR Web Token (CWT) as specified by [RFC8392] is always wrapped in a CBOR Object Signing and Encryption (COSE, [RFC9052]) envelope. COSE provides -- amongst other things -- end-to-end data origin authentication and integrity protection employed by RFC 8392 as well as optional encryption for CWTs. Under the right circumstances (Section 3), though, a signature providing proof for authenticity and integrity can be provided through the transfer protocol and thus omitted from the information in a CWT without compromising the intended goal of authenticity and integrity. In other words, if communicating parties have a pre-existing security association, they can reuse it to provide authenticity and integrity for their messages, enabling the basic principle of using resources parsimoniously. Specifically, if a mutually secured channel is established between two remote peers, and if that secure channel provides the required properties (as discussed below), it is possible to omit the protection provided by COSE, creating a use case for unprotected CWT Claims Sets. Similarly, if there is one-way authentication, the party that did not authenticate may be in a position to send authentication information through this channel that allows the already authenticated party to authenticate the other party; this effectively turns the channel into a mutually secured channel.

This specification allocates a CBOR tag to mark Unprotected CWT Claims Sets (UCCS) as such and discusses conditions for its proper use in the scope of Remote Attestation Procedures (RATS [RFC9334]) for the conveyance of RATS Conceptual Messages.

This specification does not change [RFC8392]: A true CWT does not make use of the tag allocated here; the UCCS tag is an alternative to using COSE protection and a CWT tag. Consequently, within the well-defined scope of a secure channel, it can be acceptable and economic to use the contents of a CWT without its COSE container and tag it with a UCCS CBOR tag for further processing within that scope -- or to use the contents of a UCCS CBOR tag for building a CWT to be signed by some entity that can vouch for those contents.

1.1. Terminology

The term Claim is used as in [RFC7519].

The terms Claim Key, Claim Value, and CWT Claims Set are used as in [RFC8392].

The terms Attester, Attesting Environment, Evidence, Relying Party and Verifier are used as in [RFC9334].

UCCS: Unprotected CWT Claims Set(s); CBOR map(s) of Claims as defined by the CWT Claims Registry that are composed of pairs of Claim Keys and Claim Values.

Secure Channel: [NIST-SP800-90Ar1] defines a Secure Channel as follows:

> "A path for transferring data between two entities or components that ensures confidentiality, integrity and replay protection, as well as mutual authentication between the entities or components. The secure channel may be provided using approved cryptographic, physical or procedural methods, or a combination thereof"

For the purposes of the present document, we focus on a protected communication channel used for conveyance that can ensure the same qualities as CWT without having the COSE protection available: mutual authentication, integrity protection, confidentiality. (Replay protection can be added by including a nonce claim such as Nonce (claim 10 [IANA.cwt]).) Examples include conveyance via PCIe (Peripheral Component Interconnect Express) IDE (Integrity and Data Encryption), or a TLS tunnel.

All terms referenced or defined in this section are capitalized in the remainder of 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.

2. Deployment and Usage of UCCS

Usage scenarios involving the conveyance of Claims, in particular RATS, require a standardized data definition and encoding format that can be transferred and transported using different communication channels. As these are Claims, the Claims sets defined in [RFC8392] are a suitable format. However, the way these Claims are secured depends on the deployment, the security capabilities of the device, as well as their software stack. For example, a Claim may be securely stored and conveyed using a device's Trusted Execution Environment (TEE, see [RFC9397]) or a Trusted Platform Module (TPM, see [TPM2]). Especially in some resource constrained environments, the same process that provides the secure communication transport is also the delegate to compose the Claim to be conveyed. Whether it is a transfer or transport, a Secure Channel is presumed to be used for conveying such UCCS. The following sections elaborate on Secure Channel characteristics in general and further describe RATS usage scenarios and corresponding requirements for UCCS deployment.

3. Characteristics of a Secure Channel

A Secure Channel for the conveyance of UCCS needs to provide the security properties that would otherwise be provided by COSE for a CWT. In this regard, UCCS is similar in security considerations to JWTs [RFC8725] using the algorithm "none". RFC 8725 states:

> [...] if a JWT is cryptographically protected end-to-end by a transport layer, such as TLS using cryptographically current algorithms, there may be no need to apply another layer of cryptographic protections to the JWT. In such cases, the use of the "none" algorithm can be perfectly acceptable.

The security considerations discussed, e.g., in Sections 2.1, 3.1, and 3.2 of [RFC8725] apply in an analogous way to the use of UCCS as elaborated on in this document.

Secure Channels are often set up in a handshake protocol that mutually derives a session key, where the handshake protocol establishes the (identity and thus) authenticity of one or both ends of the communication. The session key can then be used to provide confidentiality and integrity of the transfer of information inside the Secure Channel. (Where the handshake did not provide a mutually secure channel, further authentication information can be conveyed by the party not yet authenticated, leading to a mutually secured channel.) A well-known example of a such a Secure Channel setup protocol is the TLS [RFC8446] handshake; the TLS record protocol can then be used for secure conveyance.

As UCCS were initially created for use in RATS Secure Channels, the following section provides a discussion of their use in these channels. Where other environments are intended to be used to convey UCCS, similar considerations need to be documented before UCCS can be used.

4. UCCS in RATS Conceptual Message Conveyance

This section describes a detailed usage scenario for UCCS in the context of RATS in conjunction with its attendant security requirements. The use of UCCS tag CPA601 outside of the RATS context **MUST** come with additional instruction leaflets and security considerations.

For the purposes of this section, any RATS role can be the sender or the receiver of the UCCS.

Secure Channels can be transient in nature. For the purposes of this specification, the mechanisms used to establish a Secure Channel are out of scope.

In the scope of RATS Claims, the receiver **MUST** authenticate the sender as part of the establishment of the Secure Channel. Furthermore, the channel **MUST** provide integrity of the communication between the communicating RATS roles. For data confidentiality [RFC4949], the receiving side **MUST** be authenticated as well; this is achieved if the sender and receiver mutually authenticate when establishing the Secure Channel. The quality of the receiver's authentication and authorization will influence whether the sender can disclose the UCCS.

The extent to which a Secure Channel can provide assurances that UCCS originate from a trustworthy Attesting Environment depends on the characteristics of both the cryptographic mechanisms used to establish the channel and the characteristics of the Attesting Environment itself. The assurance provided to a Relying Party depends on the authenticity and integrity properties of the Secure Channel used for conveying the UCCS to it.

Ultimately, it is up to the receiver's policy to determine whether to accept a UCCS from the sender and to the type of Secure Channel it must negotiate. While the security considerations of the cryptographic algorithms used are similar to COSE, the considerations of the Secure Channel should also adhere to the policy configured at each of end of the Secure Channel. However, the policy controls and definitions are out of scope for this document.

Where an Attesting Environment serves as an endpoint of a Secure Channel used to convey a UCCS, the security assurance required of that Attesting Environment by a Relying Party generally calls for the Attesting Environment to be implemented using techniques designed to provide enhanced protection from an attacker wishing to tamper with or forge UCCS originating from that Attesting Environment. A possible approach might be to implement the Attesting Environment in a hardened environment such as a TEE [RFC9397] or a TPM [TPM2].

When UCCS emerge from the Secure Channel and into the receiver, the security properties of the secure channel no longer protect the UCCS, which now are subject to the same security properties as any other unprotected data in the Verifier environment. If the receiver subsequently forwards UCCS, they are treated as though they originated within the receiver.

The Secure Channel context does not govern fully formed CWTs in the same way it governs UCCS. As with EATs nested in other EATs (Section 4.2.18.3 (Nested Tokens) of [I-D.ietf-rats-eat]), the Secure Channel does not endorse fully formed CWTs transferred through it. Effectively, the COSE envelope of a CWT (or a nested EAT) shields the CWT Claims Set from the endorsement of the secure channel. (Note that EAT might add a nested UCCS Claim, and this statement does not apply to UCCS nested into UCCS, only to fully formed CWTs.)

5. Considerations for Using UCCS in Other RATS Contexts

This section discusses two additional usage scenarios for UCCS in the context of RATS.

5.1. Delegated Attestation

Another usage scenario is that of a sub-Attester that has no signing keys (for example, to keep the implementation complexity to a minimum) and has a Secure Channel, such as local inter-process communication, to interact with a lead Attester (see Composite Device, Section 3.3 of [RFC9334]). The sub-Attester produces a UCCS with the required CWT Claims Set and sends the UCCS through the Secure Channel to the lead Attester. The lead Attester then computes a cryptographic hash of the UCCS and protects that hash using its signing key for Evidence, for example, using a Detached-Submodule-Digest or Detached EAT Bundle (Section 5 of [I-D.ietf-rats-eat]).

5.2. Privacy Preservation

A Secure Channel which preserves the privacy of the Attester may provide security properties equivalent to COSE, but only inside the life-span of the session established. In general, when a privacy preserving Secure Channel is employed for conveying a conceptual message the receiver cannot correlate the message with the senders of other received UCCS messages.

An Attester must consider whether any UCCS it returns over a privacy preserving Secure Channel compromises the privacy in unacceptable ways. As an example, the use of the EAT UEID Claim Section 4.2.1 of [I-D.ietf-rats-eat] in UCCS over a privacy preserving Secure Channel allows a Verifier to correlate UCCS from a single Attesting Environment across many Secure Channel sessions. This may be acceptable in some use-cases (e.g., if the Attesting Environment is a physical sensor in a factory) and unacceptable in others (e.g., if the Attesting Environment is a user device belonging to a child).

6. IANA Considerations

In the CBOR Tags registry [IANA.cbor-tags] as defined in Section 9.2 of [RFC8949], IANA is requested to allocate the tag in Table 1 from the Specification Required space (1+2 size), with the present document as the specification reference.

Tag	Data Item	Semantics
CPA601	map (Claims-Set as per Appendix A of [RFCthis])	Unprotected CWT Claims Set [RFCthis]

Table 1: Values for Tags

RFC-Editor: This document uses the CPA (code point allocation) convention described in [I-D.bormann-cbor-draft-numbers]. For each usage of the term "CPA", please remove the prefix "CPA" from the indicated value and replace the residue with the value assigned by IANA; perform an analogous substitution for all other occurrences of the prefix "CPA" in the document. Finally, please remove this note.

7. Security Considerations

The security considerations of [RFC8949] apply. The security considerations of [RFC8392] need to be applied analogously, replacing the function of COSE with that of the Secure Channel.

Section 3 discusses security considerations for Secure Channels, in which UCCS might be used. This document provides the CBOR tag definition for UCCS and a discussion on security consideration for the use of UCCS in RATS. Uses of UCCS outside the scope of RATS are not covered by this document. The UCCS specification -- and the use of the UCCS CBOR tag, correspondingly -- is not intended for use in a scope where a scope-specific security consideration discussion has not been conducted, vetted and approved for that use. In order to be able to use the UCCS CBOR tag in another such scope, the secure channel and/or the application protocol (e.g., TLS and the protocol identified by ALPN) **MUST** specify the roles of the endpoints in a fashion that the security properties of conveying UCCS via a Secure Channel between the roles are well-defined.

7.1. General Considerations

Implementations of Secure Channels are often separate from the application logic that has security requirements on them. Similar security considerations to those described in [RFC9052] for obtaining the required levels of assurance include:

* Implementations need to provide sufficient protection for private or secret key material used to establish or protect the Secure Channel.

* Using a key for more than one algorithm can leak information about the key and is not recommended.

* An algorithm used to establish or protect the Secure Channel may have limits on the number of times that a key can be used without leaking information about the key.

* Evidence in a UCCS conveyed in a Secure Channel generally cannot be used to support trust in the credentials that were used to establish that secure channel, as this would create a circular dependency.

The Verifier needs to ensure that the management of key material used to establish or protect the Secure Channel is acceptable. This may include factors such as:

* Ensuring that any permissions associated with key ownership are respected in the establishment of the Secure Channel.

* Using cryptographic algorithms appropriately.

* Using key material in accordance with any usage restrictions such as freshness or algorithm restrictions.

* Ensuring that appropriate protections are in place to address potential traffic analysis attacks.

The remaining subsections of this section highlight some aspects of specific cryptography choices that are detailed further in [RFC9053].

7.2. AES-CBC_MAC

* A given key should only be used for messages of fixed or known length.

* Different keys should be used for authentication and encryption operations.

* A mechanism to ensure that IV cannot be modified is required.

Section 3.2.1 of [RFC9053] contains a detailed explanation of these considerations.

7.3. AES-GCM

* The key and nonce pair is unique for every encrypted message.

* The maximum number of messages to be encrypted for a given key is not exceeded.

Section 4.1.1 of [RFC9053] contains a detailed explanation of these considerations.

7.4. AES-CCM

* The key and nonce pair is unique for every encrypted message.

* The maximum number of messages to be encrypted for a given block cipher is not exceeded.

* The number of messages both successfully and unsuccessfully decrypted is used to determine when rekeying is required.

Section 4.2.1 of [RFC9053] contains a detailed explanation of these considerations.

7.5. ChaCha20 and Poly1305

* The nonce is unique for every encrypted message.

* The number of messages both successfully and unsuccessfully decrypted is used to determine when rekeying is required.

Section 4.3.1 of [RFC9053] contains a detailed explanation of these considerations.

8. References

8.1. Normative References

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

[IANA.cwt]
IANA, "CBOR Web Token (CWT) Claims", <https://www.iana.org/assignments/cwt>.

[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/rfc/rfc2119>.

[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>.

[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/rfc/rfc8174>.

[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>.

[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>.

[RFC8725] Sheffer, Y., Hardt, D., and M. Jones, "JSON Web Token Best Current Practices", BCP 225, RFC 8725, DOI 10.17487/RFC8725, February 2020, <https://www.rfc-editor.org/rfc/rfc8725>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <https://www.rfc-editor.org/rfc/rfc8949>.

[RFC9165] Bormann, C., "Additional Control Operators for the Concise Data Definition Language (CDDL)", RFC 9165, DOI 10.17487/RFC9165, December 2021, <https://www.rfc-editor.org/rfc/rfc9165>.

8.2. Informative References

[I-D.ietf-rats-eat] Lundblade, L., Mandyam, G., O'Donoghue, J., and C. Wallace, "The Entity Attestation Token (EAT)", Work in Progress, Internet-Draft, draft-ietf-rats-eat-25, 15 January 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-eat-25>.

[NIST-SP800-90Ar1] Barker, E. and J. Kelsey, "Recommendation for Random Number Generation Using Deterministic Random Bit

Generators", National Institute of Standards and Technology, DOI 10.6028/nist.sp.800-90ar1, June 2015, <https://doi.org/10.6028/nist.sp.800-90ar1>.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007, <https://www.rfc-editor.org/rfc/rfc4949>.

[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/rfc/rfc8446>.

[RFC8747] Jones, M., Seitz, L., Selander, G., Erdtman, S., and H. Tschofenig, "Proof-of-Possession Key Semantics for CBOR Web Tokens (CWTs)", RFC 8747, DOI 10.17487/RFC8747, March 2020, <https://www.rfc-editor.org/rfc/rfc8747>.

[RFC9052] Schaad, J., "CBOR Object Signing and Encryption (COSE): Structures and Process", STD 96, RFC 9052, DOI 10.17487/RFC9052, August 2022, <https://www.rfc-editor.org/rfc/rfc9052>.

[RFC9053] Schaad, J., "CBOR Object Signing and Encryption (COSE): Initial Algorithms", RFC 9053, DOI 10.17487/RFC9053, August 2022, <https://www.rfc-editor.org/rfc/rfc9053>.

[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/rfc/rfc9334>.

[RFC9397] Pei, M., Tschofenig, H., Thaler, D., and D. Wheeler, "Trusted Execution Environment Provisioning (TEEP) Architecture", RFC 9397, DOI 10.17487/RFC9397, July 2023, <https://www.rfc-editor.org/rfc/rfc9397>.

[TPM2] "Trusted Platform Module Library Specification, Family “2.0”, Level 00, Revision 01.59 ed., Trusted Computing Group", 2019.

Appendix A. CDDL

The Concise Data Definition Language (CDDL), as defined in [RFC8610] and [RFC9165], provides an easy and unambiguous way to express structures for protocol messages and data formats that use CBOR or JSON.

[RFC8392] does not define CDDL for CWT Claims Sets.

RFC-Editor: This document uses the CPA (code point allocation) convention described in [I-D.bormann-cbor-draft-numbers]. Please replace the number 601 in the code blocks below by the value that has been assigned for CPA601 and remove this note.

This specification proposes using the definitions in Figure 1 for the CWT Claims Set defined in [RFC8392]. Note that these definitions have been built such that they also can describe [RFC7519] Claims sets by disabling feature "cbor" and enabling feature "json", but this flexibility is not the subject of the present specification.

```
UCCS-Untagged = Claims-Set
UCCS-Tagged = #6.601(UCCS-Untagged)

Claims-Set = {
 * $$Claims-Set-Claims
 * Claim-Label .feature "extended-claims-label" => any
}
Claim-Label = CBOR-ONLY<int> / text
string-or-uri = text

$$Claims-Set-Claims //= ( iss-claim-label => string-or-uri )
$$Claims-Set-Claims //= ( sub-claim-label => string-or-uri )
$$Claims-Set-Claims //= ( aud-claim-label => string-or-uri )
$$Claims-Set-Claims //= ( exp-claim-label => ~time )
$$Claims-Set-Claims //= ( nbf-claim-label => ~time )
$$Claims-Set-Claims //= ( iat-claim-label => ~time )
$$Claims-Set-Claims //= ( cti-claim-label => bytes )

iss-claim-label = JC<"iss", 1>
sub-claim-label = JC<"sub", 2>
aud-claim-label = JC<"aud", 3>
exp-claim-label = JC<"exp", 4>
nbf-claim-label = JC<"nbf", 5>
iat-claim-label = JC<"iat", 6>
cti-claim-label = CBOR-ONLY<7>  ; jti in JWT: different name and text

JSON-ONLY<J> = J .feature "json"
CBOR-ONLY<C> = C .feature "cbor"
JC<J,C> = JSON-ONLY<J> / CBOR-ONLY<C>
```

Figure 1: CDDL definition for Claims-Set

Specifications that define additional Claims should also supply additions to the $$Claims-Set-Claims socket, e.g.:

```
; [RFC8747]
$$Claims-Set-Claims //= ( 8: CWT-cnf ) ; cnf
CWT-cnf = {
  (1: CWT-COSE-Key) //
  (2: CWT-Encrypted_COSE_Key) //
  (3: CWT-kid)
}

CWT-COSE-Key = COSE_Key
CWT-Encrypted_COSE_Key = COSE_Encrypt / COSE_Encrypt0
CWT-kid = bytes

;;; Insert the required CDDL from RFC 9052 to complete these
;;; definitions.  This can be done manually or automated by a
;;; tool that implements an import directive such as:
;# import rfc9052
```

Appendix B. Example

This appendix is informative.

The example CWT Claims Set from Appendix A.1 of [RFC8392] can be turned into a UCCS by enclosing it with a tag number CPA601:

```
601(
  {
    / iss / 1: "coap://as.example.com",
    / sub / 2: "erikw",
    / aud / 3: "coap://light.example.com",
    / exp / 4: 1444064944,
    / nbf / 5: 1443944944,
    / iat / 6: 1443944944,
    / cti / 7: h'0b71'
  }
)
```

Appendix C. JSON Support

This appendix is informative.

The above definitions, concepts and security considerations all may be applied to define a JSON-encoded Claims-Set. Such an unsigned Claims-Set can be referred to as a "UJCS", an "Unprotected JWT Claims Set". The CDDL definition in Figure 1 can be used for a "UJCS".

```
UJCS = Claims-Set
```

Appendix D. EAT

This appendix is informative.

The following CDDL adds UCCS-format and UJCS-format tokens to EAT using its predefined extension points (see Section 4.2.18 (submods) of [I-D.ietf-rats-eat]).

```
$EAT-CBOR-Tagged-Token /= UCCS-Tagged
$EAT-CBOR-Untagged-Token /= UCCS-Untagged

$JSON-Selector /= [type: "UJCS", nested-token: UJCS]
```

Acknowledgements

Laurence Lundblade suggested some improvements to the CDDL. Carl Wallace provided a very useful review.

Authors' Addresses

Henk Birkholz
Fraunhofer SIT
Rheinstrasse 75
64295 Darmstadt
Germany

Email: henk.birkholz@sit.fraunhofer.de

Jeremy O'Donoghue
Qualcomm Technologies Inc.
279 Farnborough Road
Farnborough
GU14 7LS
United Kingdom

Email: jodonogh@qti.qualcomm.com

Nancy Cam-Winget
Cisco Systems
3550 Cisco Way
San Jose, CA 95134
United States of America

Email: ncamwing@cisco.com

Carsten Bormann
Universität Bremen TZI
Postfach 330440
D-28359 Bremen
Germany

Phone: +49-421-218-63921
Email: cabo@tzi.org