TRUSTED® COMPUTING GROUP

SPECIFICATION

TCG Component RIM Binding for CoRIM

Version 1.0

August 26, 2026

Contact: admin@trustedcomputinggroup.org

PUBLISHED

DISCLAIMERS, NOTICES, AND LICENSE TERMS

Acknowledgements

The writing of a specification, particularly a security specification, takes many hours for both development and review. The TCG would like to acknowledge the contribution of those individuals (listed below) and the companies who allowed them to volunteer their time to the development of this specification. Special thanks are due to Amy C Nelson and Jiewen Yao, who served as Chairs of the PC Client Working Group during the development of this specification.

The TCG would also like to give special thanks to Amy C Nelson who served as the editor of this specification.

Contributors

Muhammad Monir Hossain, Advanced Micro Devices, Inc.
Frederick Otumfuor, American Megatrends, Inc.
Karthika R, American Megatrends, Inc.
Sivasangara Subramanian S, American Megatrends, Inc
Prachotran Bathi, ARM Ltd.
Maxwell Prittle, ARM Ltd.
Stuart Yoder, ARM Ltd.
Dong Wei, ARM Ltd.
Bill Sulzen, Cisco Systems
Goutam Madhukeshwar Hegde, Cisco Systems
Eugene Cho, Dell, Inc.
Travis Gilbert, Dell, Inc.
Amy C Nelson, Dell, Inc.
Henk Birkholz, Fraunhofer Institute for Secure Information Technology
Jeff Anderson, Google Inc.
Chris Fenner, Google Inc.
Brad Litterell, Google Inc.
Jessica McClintock, Google Inc.
Shiva Dasari, Hewlett Packard Enterprise
Ken Geer, Hewlett Packard Enterprise
Kevin Micciche, Hewlett Packard Enterprise
Subramanian Swaminathan, Hewlett Packard Enterprise
Vali Ali, HP Inc.
Endrigo Pinheiro, HP Inc.
Joshua Schiffman, HP Inc.
Adrian Shaw, HP Inc.
Thomas Deleuran, Huawei Technologies Co., Ltd.
Yide Lian, Huawei Technologies Co., Ltd.
Zhuo Liu, Huawei Technologies Co., Ltd.
Silvia Vlasceanu, Huawei Technologies, Co., Ltd.
Peng Xu, Huawei Technologies Co., Ltd.
Ken Goldman, IBM
Klaus Kiwi, IBM
Eran Rom, IBM
Ga-Wai Chin, Infineon Technologies
Antonio Javier Cabrera Gutierrez, Infineon Technologies
Joseph Kohn, Infineon Technologies
Florian Schweiger, Infineon Technologies
Thomas Bowen, Intel Corporation
TianTeck Chong, Intel Corporation
Liran Perez, Intel Corporation
Jiewen Yao, Intel Corporation

Jim Mann, Jim Mann
Bin Hu, Lenovo (US) Inc.
Sandy Collins, Lenovo (US) Inc.
Bill Keown, Lenovo (US) Inc.
Ruxin Li, Lenovo (US) Inc.
Collin Parker, Lenovo (US) Inc.
Kazuo Shiba, Lenovo (US), Inc.
Jonathan McDowell, Meta Platforms Inc.
Jeannette Wilson, Microchip Technology
Derek Adam, Microsoft
Austin Fisher, Microsoft
Charles Ludwig, Microsoft
Dennis Mattoon, Microsoft
Prasanna Karthik Mutharaju, Microsoft
Nathan Sherman, Microsoft
Xin Liu, NSING Technologies Inc.
Yaakov Hayun, Nuvoton Technology Corporation
Galit Heller, Nuvoton Technology Corporation
Dan Morav Nuvoton Technology Corporation
Ludovic Jacquin, NVIDIA Corporation
Theo Koulouris, NVIDIA Corporation
Joe Pennisi, NVIDIA Corporation
Yazan Siam, NVIDIA Corporation
Sahil Uppal, NVIDIA Corporation
Hank Cohen, OmniTrust LLC
Xeno Kovah, OpenSecurityTraining Inc. (DBA OpenSecurityTraining2)
Tim Hoppen, Phoenix Technologies
Saheb Biswas, Seagate Technoogy
Eric Nelson, Seagate Technology
Patrick Debaenst, SEALSQ
Jean Fioretti, SEALSQ
Will Pan, SEALSQ
Mario Ando, STMicroelectronics
Olivier Collart, STMicroelectronics
Nourdine El Idrissi, STMicroelectronics
Yves Magnaud, STMicroelectronics
Charly Villette, STMicroelectronics
Thanassis Giannetsos, Technical University of Denmark
Greg North, Texas Instruments
John Kucera, United States Government
Lawrence Reinert, United States Government

CONTENTS

Table of Tables

1 SCOPE and Context

This TCG Component RIM Binding for CoRIM Specification complies with the TCG Component Reference Integrity Manifest Information Model Specification [17] and provides additional requirements for a Concise Reference Integrity Manifest (CoRIM) file.

1.1 Audience

This Specification aids the creation of binding specifications that define the formatting, structure, and usage guidelines for a given family of platforms. Verifier developers and component developers working with Component RIM binding specifications may need to refer to this Specification for Component RIM element definitions. They may also need to refer to the TCG Reference Integrity Manifest Information Model Specification [3].

1.2 Key Words

The key words “MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,” “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” in this document normative statements are to be interpreted as described in RFC-2119, Key words for use in RFCs to Indicate Requirement Levels.

1.3 Statement Type

Please note a very important distinction between different sections of text throughout this document. There are two distinctive kinds of text: informative comment and normative statements. Because most of the text in this specification will be of the kind normative statements, the authors have informally defined it as the default and, as such, have specifically called out text of the kind informative comment. They have done this by flagging the beginning and end of each informative comment and highlighting its text in gray. This means that unless text is specifically marked as of the kind informative comment, it can be considered a kind of normative statement.

EXAMPLE: Start of informative comment

This is the first paragraph of 1–n paragraphs containing text of the kind *informative comment* ...

This is the second paragraph of text of the kind *informative comment* ...

This is the nth paragraph of text of the kind *informative comment* ...

To understand the TCG specification the user must read the specification. (This use of MUST does not require any action).

End of informative comment

2 Introduction

This specification defines the following items:

1. Component RIM IM to CoRIM Mapping
2. Component RIM CoRIM encoding.
3. Component RIM CoRIM Extension for SPDM (DMTF Security Protocol and Data Model Specification) [16].
4. Component RIM CoRIM Support RIM format for SPDM.

2.1 Glossary

This specification uses the following terms defined in Table 1 Glossary.

Table 1 Glossary

Term	Definition
Component	An element of a platform that comprises one or more integrated circuits with firmware and read-only memory (ROM), each of which may be supplied by different vendors and have different configurations of firmware and/or ROM. See the TCG Component Reference Integrity Manifest Information Model [17]
Platform	A platform comprises of one or more components assembled and working together to deliver a specific computing function but does not include any software other than the firmware that is part of the components in the platform. Examples of platforms include a notebook, a desktop, a server, a network switch, a blade, etc. See NIST SP 800-193 [26]
Assertions	See the TCG Reference Integrity Manifest (RIM) Information Model [3]
Evidence	See the TCG Component Reference Integrity Manifest IM Specification [17]
Verifiers	See the TCG Component Reference Integrity Manifest IM Specification [17]
Security Version Number (SVN)	The version number of a component that indicates that security relevant updates have been applied to the component.
Globally Unique Identifier (GUID)	Defined in the SWID ISO specification [1]
Object Identifier (OID)	Defined in IETF RFC 5280 [11]
Universal Unique Identifier (UUID)	Defined in IETF RFC 4122 [12]
Binding Specification	A TCG specification that tailors the requirements of an Information Model for specific use cases. A binding specification does not loosen requirements of the Information Model but might further constrain the Information Model, define encoding requirements, and possibly standardize values for specific fields.
Support RIM	See the TCG Reference Integrity Manifest (RIM) Information Model [3]

2.2 Relationships to Other Documents

2.2.1 TCG Specifications

Start of informative comment

The following sections summarize TCG Specifications that define or use assertion data that may assist the reader in understanding this specification.

End of informative comment

2.2.1.1 Trusted Attestation Protocol (TAP)

Start of informative comment

TCG's Trusted Attestation Protocol (TAP) Information Model Specification [8] defines the information elements used by Verifiers. Not all the information elements are required by every Verifier. A RIM is essential for TAP-based attestation. A RIM provides the assertions needed by the Verifier that implements the TAP model. Future versions of the TAP may provide guidance on obtaining RIM information from the Attester.

End of informative comment

2.2.1.2 TCG RIM Information Model

Start of informative comment

The Reference Integrity Manifest (RIM) Information Model (IM) Specification [3] defines an abstract structure for assembling reference measurements (Assertions) that manufacturers and other supply chain entities assert as expected values. The RIM IM requires that a binding specification refine the elements of a RIM IM and enable implementations.

End of informative comment

2.2.1.3 TCG Component Reference Integrity Manifest Information Model

Start of informative comment

The Component Reference Integrity Manifest (Component RIM) Information Model (IM) Specification [17] defines an abstract structure for assembling reference measurements (Assertions) that component manufacturers and other supply chain entities assert as expected values. The Component RIM IM references the RIM IM for structural definitions but further abstracts the elements to enable mapping to various encoding schemas such as Software Identification (SWID) Tags, Concise Software Identification (CoSWID) Tags, Concise Reference Integrity Manifest (CoRIM) and Concise Model Identifier (CoMID). The Component RIM IM relies on a binding specification, such as this specification, to define a realization of a RIM information model definition.

End of informative comment

2.2.1.4 TCG PC Client RIM

Start of informative comment

The PC Client RIM Specification [4] complies with the RIM Information Model and provides additional requirements for PC Client platforms. It describes the RIM file formats, RIM storage locations within the PC Client, and provides references for the content of the support RIM.

End of informative comment

2.2.1.5 TCG Firmware Integrity Measurement (FIM)

Start of informative comment

The PC Client Firmware Integrity Measurement (FIM) Specification [5] outlines the basic process for collecting, reporting, and processing (attestation) of PC Client firmware. It also provides requirements for and a mechanism to locate a Platform Certificate compliant with the TCG Platform Certificate Profile Specification [6].

End of informative comment

2.2.1.6 TCG Platform Certificate

Start of informative comment

The TCG Platform Certificate Profile Specification [6] contains assertions about trust made by a platform manufacturer. The certificate asserts the platform's security properties and configuration as shipped. The Platform Certificate Profile defines a mechanism that can be used to incorporate a Component Certificate that might be coupled with a Component RIM.

End of informative comment

2.2.1.7 DICE Attestation Architecture

Start of informative comment

The DICE Attestation Architecture [7] defines an attestation architecture for Device Identifier Composition Engine (DICE) layering architectures.

End of informative comment

2.2.1.8 TCG Algorithm Registry

Start of informative comment

The TCG Algorithm Registry [19] defines the cryptographic algorithms assigned identifiers (IDs) by TCG. Algorithms used in this specification have been assigned algorithm IDs. This specification does not permit the use of algorithms that are not recognized by TCG.

End of informative comment

2.2.2 Non-TCG Documents

Start of informative comment

This section identifies industry (non-TCG) information model specifications for manifest structures. This Specification is patterned after this work.

End of informative comment

2.2.2.1 DMTF Security Protocol and Data Model (SPDM) Specification

Start of informative comment

DMTF DSP0274 Security Protocol and Data Model (SPDM) Specification [16] defines the data structures and mechanisms for a caller to authenticate the identity of an SPDM device and/or obtain a measurement of the device's state for the purposes of attestation.

End of informative comment

2.2.2.2 NIST IR 8060

Start of informative comment

The National Institute of Standards and Technology Interagency Report (NIST IR) 8060 [2], "Guidelines for the Creation of Interoperable Software Identification (SWID) Tags" is the primary reference for the elements described in this specification. NIST IR 8060 derives its definitions from ISO-IEC 19770-2 [1] . Because the TCG RIM Model Specification [3] is focused on integrity there are further restrictions and additional requirements for the information elements above and beyond the guidelines found in NIST IR 8060.

End of informative comment

2.2.2.3 ISO-IEC 19770-2 (SWID)

Start of informative comment

ISO-IEC 19770-2 [1] International Organization for Standardization/International Electrotechnical Commission "Software identification tag" is known as the "SWID Specification" and is the main reference source for NIST IR 8060.

End of informative comment

2.2.2.4 CoSWID

Start of informative comment

Concise Software Identification (CoSWID) Tags [13] defines a concise representation of ISO/IEC 19770-2:2015 Software Identification (SWID) Tags that are interoperable with the XML schema definition of ISO/IEC 19770-2:2015 and augmented for application in Constrained-Node Networks.

End of informative comment

2.2.2.5 CoRIM

Start of informative comment

Concise Reference Integrity Manifest (CoRIM) [14] represents Endorsements and Reference Values in CBOR [9] format. Composite devices or systems are represented by a collection of Concise Model Identifier (CoMID) and Concise Software Identifiers (CoSWID) bundled in a CoRIM document.

End of informative comment

2.2.2.6 XML Signature Syntax and Processing

Start of informative comment

The XML Signature Syntax and Processing Version 2.0 [15] is an informative W3C Working Group Note that describes XML digital signature processing rules and syntax. XML Signatures provide integrity, message authentication, and/or signer authentication services for data of any type, whether located within the XML that includes the signature or elsewhere.

End of informative comment

2.2.2.7 CBOR Object Signing and Encryption (COSE)

Start of informative comment

The CBOR Object Signing and Encryption (COSE) [22], which supersedes [10], describes how to create and process signatures, message authentication codes, and encryption using CBOR for serialization. The signature format is used by CoSWID [13] and CoRIM [14]. The COSE algorithm is defined by IANA COSE algorithm [21].

End of informative comment

2.2.2.8 IANA CBOR Tags Registry

Start of informative comment

The CBOR Tags are defined in the IANA registry [20]. This registry assigns identifiers that are used by Component RIMs.

End of informative comment

2.2.2.9 IETF CDDL

Start of informative comment

The IETF specification for Concise Data Definition Language (CDDL) [25] is used for representing CoRIM formatted structures.

End of informative comment.

2.3 Background

Start of Informative Comment

There are different types of component assemblies that range from a simple integrated circuit with firmware and some ROM to printed circuit board assemblies consisting of multiple integrated circuits, each of which may be supplied by different vendors and have different configurations of firmware and/or ROM. This specification supports these different types of components, which vary in complexity. To help a Verifier in understanding the complexity of firmware composition for component assemblies, this specification treats Component RIMs as composable elements.

A Component RIM contains identifying information that describes the composition of the component, its Attesting Environment, an endorsement by one or more authorities or Endorsers, and potentially references pointing to related RIMs. A Component RIM also contains one or more reference measurements that the Endorser asserts are “correct” for that class of component and firmware.

There are various schemas and encoding methods for RIMs. A Component RIM may use the schema for a SWID or CoSWID tag, as defined in the TCG RIM Information Model [3], or it may use the schema for a CoRIM, as defined in

[14]. This specification tailors the Information Module based on the CoRIM schemas specifically as relates to SPDM-based measurements.

End of informative comment

3 Requirements

3.1 TAG Roles

Start of informative comment

Depending on the encoding chosen for a RIM, different roles may exist. Table 2 RIM Roles, as defined in the Component RIM IM, maps the roles for the encoding options to the roles. This Specification uses only the CoMID and CoRIM roles.

This Specification does not normatively define any of the roles in Table 2 RIM Roles except for the Component RIM roles. The other roles identified in Table 2 RIM Roles are normatively defined in referenced specifications identified in the column heading.

Table 2 RIM Roles

RIM Roles	Component RIM IM	SWID [1]	CoSWID [13]	CoMID [14]	CoRIM [14]
Tag Creator	tagCreator	tagCreator	tag-creator	tag-creator	n/a
SW/FW Creator	softwareCreator	softwareCreator	software-creator	creator	n/a
Manifest Creator	tagCreator	tagCreator	tag-creator	n/a	manifest-creator
Tag Endorser	tagSigner	tagSigner			corim-signer

End of informative comment

1. The entity responsible for creation of a RIM SHALL assume the role of manifest-creator in CoRIM and/or tag-creator in CoMID.
2. The entity responsible for creation of the firmware described by a RIM SHALL assume the role of creator in CoMID or softwareCreator in CoSWID.
3. The entity responsible for endorsement of a RIM SHALL assume the role of corim-signer in CoRIM.
4. When a RIM is created, the following rules apply to the RIM entity field:
    a. The entity that creates the RIM SHALL be identified in the RIM entity field with the manifest-creator role in CoRIM.
    b. At a minimum, the manifest-creator SHALL be populated in the entity field.

    Note: see Table 3 Component RIM CoRIM Encoding for more information.

3.2 Component RIM CoRIM Requirements

Start of informative comment

A Verifier correlates a Tag with a Component and the firmware package deployed on the component using the Identity and Meta element attributes. Attributes componentModel and componentVersion are the manufacturer model or marketing name of the component and the version (or revision) of the component. The combination of the componentManufacturer, componentModel, and firmwareVersion allow matching of a Tag to a Class of Component with a specific firmware image. The combination of componentManufacturer and componentModel, possibly along with the firmwareVersion, uniquely identify the Tag.

A Component manufacturer may provide a Component Attribute or Identity Certificate separate from the Component RIM. If so, the Component RIM may include a reference to locate the Component Certificate. A Verifier may use a Certificate to map the Tag to the Attesting Environment. To facilitate this, binding specifications may require the presence of the component componentLocator fields. The componentLocator attribute may be used to obtain device identity or attribute certificates, but without device instance information.

A Component manufacturer may provide a componentManufacturerID that includes a standards body name and standards body defined vendor ID. The component manufacturer needs to choose the standards body based on the type of component and the interface the component provides to query the defined ID. For example, a USB device would use the USB organization defined vendor identifier; a PCIe device would use the PCI-SIG defined vendor identifier. In these examples, the standards body name for the USB device would be “USB” and the standards body name for the PCIe would be “PCI-SIG”.

End of informative comment

Table 3 Component RIM CoRIM Encoding according to IETF Concise Reference Integrity Manifest [14]

CoRIM Element	CoRIM Attribute	General Name in Component RIM IM	Type	Required Mandatory (M) / Optional (O)	Description
corim-map	entities.corim-role-type-choice (manifest-creator, manifest-signer)	Endorser Role	String	M	Role of the entity identified by the Name element
	entities.entity-name	Endorser	String	M	Name of the organization that created the firmware and/or created and endorsed this RIM. If the Tag creator is different than the firmware creator, there might be more than one entity present. See Section 3.1 TAG Roles
	entities.reg-id	Endorser URI	String	O	URI for the entity identified by the name element
	dependent-rims	Component Identity or Attribute Cert URI	URI	O	URI for the repository for information about the Component.
concise-mid-tag	tag-identity.tag-id	Tag Identity	UID	M	Unique identifier used to identify this Tag.
	tag-identity.tag-version	Tag Version	String	M	Version of the Tag Integer.
	entities.comid-role-type-choice (tag-creator, creator)	Endorser Role	String	M	Role of the entity identified by the Name element
	entities.entity-name	Endorser	String	M	Name of the organization that created the firmware and/or created and endorsed this RIM. If the Tag creator is different than the firmware creator, there might be more than one entity present. See Section 3.1 TAG Roles
	entities.reg-id	Endorser URI	String	O	URI for the entity identified by the name element
	linked-tags.tag-rel (tag-rel-type-choice=replaces)	Tag Lifecycle Type - Patch	String	O1	Indicates a Patch RIM. True for a Patch RIM. False for a Primary RIM, if attribute is included.

	linked-tags.tag-rel (tag-rel-type-choice=supplements)	Tag Lifecycle Type - Supplemental	String	O1	Indicates a Supplemental RIM. True for a Supplemental RIM. False for a Primary RIM, if attribute is included.
	triples.reference-triples.ref-env.class.vendor	Attesting Component Manufacturer	String	O1	Manufacturer name or Vendor ID for the Manufacturer of the Component associated with this Tag. If vendor ID is used, the string is a concatenation of a standards body name and the vendor ID defined by the standard body, with following format: “<standard_body_name>:<vendor_id>”. The standard body name SHALL be one of the following: • “TCG” – vendor ID is defined in TCG Vendor ID Registry [27]. • “PCI-SIG” – vendor ID is defined in PCI-SIG Vendor ID [28]. • “USB” – vendor ID is defined in USB Vendor ID [29]. • “IANA” – vendor ID is defined in IANA Private Enterprise Number (PEN) [30]. • “MIPI” – vendor ID is defined in MIPI Manufacturer ID [31]. • “JEDEC” – Vendor ID is defined in JEDEC Manufacturer ID [32]. See Section 3.2.3 Component RIM Attesting Environment Element for specific requirements.
	triples.reference-triples.ref-env.class.model	Attesting component model	String	M	Model of the Component with which this RIM is associated

	triples.reference-triples.ref-claims.mval.version	Attesting Component FW version	String	M	The Version Number for the FW described by this Tag.
	triples.reference-triples.ref-claims.mval.svn	Attesting Component FW SVN	String	M	The Security Version Number (SVN) for the FW described by this Tag.
	triples.reference-triples.ref-claims.mval.digests	Reference Value Collection Hash	String	O1	A digest of type ReferenceValue. See Section 3.2.5 Component RIM Reference Values Element for specific requirements.
	triples.reference-triples.ref-claims.mval.raw-value	Reference Value	String	O1	A raw value of type ReferenceValue. See Section 3.2.5 Component RIM Reference Values Element for specific requirements.
	triples.reference-triples.ref-claims.mval.name	Reference Value Name	String	O1	Name of the measurement.
	tcg-component-rim-consise-mid-tag-ext.binding-spec	Binding Spec Name	String	M	Name of the Binding Specification with which the RIM complies. See Section 4.2.1 SPDM Measurement Information in CoRIM CBOR for specific extension.
	tcg-component-rim-consise-mid-tag-ext.binding-spec-version	Binding Spec Version	String	O	Version of the Binding Specification with which the RIM complies. See Section 4.2.1 SPDM Measurement Information in CoRIM CBOR for specific extension.
	tcg-component-rim-consise-mid-tag-ext.payload-type	Reference Value Location	String	M	Indicates whether the Reference Values are included in the Tag directly or by reference. Only “Direct” is supported in this version.
Signature	algorithm identifier (alg)	Signature	Integer	M	Required to be a TCG-listed algorithm [19]. The algorithm number value is defined as an IANA COSE algorithm [21].
	key identifier (kid)	Signature	Byte String	M	Reference to the certificate used to validate the Tag signature
	timestamp	Signature	String	O	Supported in COSE Countersignature [23] only

	signature	Signature	Byte String	M	See Section 3.2.6 Component RIM Signature Element

Note: O1 indicates the element is conditionally optional

All Attributes defined in Table 3 Component RIM CoRIM Encoding as Mandatory (M) SHALL be present in a Component RIM encoded as CoRIM. All Attributes defined in Table 3 Component RIM CoRIM Encoding as conditionally optional (O1) have conditions applied to their presence defined in the table or subsequent sections.

The following TCG-defined CDDL definition uses the existing CDDL extension points as defined in CoRIM [14].

```
$$concise-mid-tag-extension //= (
  &(tcg-component-rim-concise-mid-tag-ext: 65537) => tcg-component-
rim-concise-mid-tag-ext-entry
)

tcg-component-rim-concise-mid-tag-ext-entry = {
  ? (binding-spec: 0) => text,
  ? (binding-spec-version: 1) => text,
  ? (payload-type: 2) => text,
}
```

CoRIM allows CoSWID tag entries. The definition for CoSWID can be found in TCG Component RIM Binding for SWID and CoSWID [18].

3.2.1 Component RIM Endorsement Identity and Life Cycle Elements

Start of informative comment

The Endorsement Identity and Life Cycle Elements, as defined in the TCG Component RIM IM [17] provide a Verifier with the information necessary to correlate a RIM with a firmware version or ROM version and the measurements collected for that firmware package in its Attesting Environment.

End of informative comment

1. A base RIM SHALL include the reference values for all measurements.
2. A patch RIM SHOULD be used when the firmware has a patch version release, e.g., firmware bug fix. A patch RIM SHALL include the reference values for the measurements that have changes. A patch RIM MAY include the reference values for all measurements.
3. A supplemental RIM SHOULD be used when the firmware is configured by tools after release, e.g., the firmware configuration changes. A supplemental RIM SHALL include the reference values for the measurements that have changes. A supplemental RIM MAY include the reference values for all measurements.
4. The firmware version definition SHOULD follow “major.minor.patch” version semantics.

3.2.2 Component RIM Endorser Identity Element

Start of informative comment

The Endorser Identity Element, as defined in [17], contains information about the entities that play a role in the development of the firmware for a component and the creation of its Component RIM. A role is associated with an Endorser Identity. The simplest case is a single Endorser that fulfils the roles of tagCreator, tagSigner, and softwareCreator. To enable a broad set of platform and component types, this information model provides for, but does not require, multiple Endorser Identities to be represented in the RIM.

End of informative comment

3.2.3 Component RIM Attesting Environment Element

Start of informative comment

The Attesting Environment Element, as defined in [17], contains additional information useful for a Verifier to associate a RIM with a particular Attesting Environment, as well as the binding specification information with which the RIM complies.

End of informative comment

1. If present, concise-mid-tag.triples.reference-triples.ref-claims.mval.version SHALL identify the version of firmware.
2. If present, concise-mid-tag.triples.reference-triples.ref-claims.mval.svn SHALL identify the security version of the firmware.
3. A class of components SHALL be uniquely identified in the RIM as triples.reference-triples.ref-env.class.vendor.

3.2.4 Additional Information Element

Start of informative comment

The Additional Information Element, as defined in [17], contains information that a Verifier might use to enable correlation of the RIM with a particular component. This includes information such as a reference to a Component Identity Certificate, and information to enable parsing of the RIM, such as the Binding Specification element.

End of informative comment

3.2.5 Component RIM Reference Values Element

Start of informative comment

The payload Attribute (see Section 2.2.1.2 TCG RIM Information Model) contains Reference Values or a link to the Support Files.

End of informative comment

1. The tcg-component-rim-concise-mid-tag-ext.payload-type SHALL be present. Only “Direct” is supported in this version. The reference measurements SHALL be included in the triples.reference-triples.ref-claims.mval.
2. If a Component RIM represents a component that supports the SPDM protocol, the following information SHALL be present in the Payload for each reference measurement:
    a. SPDM Version
    b. SPDM MeasurementBlockIndex
    c. SPDM MeasurementSpecification
    d. DMTFSpecMeasurementValueType
    e. The DMTFSpecMeasurementValue SHALL contain the data corresponding to the measurement reported by the SPDM responder.

3. If a Component RIM represents a component that does not support the SPDM Protocol or DICE, the reference measurement SHOULD be provided with sufficient class information for a Verifier to correlate the reference values to the evidence from the Attesting Environment.
4. If a RIM represents a component that supports DICE:
    a. The following class information from TcbInfo defined in DICE Attestation Architecture [7] SHOULD be present in the Payload for each reference measurement:
        i. Vendor
        ii. Model
        iii. Layer
        iv. Index
        v. Type
        vi. Version
        vii. SVN
        viii. Flags
    b. Additional class information from TcbInfo MAY be present.
    c. The Payload reference measurement SHALL contain TcbInfo Fwids defined in DICE Attestation Architecture [7].

3.2.6 Component RIM Signature Element

Start of informative comment

The Signature Element (see Section 2.2.1.3 TCG Component Reference Integrity Manifest Information Model) constitutes a single “endorsement”.

End of informative comment

1. If a Component RIM includes a signature using CoRIM, the COSE Structures and Process [22] MUST be used.
    a. The Signature SHALL be a single signature as defined in [22], e.g. COSE_Sign1. The externally supplied data SHALL be a zero-length byte string as defined in RFC9052 [22].
    b. COSE Countersignatures [23] MAY be used.
    c. The signature MUST contain an algorithm identifier (alg) field that references a TCG-listed algorithm [19] used to sign the RIM. The algorithm number value is defined as an IANA COSE algorithm [21]. It MUST be in a protected header, as defined in [23].
    d. The signature MUST contain a key identifier field (kid) that references a certificate used by a Verifier to validate the signature on the RIM.
        i. The kid SHOULD be the subject key identifier field from the signing certificate.
        ii. The signature SHOULD be in a protected header, as defined in [23].
    e. The signature MUST contain a signature field that contains a cryptographic signature used by a Verifier to validate the signature on the RIM.

4 Component RIM Attributes Mapping for CoRIM

4.1 Component RIM Format

The format SHALL be compliant with IETF Concise Reference Integrity Manifest Specification [14].

4.2 RIM CoRIM Extension for SPDM

Start of informative comment

When the payload type is “Direct”, the Payload/Resource (see Section 2.2.1.3 TCG Component Reference Integrity Manifest Information Model) includes SPDM measurement blocks information. Resource Collections may include multiple resources. Each resource includes one SPDM measurement information.

End of informative comment

4.2.1 SPDM Measurement Information in CoRIM CBOR

The following TCG-defined CDDL definition uses the existing CDDL extension points as defined in CoRIM [14].

```
$$measurement-values-map-extension //= (
  &(spdm-measurement-info: 65537) => spdm-measurement-info-map
)

spdm-measurement-info-map = {
  ? &(spdm-version: 0) => text
  ? &(spdm-measurement-block-index: 1) => text
  ? &(spdm-measurement-specification: 2) => text
  ? &(spdm-dmtf-spec-measurement-value-type: 3) => text
}
```


Appendix A: Component RIM Example in CoRIM

A.1 RIM CoRIM Extension for SPDM

A.1.1 SPDM measurement in CoRIM CBOR

The following example shows the CBOR based SPDM measurement block in a CoRIM Tag. The definition can be found in 4.2.1 SPDM Measurement Information in CoRIM CBOR.

```
      / comid.triples / 4 : {
        / comid.reference-triples / 0 : [
          [
            / environment-map / {
              / comid.class / 0 : {
                / comid.vendor / 1 : "device_vendor_AAA",
                / comid.model / 2 : "device_model_AAA",
                / comid.layer / 3 : 0
              }
            },
            / measurement-map / {
              / comid.mval / 1 : {
                / spdm-measurement-info / 65537 : {
                  / spdm-version / 0 : "0x1200",
                  / spdm-measurement-block-index / 1 : "0x01",
                  / spdm-measurement-specification / 2 : "0x01",
                  / spdm-dmtf-spec-measurement-value-type / 3 : "0x00"
                },
                / comid.digests / 2 :  [
                  [
                    / hash-alg-id / 8, / SHA512 /
                    / hash-value /
h'8d531d77d821e167114d1eb07e0ae19cfb565152408843c768f1135b548fdfa13a203e5c7f129ceacc0
17df26c999f62da26dbf2e1128345ec0f65d37f87ca41'
                  ]
                ]
              }
```



```
      }
    ],
    [
      / environment-map / {
        / comid.class / 0 : {
          / comid.vendor / 1 : "device_vendor_AAA",
          / comid.model / 2 : "device_model_AAA",
          / comid.layer / 3 : 0
        }
      },
      / measurement-map / {
        / comid.mval / 1 : {
          / spdm-measurement-info / 65537 : {
            / spdm-version / 0 : "0x1200",
            / spdm-measurement-block-index / 1 : "0x02",
            / spdm-measurement-specification / 2 : "0x01",
            / spdm-dmtf-spec-measurement-value-type / 3 : "0x01"
          },
          / comid.digests / 2 :  [
            [
              / hash-alg-id / 8, / SHA512 /
              / hash-value /
h'9effd8a668f76d3fce35451a136f8ef6710260e9ca28beef897f559fcdba48a4c066560fb4900195cae
4d4fab1f7d11243421008af8614d92a3fcabbbf75248f'
            ]
          ]
        }
      }
    ],
    [
      / environment-map / {
        / comid.class / 0 : {
```



```
            / comid.vendor / 1 : "device_vendor_AAA",
            / comid.model / 2 : "device_model_AAA",
            / comid.layer / 3 : 1
          }
        },
        / measurement-map / {
          / comid.mval / 1 : {
            / spdm-measurement-info / 65537 : {
              / spdm-version / 0 : "0x1200",
              / spdm-measurement-block-index / 1 : "0x03",
              / spdm-measurement-specification / 2 : "0x01",
              / spdm-dmtf-spec-measurement-value-type / 3 : "0x02"
            },
            / comid.digests / 2 :  [
              [
                / hash-alg-id / 8, / SHA512 /
                / hash-value /
h'ffde42483a687dd47d05f956a2d62007b71a2988084da1095ec2e43bca156680cae07d0b84cbc7fc9b1
d4e80cd8669aa956aed8bb17b0a20a5031c288dfa8b9f'
              ]
            ]
          }
        }
      ],
      [
        / environment-map / {
          / comid.class / 0 : {
            / comid.vendor / 1 : "device_vendor_AAA",
            / comid.model / 2 : "device_model_AAA",
            / comid.layer / 3 : 1
          }
        },
```



```
            / measurement-map / {
              / comid.mval / 1 : {
                / spdm-measurement-info / 65537 : {
                  / spdm-version / 0 : "0x1200",
                  / spdm-measurement-block-index / 1 : "0x04",
                  / spdm-measurement-specification / 2 : "0x01",
                  / spdm-dmtf-spec-measurement-value-type / 3 : "0x03"
                },
                / comid.digests / 2 :  [
                  [
                    / hash-alg-id / 8, / SHA512 /
                    / hash-value /
h'3a0bd5b08436b1d386122090cfa0446cf2571b74f2a15f44df735695dab84bbb1bebb3aef39af6a0f97
279b5fb04d513a52dd16547fe88d0455815520c861ed4'
                  ]
                ]
              }
            }
          ],
          [
            / environment-map / {
              / comid.class / 0 : {
                / comid.vendor / 1 : "device_vendor_AAA",
                / comid.model / 2 : "device_model_AAA",
                / comid.layer / 3 : 0
              }
            },
            / measurement-map / {
              / comid.mval / 1 : {
                / spdm-measurement-info / 65537 : {
                  / spdm-version / 0 : "0x1200",
                  / spdm-measurement-block-index / 1 : "0x10",
```



```
            / spdm-measurement-specification / 2 : "0x01",
            / spdm-dmtf-spec-measurement-value-type / 3 : "0x87"
          },
          / comid.svn / 1 :  552(7)
        }
      }
    ],
    [
      / environment-map / {
        / comid.class / 0 : {
          / comid.vendor / 1 : "device_vendor_AAA",
          / comid.model / 2 : "device_model_AAA",
          / comid.layer / 3 : 0
        }
      },
      / measurement-map / {
        / comid.mval / 1 : {
          / spdm-measurement-info / 65537 : {
            / spdm-version / 0 : "0x1200",
            / spdm-measurement-block-index / 1 : "0xFE",
            / spdm-measurement-specification / 2 : "0x01",
            / spdm-dmtf-spec-measurement-value-type / 3 : "0x85"
          },
          / raw-value-group /
          / comid.raw-value / 4 : 560(h'3F000000040000001F00000011000000')
        }
      }
    ]
  ]
}
```


Appendix B: References

[1] ISO/IEC 19770-2:2015 International Organization for Standardization/International Electrotechnical Commission, Information technology -- Software asset management -- Part 2: Software identification tag. https://www.iso.org/standard/65666.html

[2] NISTIR-8660, “Guidelines for the Creation of Interoperable Software ID (SWID) Tags”, April 2016, https://csrc.nist.gov/publications/detail/nistir/8060/final

[3] TCG, “TCG Reference Integrity Manifest (RIM) Information Model”, version 1.1 Revision 1.0, April 2024, https://trustedcomputinggroup.org/resource/tcg-reference-integrity-manifest-rim-information-model/

[4] TCG, “TCG PC Client Reference Integrity Manifest”, version 1.1 Revision 1.1, April 2024, https://trustedcomputinggroup.org/resource/tcg-pc-client-reference-integrity-manifest-specification/

[5] TCG, “TCG PC Client Platform Firmware Integrity Measurement”, Version 1.0, Revision 43, May 2021, https://trustedcomputinggroup.org/resource/tcg-pc-client-platform-firmware-integrity-measurement/

[6] TCG, “TCG Platform Certificate Profile”, Version 2.1, January 2026, https://trustedcomputinggroup.org/resource/tcg-platform-certificate-profile/

[7] TCG, “DICE Attestation Architecture”, Version 1.2, April 2025, https://trustedcomputinggroup.org/resource/dice-attestation-architecture/

[8] TCG, “TCG Trusted Attestation Protocol (TAP) Information Model for TPM families 1.2 and 2.0 and DICE family 1.0”, Version 1.0 Revision 0.36, September 2019, https://trustedcomputinggroup.org/resource/tcg-tap-information-model/

[9] IETF RFC-8949, “Concise Binary Object Representation (CBOR)”, December 2020, https://datatracker.ietf.org/doc/rfc8949/

[10] IETF RFC-8152, “CBOR Object Signing and Encryption (COSE)”, July 2017, https://datatracker.ietf.org/doc/rfc8152/

[11] IETF RFC-5280, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”, May 2008, https://datatracker.ietf.org/doc/rfc5280/

[12] IETF RFC-4122, “A Universally Unique IDentifier (UUID) URN Namespace”, July 2005, https://datatracker.ietf.org/doc/rfc4122/

[13] IETF RFC-9393, “Concise Software Identification Tags”, June 2023, https://datatracker.ietf.org/doc/rfc9393/

[14] IETF draft, “Concise Reference Integrity Manifest”, draft-ietf-rats-corim-09, October 2025, https://datatracker.ietf.org/doc/draft-ietf-rats-corim/

[15] W3C Working Group Note, "XML Signature Syntax and Processing Version 2.0", 23 July 2015, https://www.w3.org/TR/xmldsig-core2/

[16] DMTF DSP0274, “Security Protocol and Data Model (SPDM) Specification”, Version 1.4.0 May 2025, https://www.dmtf.org/dsp/DSP0274

[17] TCG, “TCG Component Reference Integrity Manifest Information Model”, Version 1.0, April 2025, https://trustedcomputinggroup.org/resource/tcg-component-reference-integrity-manifest-information-model/

[18] TCG, “TCG Component RIM Binding for SWID and CoSWID”, Version 1.0, August 2025, https://trustedcomputinggroup.org/resource/tcg-component-rim-binding-for-swid-and-coswid/

[19] TCG, “TCG Algorithm Registry”, Version 2.0, July 2025, https://trustedcomputinggroup.org/resource/tcg-algorithm-registry/

[20] IANA CBOR Tag Registry, https://www.iana.org/assignments/cbor-tags/cbor-tags.xhtml

[21] IANA CBOR Object Signing and Encryption (COSE) algorithms, https://www.iana.org/assignments/cose/cose.xhtml#algorithms

[22] IETF RFC-9052, “CBOR Object Signing and Encryption (COSE): Structures and Process”, August 2022, https://datatracker.ietf.org/doc/rfc9052/
[23] IETF RFC-9338, “CBOR Object Signing and Encryption (COSE): Countersignatures”, December 2022, https://datatracker.ietf.org/doc/rfc9338/
[24] TCG, “PC Client Platform Firmware Profile for TPM 2.0 Systems”, Version 1.06 Revision 52, https://trustedcomputinggroup.org/resource/pc-client-specific-platform-firmware-profile-specification/
[25] IEFT RFC-8610, “Concise Data Definition Language (CDDL)”, June 2019, https://datatracker.ietf.org/doc/rfc8610/
[26]NIST SP800-193 “Platform Firmware Resiliency Guidelines”, May 2018, https://csrc.nist.gov/pubs/sp/800/193/final
[27]TCG, “TCG TPM Vendor ID Registry Family 1.2 and 2.0”, Version 1.07 Revision 0.02, August 2024, https://trustedcomputinggroup.org/resource/vendor-id-registry
[28]PCI-SIG, Member Companies, https://pcisig.com/membership/member-companies
[29]USB, Valid USB Vendor ID Numbers, https://www.usb.org/developers
[30]IANA, Private Enterprise Number (PEN), https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers
[31]MIPI, MIPI Alliance Manufacturer ID Page, https://mid.mipi.org/
[32]JEDEC, Standard Manufacturers Identification Code, https://www.jedec.org/standards-documents/docs/jep-106ab

