EAT Information Model and Data Model

Laurence Lundblade

July 2019

New EAT draft separates information from data model

- CDDL is ***NORMATIVE.*** First time in an RFC??
- Section 3 gives the information model of each claim
  - Textual description of the full semantics
  - CDDL describing each claim
  - Neutral to serialization / representation
- Section 4 describes serialization / representation
  - Aggregates all CDDL and CDDL that forms a token
  - One section for CBOR, another for JSON
  - CBOR / JSON details are contained to section 4

Location as an Example

- Section 3 gives the information model
  - The location describes the location of the device entity from which the attestation originates. … The location coordinate claims are consistent with the WGS84 coordinate system {{WGS84}}….

```
location_type = {
    latitude => number,
    longitude => number,
    altitude => number,
    accuracy => number,
    altitude_accuracy => number,
    heading => number,
    speed => number
}

location_claim = (
location: location_type )
```

- Section 4 gives CBOR / JSON details
- Labels for JSON

```
latitude = "lat""
longitude = "long""
altitude = "alt"
accuracy = "accry"
altitude_accuracy = "alt_accry"
heading = "heading"
speed = "speed"
```

- CBOR

```
latitude = 1
longitude = 2
altitude = 3
accuracy = 4
altitude_accuracy = 5
heading = 6
speed = 7
```

Section 4 General Requirements

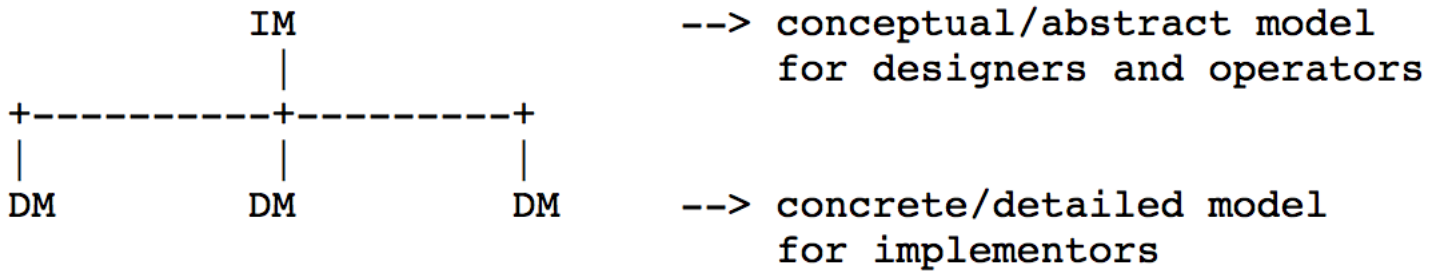
JSON

- JSON should be encoded per RFC 8610 Appendix E. In addition, the following CDDL types are encoded in JSON as follows:
- bstr -- must be base64url encoded
- time -- must be encoded as NumericDate as described section 2 of RFC7519.
- string_or_uri -- must be encoded as StringOrURI as described section 2 of RFC7519.

CBOR (sample of some rules)

- Canonical CBOR encoding, Preferred Serialization and Deterministically Encoded CBOR are explicitly NOT required.
- Integer Encoding (major type 0, 1) -- The entity may use any integer encoding allowed by CBOR. The server MUST accept all integer encodings allowed by CBOR.
- String Encoding (major type 2 and 3) -- The entity can use any string encoding allowed by CBOR including indefinite lengths. …. The server must accept all string encodings.
- Major type 2, bstr, SHOULD be have tag 21 to indicate conversion to base64url in case that conversion is performed.
- Map and Array Encoding (major type 4 and 5) -- The entity can use any array or map encoding allowed by CBOR including indefinite lengths. Sorting of map keys is not required. Duplicate map keys are not allowed. The server must accept all array and map encodings. The server may reject maps with duplicate map keys

Info Model and Data Model – Quotes from RFC 3444:



- …it is not always possible to precisely define what kind of details should be expressed in an IM and which ones belong in a DM.
- There is a gray area where IMs and DMs overlap…
- In some cases, it is very difficult to determine whether an abstraction belongs to an IM or a DM.

EAT is a CWT or a JWT

Laurence Lundblade

July 2019

EAT relation to CWT and JWT

- An EAT is either a CWT or a JWT. The difference is just syntax / encoding.
- When this WG defines an EAT claim we will define it for both CWT and JWT and register it in both registries.
- Just like with CWT and JWT, there is no formal requirement or mechanism that requires that a claim be defined or registered for both, so folks outside this WG may define an attestation for just CWT or just JWT, the same as it is for the authentication use cases.

Registry Rules

CWT and JWT Claim Types and Registration Requirements					
	Registered			Unregistered	
	Expert Review				
	Specification Required				
Token	Standards Action			Collision Resistant	Not Collision Resistant
JWT	All JWT registered labels must be Specification Required. Expert review by itself is not enough.		Not allowed	Strings in OID, DNS or similar forms	Not recommended
CWT	Numeric: -256 to 255 String: length == 1	Numeric: -64K to -257 Numeric: 256 to 64K String length == 2	Numeric: >64K String length > 2	Not supported	Numeric: < -64K
Notes:	Standards Action implies there will be a specification and experts RFC 5226 says that Specification Required includes Expert Review JWT uses no specific syntax to separate the types; CWT always separates types by syntax Fully private use is not shown				

- We will not modify the main rules in the CWT and JWT RFCs for registration of new claims.
- Minimum common to JWT/CWT is Specification Required
- The primary means for use cases and implementations to settle on only high-quality claims is through the creation of profiles, not through the claims registration process (expert review and such). Some profiles could be IETF standards.