CORIM Based Attestation Framework

Presenter: Shanwei Cen

For CCC Attestation SIG

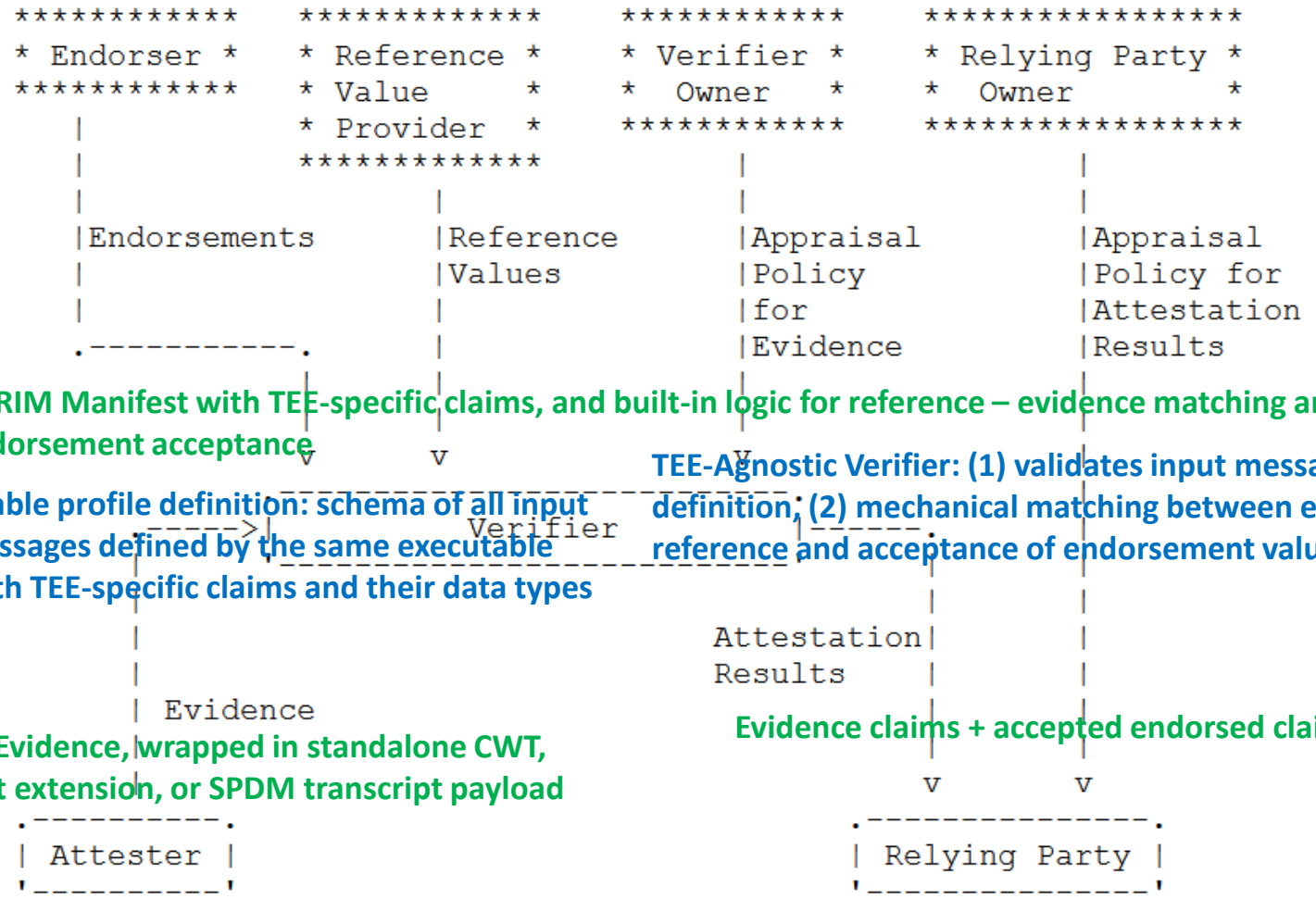
July 18, 2023

Outline

- **Overview**: CORIM based Attestation Framework
- **CORIM** definition
- **Executable TEE Profile** Definition in CDDL
- **TEE-agnostic Verifier** Mechanical Matching
- Example: CORIM-based SGX Attestation
  - Profile definition
  - Concise Evidence
  - CORIM Manifest
  - Verifier Behavior

Overview

CORIM based Attestation Framework: all input and output messages schemas defined inexecutable CDDL profile definition, and TEE-agnostic verifier behavior



CORIM Manifest with TEE-specific claims, and built-in logic for reference – evidence matching and endorsement acceptance

CORIM Executable profile definition: schema of all input and output messages defined by the same executable CDDL script with TEE-specific claims and their data types

TEE-Agnostic Verifier: (1) validates input messages against profile definition; (2) mechanical matching between evidence and reference and acceptance of endorsement values

Concise Evidence, wrapped in standalone CWT, DICE cert extension, or SPDM transcript payload

Evidence claims + accepted endorsed claims

CORIM Based Definitions

- CORIM with support of TEE-specific claims, and logic for reference – evidence matching and endorsement acceptance
  - Draft IETF RATS CORIM triples-map with extensible measurement-values-map, conditional endorsement and series triples
    - **Extensible measurement-values-map** enables TEE-specific profile definition, with its custom claims and their data types (for evidence, reference, and endorsement values)
    - **Conditional endorsement** and **series** triples Uses stateful environment to accept endorsement claims for a specific target environment with specified environment and measurement claims
  - Intel Profile for CoRIM defines **non-exact matching expression** for reference values
    - Matching semantics of equivalence, range, or set membership etc.
    - Broad interest in the community for non-exact matching use cases
- CORIM based concise-evidence and its wrappers (CWT, DICE cert, SPDM)
  - TCG standard concise-evidence, encoded in CBOR
    - Based on the same CORIM CDDL struct reference-triple-record originally for reference values
  - TCG draft DICE Attestation Arch spec defines X.509 cert extension for concise- evidence
    - Extension OID 2.23.133.5.4.9 for CMW that covers CBOR encoded tagged-concise-evidence
  - TCG draft Evidence Binding for SPDM defines SPDM transcript that holds concise-evidence in its payload

CORIM with extensible measurement-values-map, Conditional Endorsements and Series

- IETF RATS draft CORIM standard: manifest as a signed CWT (Concise Web Token)
- Payload is tagged-corim-map -> corim-map -> concise-mid-tag -> triples-map
- triples-map with conditional-endorsement-series-triples and conditional-endorsement-triples
- measurement-values-map extensible to support TEE-specific claims / attributes, for evidence, reference, and endorsements

```
corim = #6.500($concise-rim-type-choice)

$concise-rim-type-choice /= #6.501(corim-map)
$concise-rim-type-choice /= #6.502(signed-corim)

signed-corim = #6.18(COSE-Sign1-corim)

COSE-Sign1-corim = [
  protected: bstr .cbor protected-corim-header-map
  unprotected: unprotected-corim-header-map
  payload: bstr .cbor tagged-corim-map
  signature: bstr
]

tagged-corim-map = #6.501(corim-map)

corim-map = {
  &(id: 0) => $corim-id-type-choice
  &(tags: 1) => [ + $concise-tag-type-choice ]
  ? &(dependent-rims: 2) => [ + corim-locator-map ]
  ? &(profile: 3) => [ + profile-type-choice ]
  ? &(rim-validity: 4) => validity-map
  ? &(entities: 5) => [ + corim-entity-map ]
  * $$corim-map-extension
}
```

```
$concise-tag-type-choice /= #6.506(bytes .cbor concise-mid-tag)
triples-map = non-empty<{
  ? &(reference-triples: 0) => [ + reference-triple-record ]
  ; other entries omitted in this presentation
  ? &(conditional-endorsement-series-triples: 8) =>
    [ + conditional-endorsement-series-triple-record ]
  ? &(conditional-endorsement-triples: 9) =>
    [ + conditional-endorsement-triple-record ]
  * $$triples-map-extension
}>
conditional-endorsement-series-triple-record = [
  stateful-environment-record
  [ + conditional-series-record ]
]
conditional-endorsement-triple-record = [
  stateful-environment-record,
  measurement-values-map
]
conditional-series-record = [
  refv: measurement-values-map
  endv: measurement-values-map
]
```

```
measurement-values-map = non-empty<{
  ? &(version: 0) => version-map
  ? &(svn: 1) => svn-type-choice
  ? &(digests: 2) => [ + digest ]
  ? &(flags: 3) => flags-map
  ? (
      &(raw-value: 4) => $raw-value-type-choice,
      ? &(raw-value-mask: 5) => raw-value-mask-type
    )
  ? &(mac-addr: 6) => mac-addr-type-choice
  ? &(ip-addr: 7) =>  ip-addr-type-choice
  ? &(serial-number: 8) => text
  ? &(ueid: 9) => ueid-type
  ? &(uuid: 10) => uuid-type
  ? &(name: 11) => text
  * $$measurement-values-map-extension
}>
```

CORIM with Matching Expressions

Currently defined in [Intel Profile for CoRIM]

- Matching expressions with matching operators, for non-exact matching between evidence and reference
- Wrapped by CBOR tag (#6.60010 for now), for processing by verifier

```
; Numeric matching operators and expressions
gt = 1
ge = 2
lt = 3
le = 4
numeric-type = integer / unsigned / float
numeric-operator = gt / ge / lt / le
numeric-expression = [ numeric-operator, numeric-type ]
tagged-numeric-expression = #6.60010(numeric-expression)

; Set operators and expressions
member = 6
not-member = 7

set-type = [ * any ]
set-operator = member / not-member
set-expression = [ set-operator, set-type ]
tagged-set-expression = #6.60010( set-expression )

tagged-exp-member = #6.60010([
    member .within set-operator, set-type ])

tagged-exp-not-member = #6.60010([
    not-member .within set-operator, set-type ])
```

TCG Concise Evidence

- TCG repo concise-evidence, built on IETF draft CORIM standard
- Uses CORIM reference-triple-record, the same CORIM schema for reference values, that is a CBOR array composed of environment-map and measurement-map

```
tagged-concise-evidence = #6.571(concise-evidence-map)
concise-evidence = concise-evidence-map

concise-evidence-map = {
  &(ce.ev-triples: 0) => ev-triples-map
  * $$concise-evidence-map-extension
}
$evidence-id-type-choice /= tagged-uuid-type
; additional evidence identifier types may be added here

ev-triples-map = non-empty< {
  ? &(ce.evidence-triples: 0) => [ + reference-triple-record ]
  ? &(ce.identity-triples: 1) => [ + identity-triple-record ]
  ? &(ce.dependency-triples: 2) => [ + domain-dependency-triple-record ]
  ? &(ce.domain-membership-triples: 3) => [ + domain-membership-triple-record ]
  ? &(ce.coswid-triples: 4) => [ + ev-coswid-triple-record ]
  * $$ev-triples-map-extension
} >
```

DICE Cert Extension for Tagged Concise Evidence

- TCG DICE Attestation Architecture v1.1 spec defined ConceptualMessageWrapper extension (OID 2.23.133.5.4.9) for tagged CBOR byte string
  - This is in addition to the existing DiceTcbInfo and DiceTcbInfoSeq extensions
- TCG DICE SPDM (Security Protocol and Data Model) Binding spec defines CBOR tag 571 for concise-evidence

```
tcg-dice-conceptual-message-wrapper OBJECT IDENTIFIER ::=
{tcg-dice 9}
```

The ASN.1 definition is as follows:

```
ConceptualMessageWrapper ::= SEQUENCE {
   cmw OCTET STRING
}
```

The `ConceptualMessageWrapper` sequence contents can be encoded as JSON, CBOR, or tagged CBOR. A parser decodes the OCTET STRING into a byte buffer and then does a 1-byte lookahead using the following pseudo code to decide which format to use to decode the remainder of the octet string:

```
switch b[0] {
case 0x82:
        return CBORArray
case 0x5b:
        return JSONArray
case 0xc0..0xdf:
        return CBORTag
}
```

Executable TEE Profile Definition in CDDL

- **Profile CDDL defines TEE-specific attributes**: their IDs (code points), data types and matching expressions
- **A single CDDL definition for all data**: evidence, reference / endorsements / appraisal policies, attestation results
- **TEE profile CDDL as input to verifier for schema compliance** validation of evidence and reference / endorsements
  - Verifier behavior agnostic to TEE-specific attributes application semantics

TEE-agnostic Verifier Mechanical Matching

- **Matching expressions** contained in Reference Manifest
- **Verifier mechanical matching** by evaluating reference matching expression against evidence value of the same code point
- **Verifier behavior agnostic to TEE-specific** attributes application semantics
  - TEE profile CDDL as input to verifier only for schema compliance validation of evidence and reference data
- **Endorsements are conditional** on the matched references are added to accepted claim set, as part of attestation result

Example: Intel SGX Profile Definition

In Appendix A of Intel Profile for CoRIM , assuming Matching Expression is defined as part of base CORIM

- Executable CDDL definition of TEE specific attributes: codepoints and their evidence and reference data types

```
; measurement-values-map claims codepoints and data types
$$measurement-values-map-extension //= (
  &(tee.mrenclave: -83) => $tee-digest-type
)
$$measurement-values-map-extension //= (
  &(tee.mrsigner: -84) => $tee-digest-type
)
$tee-digest-type /= hash-entry .within set-type
$tee-digest-type /= tagged-exp-member

$$measurement-values-map-extension //= (
  &(tee.isvprodid: -85) => $tee-isvprodid-type
)
$tee-isvprodid-type /= uint

$$measurement-values-map-extension //= (
  &(tee.isvsvn: -73) => $tee-svn-type
)
$tee-svn-type /= numeric-type
$tee-svn-type /= tagged-numeric-ge

$$measurement-values-map-extension //= (
  &(tee.tcbstatus: -88) => $tee-tcbstatus-type
)
$tee-tcbstatus-type /= ([ + tstr ])
$tee-tcbstatus-type /= tagged-exp-subset

$$measurement-values-map-extension //= (
  &(tee.tcbdate: -72) => $tee-date-type
)
$tee-date-type /= tdate
$tee-date-type /= tagged-exp-tdate-ge
$tee-date-type /= tagged-exp-epoch-ge
```

Example: Concise Evidence diag file

Source: Intel Profile for CoRIM example [ice-qe.diag]

```
    / ce.evidence-triples / 0 : [
      [ /** uses reference-triple-record schema **/
        / environment-map / {
          / class / 0 : {
            / class-id / 0 :
              / tagged-oid-type / 111(h'6086480186F84D0102030401'), / 2.16.840.1.
              113741.1.2.3.4.1 - <OID-for-SGX-QE-TCB> /
            / vendor / 1 : "Intel Corporation",
            / model / 2 : "SGX QE TCB"
          }
        },
        / measurement-map / {
          / mval / 1 : {
            / miscselect / -81 : h'00000000', / *** 4 bytes *** /
            / isvprodid / -85 : 1,
            / mrsigner / -84 : [
              / hash-alg-id / 1, / sha256 /
              / hash-value /
              h'A314FC2DC663AE7A6B6BC6787594057396E6B3F569CD50FD5DDB4D1BBAFD2B6A'
            ],
            / attributes / -82 : h'C00000000000000000000000000000000', / *** 16
            bytes *** /
            / mrenclave / -83 : [
              / hash-alg-id / 1, / sha256 /
              / hash-value /
              h'B2F5EB1CB5529E7A6B6BC6787594057396E6B3F569CD50FD5DDB5E2CCB0E3C7B'
            ], / not expected to match ref val /
            / isvsvn / -73 : 2
          },
          / authorized-by / 2 : [
            / tagged-pkix-base64-key-type / 554("base64_key_X")
          ]
        }
      ]
    ]
  }
}
```

Example: CORIM Manifest Diag file

Source: Intel Profile for CoRIM example irim-qe-cend.diag

```
/ concise-mid-tag / {
  / tag-identity / 1 : {
    / tag-id / 0 : "Sample Quoting Enclave RIM"
  },
    / entity / 2 : [ {
    / entity-name / 0 : "INTEL",
    / reg-id / 1 : 32("https://intel.com"),
    / role / 2 : [ 1,0,2 ] / creator, tag-creator, maintainer /
  } ],
  / triples / 4 : {
    / conditional-endorsement-series-triples / 8 : [
      [
        / stateful-environment-record / [
          / environment-map / {
          / class / 0 : {
            / class-id / 0 :
            / tagged-oid-type / 111(h'6086480186F84D0102030401'), / 2.16.840.1.
            113741.1.2.3.4.1 - QE /
          / vendor / 1 : "Intel Corporation",
          / model / 2 : "0123456789ABCDEF" /** CPUID[0x01].EAX.FMSP & 0x0FFF0FF0
          **/
          }
        },
        / measurement-map / {
          / mval / 1 : / measurement-values-map / {
            / miscselect / -81 : 60010([ / mask-eq / 1, h'00000000',
            h'FFFFFFFF' ]),
            / attributes / -82 : 60010([ 1, h'11000000000000000000000000000000',
            h'FBFFFFFFFFFFFFFF0000000000000000']), / *** 16 bytes *** /
            / mrsigner / -84 : 60010([ / op.member / 6,
              / digests-type / [
                [
                  / hash-alg-id / 1, / sha256 /
                  / hash-value /
                  h'8C4F5775D796503E96137F77C68A829A0056AC8DED70140B081B094490C57B
                  FF'
                ]
              ]
            ]),
            / isvprodid / -85 : 1
          },
          / authorized-by / 2 : [
            / tagged-pkix-base64-key-type / 554("base64_key_for-RIM-creator")
          ]
        }
      ],
```

```
      [  / *** triple object - conditional endorsed record series *** /
        [ / *** record 1 *** /
          / refv: measurement-values-map / {
            / isvsvn / -73 : 60010([ / op.ge/ 2, 8 ])
          },
          / endv: measurement-values-map / {
            / tcbdate / -72 : 0("2023-02-15T00:00:00Z"),
            / tcbstatus / -88 : [ "UpToDate" ],
            / tcb-eval-num / -86 : 15
          }
        ],
        [ / *** record 2 *** /
          / refv: measurement-values-map / {
            / isvsvn / -73 : 60010([ / op.ge/ 2, 6 ])
          },
          / endv: measurement-values-map / {
            / tcbdate / -72 : 0("2021-11-10T00:00:00Z"),
            / tcbstatus / -88 : [ "OutOfDate" ]
          }
        ],
        / ... /
        [ / *** record 6 *** /
          / refv: measurement-values-map / {
            / isvsvn / -73 : 60010([ / op.ge/ 2, 1 ])
          },
          / endv: measurement-values-map / {
            / tcbdate / -72 : 0("2018-08-15T00:00:00Z"),
            / tcbstatus / -88 : [ "OutOfDate" ]
          }
        ]
      ]
    ]
  }
}
```

Example: SGX Evidence Verification

TEE Profile CDDL definition

```
; measurement-values-map claims codepoints and data types
$$measurement-values-map-extension //= (
  &(tee.mrenclave: -83) => $tee-digest-type
)
$$measurement-values-map-extension //= (
  &(tee.mrsigner: -84) => $tee-digest-type
)
$tee-digest-type /= hash-entry .within set-type
$tee-digest-type /= tagged-exp-member

$$measurement-values-map-extension //= (
  &(tee.isvprodid: -85) => $tee-isvprodid-type
)
$tee-isvprodid-type /= uint

$$measurement-values-map-extension //= (
  &(tee.isvsvn: -73) => $tee-svn-type
)
$tee-svn-type /= numeric-type
$tee-svn-type /= tagged-numeric-ge

$$measurement-values-map-extension //= (
  &(tee.tcbstatus: -88) => $tee-tcbstatus-type
)
$tee-tcbstatus-type /= ([ + tstr ])
$tee-tcbstatus-type /= tagged-exp-subset

$$measurement-values-map-extension //= (
  &(tee.tcbdate: -72) => $tee-date-type
)
$tee-date-type /= tdate
$tee-date-type /= tagged-exp-tdate-ge
$tee-date-type /= tagged-exp-epoch-ge
```

CORIM Manifest

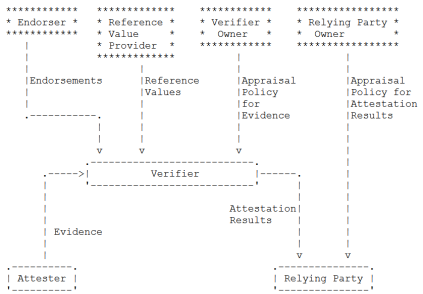
```
CoRIM conditional-endorsement-series-triple-record
    stateful-environment-map
        environment-map
            class-id: 111(h'6086480186F84D0102030401')
        measurement-values-map
            miscselect: 60010([ / mask-eq / 1, h'00000000', h'FFFFFFFF' ]),
            attributes: 60010([ / mask-eq / 1,
                                h'11000000000000000000000000000000',
                                h'FBFFFFFFFFFFFFFF0000000000000000'])
            mrsigner: 60010([/ op.member / 6, [ xxx ] ])
            isvprodid: 1
    conditional-series-record 1
        refv: measurement-values-map
            isvsvn: 60010([ / op.ge/ 2, 6 ])
        endv: measurement-values-map
            tcbdate : 0("2023-02-15T00:00:00Z"),
            tcbstatus : [ "UpToDate" ],
    ...
    conditional-series-record 6
        refv: measurement-values-map
            isvsvn: 60010([ / op.ge/ 2, 1 ])
        endv: measurement-values-map
            tcbdate : 0("2018-08-15T00:00:00Z"),
            tcbstatus : [ "OutOfDate" ],
```

TEE-agnostic Verifier behavior:

- Validates input evidence against profile CDDL script
- Evaluate of CORIM conditional-endorsement-series-triple-record
  - Matches stateful-environment-map environment-map class-id
  - Matches stateful-environment-map measurement-values-map claims against evidence claims
    - Evaluate matching expressions
  - Find first successful conditional-series-record
    - Matches refv measurement-values-map against evidence
      - Evaluate matching expressions
    - Accept endorsement claims in endv measurement-values-map

```
Evidence reference-triple-record
    environment-map
        class-id: 111(h'6086480186F84D0102030401')
    measurement-values-map
        miscselect: h'00000000'
        attributes: h'C0000000000000000000000000000000'
        mrenclave: xxx
        mrsigner: xxx
        isvprodid: 1
        isvsvn: 6
```

Concise Evidence



TEE-agnostic Verifier

```
Attestation Result reference-triple-record
    environment-map
        class-id: 111(h'6086480186F84D0102030401')
    measurement-values-map
        miscselect: h'00000000'
        attributes: h'C0000000000000000000000000000000'
        mrenclave: xxx
        mrsigner: xxx
        isvprodid: 1
        isvsvn: 2
        tcbdate : 0("2023-02-15T00:00:00Z"),
        tcbstatus : [ "UpToDate" ],
```

Attestation Result

Backup