ETSI TS 119 472-2 V1.2.1 (2026-03)

TECHNICAL SPECIFICATION

Electronic Signatures and Trust Infrastructures (ESI); Profiles for Electronic Attestation of Attributes; Part 2: Profiles for EAA/PID Presentations to Relying Party

Reference
RTS/ESI-0019472-2v121

Keywords
attribute attestation, digital identity, EUDI Wallet, trust services

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

BLUETOOTH® is a trademark registered and owned by Bluetooth SIG, Inc.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Electronic Signatures and Trust Infrastructures (ESI).

The present document is part 2 of a multi-part deliverable. Full details of the entire series can be found in part 1 [5].

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document:

1) Specifies three (3) realizations for Presentations of Electronic Attestation of Attributes (EAAP hereinafter) built on the realizations of Electronic Attestation of Attributes (EAA hereinafter), specified in ETSI TS 119 472-1 [5] namely:

   - SD-JWT VC EAAP (clause 4.1);
   - ISO/IEC-mdoc EAAP (clause 4.2); and
   - JSON-LD W3C VC EAAP (Annex A).

NOTE 1: The realization of X509-AC EAAP will be added in the next version of the present document.

2) Specifies two (2) profiles of protocols for allowing Relying Parties (RP hereinafter) to request EAAPs or Personal Identification Data (PID hereinafter) to the EUDI Wallet, and the EUDI Wallet to send the requested EAAPs/PIDs to the RP. One of the profiles supports two transmission mechanisms, namely: API-mediated and non-API mediated. The other profile only supports the non-API mediated transmission mechanisms, as shown below:

   a) A profile is built on ISO/IEC 18013-5 [10] for non-API mediated transmission mechanism only. This profile is named ISO/IEC-mdoc profile and it is defined in clause 5 of the present document.

   b) A profile is built on OpenID4VC-HAIP [11] for both API-mediated and non-API mediated transmission mechanisms, as follows:

      - Sections 5, 5.1, 5.3, 7, and 8 of [11] for transmission via Redirects or non-API mediated transmission mechanism; and
      - Sections 5, 5.2, 5.3, 7, and 8 of [11] for API mediated transmission mechanism.

      This profile is named OpenID4VC-HAIP profile and it is defined in clause 6 of the present document.

NOTE 2: The protocols for requesting and retrieving EAAPs/PIDs defined in OpenID4VC-HAIP [11] are profiles based on OpenID4 VP [7].

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found in the ETSI docbox.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

[1] W3C® Recommendation 15 May 2025: "Verifiable Credentials Data Model v2.0".

[2] IETF RFC 9901: "Selective Disclosure for JSON Web Tokens", November 2025.

[3] W3C® Recommendation 15 May 2025: "Securing Verifiable Credentials using JOSE and COSE".

[4] IETF RFC 2397: "The 'data' URL scheme", August 1988.

[5] ETSI TS 119 472-1: "Electronic Signatures and Trust Infrastructures (ESI); Profiles for Electronic Attestation of Attributes; Part 1: General requirements".

[6] IETF RFC 9101: "The OAuth 2.0 Authorization Framework: JWT-Secured Authorization Request (JAR)", August 2021.

[7] OpenID4 VP: "OpenID for Verifiable Presentations 1.0", July 2025.

[8] IETF RFC 7515: "JSON Web Signature (JWS)", May 2015.

[9] IETF RFC 7516: "JSON Web Encryption (JWE)", May 2015.

[10] ISO/IEC 18013-5: "Personal identification — ISO-compliant driving licence — Part 5: Mobile driving licence (mDL) application".

[11] OpenID4VC-HAIP: "OpenID4VC High Assurance Interoperability Profile 1.0", 29 December 2025.

[12] ETSI TS 119 612: "Electronic Signatures and Trust Infrastructures (ESI); Trusted Lists".

[13] FIDO Alliance: "Client to authenticator protocol (CTAP)", Review Draft, 21 March 2023.

[14] ETSI TS 119 475: "Electronic Signatures and Trust Infrastructures (ESI); Relying party attributes supporting EUDI Wallet user's authorization decisions".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] ETSI TS 119 471: "Electronic Signatures and Trust Infrastructures (ESI); Policy and Security requirements for Providers of Electronic Attestation of Attributes Services".

[i.2] ETSI TR 119 462: "Electronic Signatures and Trust Infrastructures (ESI); Wallet interfaces for trust services and signing".

[i.3] European Digital Identity: "Architecture and Reference Framework (ARF)" version 2.4.0.

[i.4] IETF RFC 8152: "CBOR Object Signing and Encryption (COSE)", July 2017.

[i.5] ETSI TS 119 182-1: "Electronic Signatures and Trust Infrastructures (ESI); JAdES digital signatures; Part 1: Building blocks and JAdES baseline signatures".

[i.6] ETSI TS 119 152-1: "Electronic Signatures and Trust Infrastructures (ESI); CB AdES (CBOR-AdES) digital signatures Part 1: Building blocks and CB-AdES baseline signatures".

[i.7] EUDI Wallet TS05: "Specification of common formats and API for Relying Party Registration information".

[i.8] W3C® DC API: "Digital Credentials", Working Draft.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI TS 119 471 [i.1], ETSI TS 119 472-1 [5], ETSI TR 119 462 [i.2], Architecture and Reference Framework (ARF) version 2.4.0 [i.3] and the following apply:

Electronic Attestation of Attributes Presentation (EAAP): data derived from an EAA that is presented to a specific Relying Party

NOTE: This data can be generated in such a way that it is subject-bound.

relying party registrar: entity in charge of registering the Relying Party's information necessary to allow for their electronic identification and authentication towards European Digital Identity Wallets

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

BLE	Bluetooth® Low Energy
CBOR	Concise Binary Object Representation
COSE	CBOR Object Signing and Encryption
DCQL	Digital Credentials Query Language
EAA	Electronic Attestation of Attributes
EAAP	Electronic Attestation of Attributes Presentation
EUDI	European Digital Identity
GEN	General
HAIP	High Assurance Interoperability Profile
JOSE	JSON Object Signing and Encryption
JSON	JavaScript Object Notation
JSON-LD W3C VC JOSE	JSON-LD W3C Verifiable Credentials secured with JOSE
JSON-LD W3C VC SD-JWT	JSON-LD W3C Verifiable Credentials secured with SD-JWT
JSON-LD W3C VC	JSON-LD serialized W3C Verifiable Credentials
JSON-LD W3C VP JOSE	JSON-LD W3C Verifiable Presentations secured with JOSE
JWS	JSON Web Signature
JWT	JSON Web Token
KB	Key Binding
KB-JWT	Key Binding JSON Web Token
LD	Linked Data
mDL	mobile Driving Licence
NFC	Near Field Communication
OIDFVP	OpenID For Verifiable Presentations
PID	Personal Identification Data
REQ	Request
RO	Request Object
RP	Relying Party
SD	Selective Disclosure
SD-JWT VC	Selective Disclosure based JSON Web Token Verifiable Credentials
SD-JWT	Selective Disclosure based on JSON Web Token
SD-JWT+KB	SD-JWT with a Key Binding JWT
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USB	Universal Serial Bus
VP	Verifiable Presentation

W3C VC	W3C Verifiable Credentials
WU	Wallet Unit
X509-AC	X.509 Attribute Certificate

3.4 Notation

The present document assigns one identifier for each requirement.

These identifiers result from the concatenation of the following two components:

1) A topic identifier, for signalling the topic targeted by the requirement.

2) A number of 2 digits. For each topic the number will start in 01 and it will increase in one unity for each requirement.

The following topic identifiers are used in the body part of the present document:

1) EAAP-SD-JWT VC for requirements on SD-JWT VC EAAPs (clause 4.1 of the present document).

2) EAAP-ISO/IEC-mdoc for requirements on ISO/IEC-mdoc EAAPs (clause 4.2 of the present document).

3) ISO/IEC 18013-SUPPORT for requirements on support by EUDI Wallet and RPs of the ISO/IEC-mdoc profile (clause 5.2 of the present document).

4) ISO/IEC 18013-GEN for common general requirements of the ISO/IEC-mdoc profile (clause 5.3.1 of the present document).

5) ISO/IEC 18013-REQ for common requirements on the EAAP Request of the ISO/IEC-mdoc profile (clause 5.3.2 of the present document).

6) ISO/IEC 18013-RESP for requirements on the EAAP Response of the ISO/IEC-mdoc profile (clause 5.3.3 of the present document).

7) OIDFVP-HAIP-SUPPORT for requirements on support by EUDI Wallet and RPs of the profile built on [11] (clause 6.2 of the present document).

8) OIDFVP-HAIP-GEN for common general requirements of the profile built on [11] (clause 6.3.1 of the present document).

9) OIDFVP-HAIP-COMMON-REQ for common requirements for EAAP Requests of the profile built on [11] (clause 6.3.2.1 of the present document).

10) OIDFVP-HAIP-COMMON-REQ-RO for common requirements for Requests Objects of the profile built on [11] (clause 6.3.2.2 of the present document).

11) OIDFVP-HAIP-COMMON-RESP for common requirements for EAAP Responses of the profile built on [11] (clause 6.3.3 of the present document).

12) OIDFVP-HAIP-REDIRECTS for specific requirements for non-API mediated transmission mechanism for the profile built on [11] (clause 6.4.1 of the present document).

13) OIDFVP-HAIP-REDIRECTS-RO for specific requirements for the Request Object in non-API mediated transmission mechanism for the profile built on [11] (clause 6.4.2 of the present document).

14) OIDFVP-HAIP-API for specific requirements for API-mediated transmission mechanism for the profile built on [11] (clause 6.5.1 of the present document).

15) OIDFVP-HAIP-ADD-API for additional requirements for API-mediated mechanism in the profile built on OpenID4VC-HAIP [11] (clause 6.5.2 of the present document).

16) EAAP-JSON-LD W3C VC JOSE for requirements on JSON-LD W3C VC EAAP (Annex A of the present document).

17) OIDFVP-HAIP-JSON_LD_EAA-GEN-REQ for requirements on JSON-LD W3C VC EAAP (Annex C of the present document).

18) X509_AC_EAA-GEN-REQ for requirements for requesting presentation of X509-AC EAAP (Annex D of the present document).

4 Implementation of Electronic Attestation of Attributes Presentations

4.1 EAAP implementation based on SD-JWT VC

The present clause specifies a realization of EAAP for the SD-JWT VC EAA defined in clause 5 of ETSI TS 119 472-1 [5].

The EAAPs implemented according to the present clause will be designated as SD-JWT VC EAAP hereinafter.

EAAP-SD-JWT VC-01: If the SD-JWT VC EAA contains the `cnf` claim, the corresponding SD-JWT VC EAAP shall be a SD-JWT+KB as specified in IETF SD-JWT [2].

EAAP-SD-JWT VC-02: If the SD-JWT VC EAA does not contain the `cnf` claim, the corresponding SD-JWT VC EAAP shall be a SD-JWT VC EAAP as specified in IETF SD-JWT [2].

NOTE: If the SD-JWT VC EAA does not contain the `cnf` claim, the EAA subject binding, if needed, can be ensured by other means (e.g. claims-based binding, biometric-based binding), as specified by the EAA Provider or the rulebook.

EAAP-SD-JWT VC-03: A SD-JWT VC EAAP shall be serialized using either the Compact Serialization, as specified in clause 5.2 of IETF SD-JWT [2], or the Flattened JSON Serialization, as specified in clause 8.2 of IETF SD-JWT [2].

EAAP-SD-JWT VC-04: The Key Binding JSON Web Token (KB-JWT) of a SD-JWT VC EAAP (which is a SD-JWT+KB) shall be signed by the EAA subject.

4.2 EAAP implementation based on ISO/IEC-mdoc

The present clause specifies a realization of EAAPs for the ISO/IEC-mdoc EAAs defined in clause 6 of ETSI TS 119 472-1 [5].

NOTE 1: Clause 6 of ETSI TS 119 472-1 [5] defines different requirements for ISO/IEC-mdoc EAAs that are mobile Driving Licenses (mDL) and ISO/IEC-mdoc EAAs that are NOT mDLs in terms of data elements and their namespaces. See ETSI TS 119 472-1 for more details.

The EAAPs implemented according to the present clause will be designated as ISO/IEC-mdoc EAAP hereinafter.

ISO/IEC 18013-5 [10] requires that the mdoc (the EUDI Wallet) builds an instance of type `DeviceResponse` in response to a correct instance of type `DeviceRequest` sent by the mdoc reader/verifier (a Relying Party).

This instance of `DeviceResponse` type can contain the `documents` member, which is an array of instances of type `Document`.

Each element in this array can contain an indication of error if the request of that element was not correctly built or any other problem has occurred during the processing of the request by the mdoc or during the generation of the corresponding document.

EAAP-ISO/IEC-mdoc-01: Each element in the `documents` member of an instance of type `DeviceResponse` as defined in clause 10.3.3 of ISO/IEC 18013-5 [10] shall be an ISO/IEC-mdoc EAAP if the mentioned element does not contain the `errors` member.

NOTE 2: As each element in the `documents` member of an instance of type `DeviceResponse` is of type `Document`, an ISO/IEC-mdoc EAAP is an instance of type `Document` that does not contain the `errors` member.

EAAP-ISO/IEC-mdoc-02: The `issuerSigned` member of the `DeviceResponse` message shall contain all the disclosed attributes to be presented that have been signed by the EAA/PID Provider.

EAAP-ISO/IEC-mdoc-03: The `deviceSigned` member of the `DeviceResponse` message may contain attributes not present in the `issuerSigned` member if and only if their inclusion has been explicitly authorized by the EAA/PID Provider.

NOTE 3: The `deviceSigned` member is mainly devoted to contain transaction specific data.

5 ISO/IEC-mdoc profile

5.1 Introduction

Clause 5 and its subclauses define a profile for a protocol allowing a RP to request EAAPs or PIDs to the EUDI Wallet, and the EUDI Wallet to send the requested EAAPs/PIDs to the RP using a non-API mediated transmission mechanism, built on ISO/IEC 18013-5 [10].

The rest of clause 5 is organized as follows:

- Clause 5.2 defines requirements on support of the profile and transmission mechanisms by RPs and EUDI Wallet.
- Clause 5.3 and its subclauses specify the requirements for the protocol.

5.2 Requirements on EUDI Wallet and RP support

ISO/IEC 18013-SUPPORT-01: Wallet Units, PID Providers, Attestation Providers, Wallet Providers, and Relying Parties shall not support server retrieval as specified in ISO/IEC 18013-5 [10] for requesting and presenting PID or attestation attributes.

ISO/IEC 18013-SUPPORT-02: The EUDI Wallet shall meet the requirements defined in clause 5.3.

ISO/IEC 18013-SUPPORT-03: A Relying Party shall meet the requirements defined in clause 5.3.

5.3 Protocol requirements

5.3.1 General requirements

ISO/IEC 18013-GEN-01: All the mandatory requirements for device retrieval data structures defined in [10] shall apply.

ISO/IEC 18013-GEN-02: All the optional requirements for device retrieval data structures defined in [10] shall remain optional unless stated otherwise in the present document.

ISO/IEC 18013-GEN-03: If the present document modifies a requirement in [10], the modified requirement defined by the present document shall prevail.

5.3.2 ISO/IEC-mdoc EAAP Request contents

The present clause defines requirements for the content of the EAAP Request that have to be sent from the RP to the EUDI Wallet.

ISO/IEC 18013-REQ-01: All the elements of the `docRequests` array shall contain the `readerAuth` member.

NOTE 1: ISO/IEC 18013-5 [10] defines `readerAuth` member as an instance of `ReaderAuth` type, which makes equal to `COSE_Sign1` type defined in IETF RFC 8152 [i.4]. Therefore, `readerAuth` is a digital signature.

ISO/IEC 18013-REQ-02: The digital signature implemented in `readerAuth` shall be generated with the private key whose corresponding public key is enclosed within the RP access certificate.

ISO/IEC 18013-REQ-03: The `x5chain` unprotected header parameter shall contain the RP access certificate in its first element, and its certificate path up to, but excluding, the trust anchor.

ISO/IEC 18013-REQ-04: The instances of type `ItemsRequest`, encapsulated in the elements in the `docRequests` member, shall contain a non-empty `requestInfo` member of type `RequestInfo`.

NOTE 2: All the elements in the `docRequests` member are instances of type `ItemsRequest` encapsulated in a CBOR byte string (type `ItemsRequestBytes`).

ISO/IEC 18013-REQ-05: The `RequestInfo` type shall be as the CDDL definition shows below.

```
RequestInfo = {

   ?"eUWrprc": bstr,                          ; contains a registration certificate if
                                              ; available (see requirements below)
   "euWrpRegistrarInfo": EUWrpRegistrarInfo   ; RP's Registrar-provided information (see
                                              ; requirements below.
}
```

ISO/IEC 18013-REQ-06: If the RP has a registration certificate, the mentioned `requestInfo` member shall contain a member with label "`euWrprc`".

ISO/IEC 18013-REQ-07: The member with label "`euWrprc`" shall map the label "`euWrprc`" to a CBOR byte string.

ISO/IEC 18013-REQ-08: The CBOR byte string mapped to the "`euWrprc`" label shall contain the serialization of the RP registration certificate.

NOTE 3: This new member is required for incorporating the RP registration certificate(s), which are placed in an extension of the `requestInfo`.

ISO/IEC 18013-REQ-09: The mentioned `requestInfo` member shall contain a member with label "`euWrpRegistrarInfo`".

ISO/IEC 18013-REQ-10: The member with label "`euWrpRegistrarInfo`" shall map the label "`euWrpRegistrarInfo`" to a CBOR map, which shall be an instance of the `EUWrpRegistrarInfo` type.

ISO/IEC 18013-REQ-11: The `EUWrpRegistrarInfo` type shall be as the CDDL definition shows below.

```
EUWrpRegistrarInfo = {

   "identifier": [+Identifier],           ; RP's identifiers as defined in
                                          ; Annex B2.2 of [14]
   "srvDescription": ServiceDescription,  ; Description of the RP's service as defined in
                                          ; Annex B.2.1 of [14]
   "registryURI": tstr,                   ; URI of the RP's Registrar API as defined in
                                          ; Annex B.2.1 of [14]
   "intendedUseIdentifier": tstr          ; Registrar-provided identifier of the RP's
                                          ; intended use, as defined in
                                          ; Annex B.2.7 of [14]
   "purpose": Purposes,                   ; Array of multilanguage strings containing the
                                          ; registered purposes of the intended data
                                          ; processing, as defined in Annex B.2.6 of [14]
   "policyURI": tstr,                     ; URI of the privacy policy of the registered
                                          ; intended use, as defined in Annex B.2.8 of
                                          ; [14]
   ?"credential": [+Credential]           ; Array of credential objects, containing the
                                          ; set of attestations and set of attributes per
                                          ; attestation registered in the context of the
                                          ; intended use, as defined in Annex B.2.9 of
                                          ; [14]
}

Identifier = {
```

```
   "type": tstr,        ; For possible values, see Annex B.2.5 of [14]
   "identifier": tstr   ; The identifier which identifies the RP.
}

ServiceDescription = [+ MultiLangString]  ; 1 or more localized text
                                          ; describing the same service

Purposes = [+ MultiLangString]  ; 1 or more localized text describing the registered
                                ; purposes of the intended data processing

MultiLangString = {
   "lang": tstr,     ; The country code of the localized text. A two-letter Alpha-2
                     ; language code according to ISO 639 [15](Set 1).
   "content": tstr   ; The localized text as a string.
}

Credential = {
   "format": tstr,      ; Format of the attestation as defined in Annex B.2.9 of [14]
   "meta": tstr,        ; Object defining additional properties as defined in Annex B.2.9
                        ; of [14]
   ?"claim": [+Claim]   ; Array of objects specifying the attributes in the requested
                        ; attestation, as defined in Annex B.2.10 of [14]. If not
                        ; available, all attributes are requested.
}

Claim = {
   "path": tstr,                    ; a path pointer that specifies the path to a claim
                                    ; within the Credential as defined in Annex B.2.10
                                    ; of [14]
   ?"values": [+(tstr / int / bool)] ; Array of strings, integers or boolean
                                    ; values that specifies the expected values
                                    ; of the claim as defined in Annex B.2.10 of [14]
}
```

NOTE 4: The information in `EUwrpRegistrarInfo` is required by the Wallet Unit to be able to verify the RP's registration information for the presentation request's intended use, and present this information to the User. Note that in case the RP uses the services of an intermediary, the identifier in the element mapped to `"identifier"` label will be different from the name and identifier of the intermediary as included in the intermediary's access certificate.

5.3.3 ISO/IEC-mdoc EAAP Response profile

The present clause defines requirements for the `DeviceResponse` message type.

ISO/IEC 18013-RESP-01: In response to a `DeviceRequest` message sent by the RP, the WU shall generate a `DeviceResponse` message as specified in ISO/IEC 18013-5 [10].

6 Profile built on OpenID4VC-HAIP

6.1 Introduction

Clause 5 and its subclauses define a profile for a protocol allowing a RP to request EAAPs or PIDs to the EUDI Wallet, and the EUDI Wallet to send the requested EAAPs/PIDs to the RP using two mechanisms of transmission for requests and responses, namely:

1) A non-API mediated transmission mechanism, built on OpenID4VC-HAIP [11], sections 5, 5.1, 5.3, 7 and 8; and

2) An API-mediated transmission mechanism built on OpenID4VC-HAIP [11], sections 5, 5.2, 5.3, 7 and 8.

The rest of clause 6 is organized as follows:

- Clause 6.2 defines requirements on support of the profile and transmission mechanisms by RPs and EUDI Wallet.
- Clause 6.3 and its subclauses specify common requirements for both transmission mechanisms.
- Clause 6.4 specifies requirements that are specific for the non-API mediated transmission mechanism.
- Clause 6.5 specifies requirements that are specific for the API-mediated transmission mechanism.
- Clause 6.6 deals with security considerations for this profile.

NOTE: For the purposes of the present document, section 5 or requirements of section 5 of OpenID4VC-HAIP [11] refers only to the general requirements listed in the introductory text of section 5 and does not extend to subsections 5.1 onwards.

6.2 Requirements on EUDI Wallet and RP support

OIDFVP-HAIP-SUPPORT-01: The EUDI Wallet shall meet the requirements defined in clauses 6.3 and 6.4 of the present document.

NOTE 1: The former requirement implies that the EUDI Wallet supports the non-API mediated mechanism for presentations flows built on OpenID4VC-HAIP [11].

OIDFVP-HAIP-SUPPORT-02: The EUDI Wallet may meet the requirements defined in clause 6.5 of the present document.

NOTE 2: The former requirement corresponds to the API-mediated mechanism for presentations flows specified in clause 5.2 of OpenID4VC-HAIP [11].

OIDFVP-HAIP-SUPPORT-03: The EUDI Wallet shall support the cross-device flow using only the API-mediated mechanism.

OIDFVP-HAIP-SUPPORT-04: A Relying Party shall meet the requirements defined in clauses 6.3 and 6.4 of the present document.

NOTE 3: The former requirement implies that a Relying Party supports the non-API mediated mechanism for presentations flows specified in clause 5.1 of OpenID4VC-HAIP [11].

OIDFVP-HAIP-SUPPORT-05: A Relying Party may meet the requirements defined in clauses 6.5 of the present document.

NOTE 4: The former requirement corresponds to the API-mediated mechanism for presentations flows specified in clause 5.2 of OpenID4VC-HAIP [11].

NOTE 5: Requirements OIDFVP-HAIP-SUPPORT-02 and OIDFVP-HAIP-SUPPORT-05 (support of API-mediated transmission mechanisms by the EUDI Wallet and RPs) could change in the future depending on the evolution of W3C DC API [i.8].

6.3 Common requirements for all the transmission mechanisms

6.3.1 General requirements

OIDFVP-HAIP-GEN-01: All the mandatory requirements defined in clauses 5, 5.3, 7 and 8 of HAIP [11] shall apply.

OIDFVP-HAIP-GEN-02: All the optional requirements defined in clauses 5, 5.3, 7 and 8 of HAIP [11] shall remain optional unless stated otherwise in the present document.

OIDFVP-HAIP-GEN-03: If the present document modifies a requirement in OpenID4VC-HAIP [11], the modified requirement defined by the present document shall prevail.

NOTE: This would allow, for instance, to convert in mandatory an optional requirement from OpenID4VC-HAIP [11] or to extend mandatory requirements.

6.3.2 Authorization Request (EAAP request) profile

6.3.2.1 General requirements

OIDFVP-HAIP-COMMON-REQ-01: The Authorization Request shall use the Client Identifier Prefix `x509_hash`.

6.3.2.2 Requirements for the Request Object

The present clause defines requirements for the Request Object regardless the transmission mechanism used.

NOTE 1: IETF RFC 9101 [6] mandates that the Request Object contains "all the parameters (including extension parameters) used to process the OAuth 2.0 (IETF RFC 6749) authorization request" except the `request` and `request_uri` parameters that are defined in IETF RFC 9101 [6].

OIDFVP-HAIP-COMMON-REQ-RO-01: The RO JWT body shall contain the `verifier_info` parameter.

OIDFVP-HAIP-COMMON-REQ-RO-02: The `verifier_info` parameter shall contain RP Registrar-provided data.

OIDFVP-HAIP-COMMON-REQ-RO-03: The element in the `verifier_info` array enclosing the RP Registrar-provided data shall be a JSON Object which shall not contain the `credential_ids` member.

OIDFVP-HAIP-COMMON-REQ-RO-04: The value of the `format` member of the element in the `verifier_info` array enclosing the RP Registrar-provided data shall be: `"registrar_dataset"`.

OIDFVP-HAIP-COMMON-REQ-RO-05: The value of the `data` member of the element in the `verifier_info` array parameter that contains the RP Registrar-provided data shall be a non-empty JSON Object.

The `data` member of the element in the `verifier_info` array parameter that contains the RP Registrar-provided data:

- OIDFVP-HAIP-COMMON-REQ-RO-06: Shall contain the `identifier` member specified in clause B.2.2 of [14], whose value shall be the RP's identifier.
- OIDFVP-HAIP-COMMON-REQ-RO-07: Shall contain the `srvDescription` member specified in clause B.2.2 of [14], which shall be an array of JSON MultiLangString objects (defined in clause B.2.6 of [14]), whose contents shall be registered user-friendly descriptions of the RP's service.
- OIDFVP-HAIP-COMMON-REQ-RO-08: Shall contain the `registryURI` member specified in clause B.2.1 of [14], whose value shall be the URI of the RP Registrar's API.
- OIDFVP-HAIP-COMMON-REQ-RO-09: Shall contain the `intendedUseIdentifier` member specified in clause B.2.7 of [14], which contains the RP Registrar-provided identifier of the RP's intended use.
- OIDFVP-HAIP-COMMON-REQ-RO-10: Shall contain the `purpose` member specified in clause B.2.7 of [14], which shall be an array of JSON MultiLangString objects (defined in clause B.2.6 of [14]), whose contents shall be the registered purposes of the intended data processing.
- OIDFVP-HAIP-COMMON-REQ-RO-11: Shall contain the `policyURI` member specified in clause B.2.8 of [14], which shall contain the privacy policy URI of the registered intended use.
- OIDFVP-HAIP-COMMON-REQ-RO-12: May contain the `credential` member specified in clause B.2.7 of [14], which shall be an Array of Credential objects (defined in clause B.2.9 of [14]), whose contents shall contain the set of attestations and set of attributes per attestation registered in the context of the intended use.

NOTE 2: According to EUDI Wallet ARF [i.3], if the User indicated after being presented the presentation request, or as a general User setting, that they want to verify the information registered about the Relying Party and its intended use, but the Relying Party did not send a registration certificate to the EUDI Wallet, it will connect to the URL of the API of the Relying Party Registrar (see EUDI Wallet TS05 [i.7]) to obtain and verify this information. Information necessary for the presentation on the UI includes the user-friendly name and unique identifier of the Relying Party and a user-friendly description of the intended use of the Relying Party, including the list of requested attestations and their attributes, and the URI of the applying privacy policy (for purposes of informing User). The URL to the API of the Registrar of the Relying Party, the unique identifier of the Relying Party and the identifier of the intended use of the Relying Party are needed for the API query that is used to verify the validity of presented data from the Registrar of the Relying Party. Note that in case the RP uses the services of an intermediary, the identifier of requirement OIDFVP-HAIP-COMMON-REQ-RO-06 will be different from the identifier of the intermediary as included in the intermediary's access certificate.

OIDFVP-HAIP-COMMON-REQ-RO-13: If the RP has a registration certificate, one of the elements of the `verifier_info` parameter shall include it.

OIDFVP-HAIP-COMMON-REQ-RO-14: The element in the `verifier_info` array enclosing the registration certificate shall be a JSON Object which shall not contain the `credential_ids` member.

OIDFVP-HAIP-COMMON-REQ-RO-15: The value of the `format` member of the element in the `verifier_info` array enclosing the registration certificate shall be: "`registration_cert`".

OIDFVP-HAIP-COMMON-REQ-RO-16: The value of the `data` member of the element in the `verifier_info` array enclosing the registration certificate shall be the base64url encoding of the serialized RP registration certificate.

OIDFVP-HAIP-COMMON-REQ-RO-17: The Authority Key Identifier shall use the ETSI trusted Lists (specified in ETSI TS 119 612 [12]) mechanism (`"etsi_tl"` type).

OIDFVP-HAIP-COMMON-REQ-RO-18: The RO JWT body shall contain the `client_metadata` parameter.

OIDFVP-HAIP-COMMON-REQ-RO-19: The `client_metadata` parameter shall contain the `jwks` member.

OIDFVP-HAIP-COMMON-REQ-RO-20: The `jwks` member of the `client_metadata` parameter shall contain the `kid` and `use` parameters for identifying the key and the use of the identified key, respectively.

OIDFVP-HAIP-COMMON-REQ-RO-21: The `kid` parameters shall univocally identify one key.

NOTE 3: Clause 6.2 specifies the key types and the algorithms for the present profile.

OIDFVP-HAIP-COMMON-REQ-RO-22: The RO JWT body shall contain the `aud` parameter.

OIDFVP-HAIP-COMMON-REQ-RO-23: The RO shall be signed by the RP using the private key whose corresponding public key is enclosed within the RP access certificate.

6.3.3 Authorization Response (EAAP response) profile

OIDFVP-HAIP-COMMON-RESP-01: The EUDI Wallet shall encrypt the authorization response.

6.4 Specific requirements for non-API mediated transmission mechanism

6.4.1 General requirements

OIDFVP-HAIP-REDIRECTS-01: All the mandatory requirements defined in clause 5.1 of HAIP [11] shall apply.

OIDFVP-HAIP-REDIRECTS-02: All the optional requirements defined in clause 5.1 of HAIP [11] shall remain optional unless stated otherwise in the present clause.

OIDFVP-HAIP-REDIRECTS-03: The EUDI Wallet shall support at least a custom URL scheme "`eu-eaap://`" for its `authorization_endpoint`.

OIDFVP-HAIP-REDIRECTS-04: The Authorization Request shall contain the `request_uri` parameter, and therefore shall not contain the Request Object (RO).

NOTE: With this mechanism of transmission, the Request Object is passed by reference to the EUDI Wallet.

6.4.2 Requirements for the Request Object

The present clause defines requirements for the Request Object for the non-API mediated transmission mechanism.

OIDFVP-HAIP-REDIRECTS_RO-01: The JWS Protected Header of the JWS signature on the RO shall incorporate the `x5c` header parameter.

OIDFVP-HAIP-REDIRECTS_RO-02: The `x5c` header parameter in the JWS Protected Header of the JWS signature on the RO shall contain the RP access certificate in its first element, and its certificate path up to, but excluding, the trust anchor.

OIDFVP-HAIP-REDIRECTS_RO-03: The JWS Protected Header of the JWS signature on the RO shall incorporate the `iat` header parameter.

6.5 Specific requirements for API-mediated transmission mechanism

6.5.1 OpenID4VC-HAIP-related requirements

OIDFVP-HAIP-API-01: All the mandatory requirements defined in clause 5.2 of HAIP [11] shall apply.

OIDFVP-HAIP-API-02: All the optional requirements defined in clause 5.2 of HAIP of HAIP [11] shall remain optional unless stated otherwise in the present clause.

OIDFVP-HAIP-API-03: The EUDI Wallet shall not support Authorization Requests with exchange protocol value set to `"openid4vp-1-unsigned"`.

6.5.2 Additional requirements

OIDFVP-HAIP-ADD-API-01: The EUDI Wallet shall by default disclose the presence of all stored EAAs' type to the mediating API that works in accordance with clause 5.2 of [11], but it shall not disclose the attributes and their values in these EAAs.

NOTE 1: The attribute value restriction applies even if such disclosure would enhance the services provided by the Operating System to the EUDI Wallet, for example, attestation selection in the context of the mediating API.

There are the considerations related to operating systems and browsers which fall out of the control of implementers, which can be taken into account as indicated in the following notes 2 to 4:

NOTE 2: A presentation request from a Relying Party supporting clause 5.2 of [11] can be processed by the browser and/or the Operating System for searching available EAAs, for preventing fraud targeting the User, or for troubleshooting purposes.

NOTE 3: A presentation request from a Relying Party supporting clause 5.2 of [11] is expected to be processed by the browser and/or the Operating System for User security purposes.

NOTE 4: A presentation request from a Relying Party supporting clause 5.2 of [11] is expected not to be processed by the browser and/or the Operating System for market analysis purposes (including as a secondary purpose) or for the browser's and/or the Operating System's internal purposes.

OIDFVP-HAIP-ADD-API-02: If an EUDI Wallet deletes on the User's request a PID or EAA previously disclosed to the mediating API that works in accordance with clause 5.2 of [11], the EUDI Wallet shall disclose the fact that it no longer stores this PID or EAA to the mediating API.

OIDFVP-HAIP-ADD-API-03: If the User uninstalls their EUDI Wallet, the EUDI Wallet shall disclose the fact that it no longer stores any previously disclosed PID(s) or EAA(s) to the mediating API that works in accordance with clause 5.2 of [11].

OIDFVP-HAIP-ADD-API-04: The EUDI Wallet shall provide a global user setting to disable the disclosure of stored EAAs via a mediating API that works as is defined in OIDFVP-HAIP-ADD-API-01. When this setting is disabled, the EUDI Wallet shall not advertise or respond to the API-mediated presentation or issuance requests.

OIDFVP-HAIP-ADD-API-05: Where supported by the browser and Operating System of the device of the EUDI Wallet, the EUDI Wallet shall use the CTAP 2.2 hybrid flow as defined in section 11.5 of [13] for cross-device communication between a browser and the device that the EUDI Wallet is installed in.

NOTE 5: This flow establishes a secure tunnel between the browser and the device, which is then used to exchange the presentation request and the corresponding response.

OIDFVP-HAIP-ADD-API-06: The browsers and Operating Systems shall use short-range CTAP transport bindings to transfer CTAP2 messages over BLE, NFC or USB, whichever applies.

NOTE 6: The requirements defined in the present clause are the requirements defined in clause 5.5.2 applied to a mediating API that works in accordance with clause 5.2 of [11].

6.6 Security considerations

The security considerations in clause 14 of OpenID4 VP [7] apply.

Annex A (normative): EAAP implementation based on JSON-LD W3C VC

A.1 Introduction

The present annex specifies requirements for generating EAAPs for the JSON-LD W3C VC EAA specified in clause 7 of ETSI TS 119 472-1 [5].

These EAAPs shall be generated as specified in W3C Recommendation (15 May 2025): [3]. This W3C Recommendation defines how to secure JSON-LD W3C Verifiable Credentials and JSON-LD W3C Verifiable Presentations either with JWS (the W3C Recommendation uses JOSE), or with SD-JWT.

The EAAPs specified in clause A.2 of the present document use JWS signatures as specified in IETF RFC 7515 [8], for generating EAAPs of both JSON-LD W3C VC JOSE EAAs and JSON-LD W3C VC SD-JWT EAAs. These EAAPs will be designated as JSON-LD W3C VP JOSE EAAPs.

A.2 JOSE-signed JSON-LD W3C EAAPs for JSON-LD W3C EAAs

EAAP-JSON-LD W3C VC JOSE-01: A JSON-LD W3C VC JOSE EAAP shall meet the requirements defined in: clause 4.13 of [1], clause 3.1.2 of W3C Recommendation (15 May 2025): [3], and the requirements defined in the present clause.

EAAP-JSON-LD W3C VC JOSE-02: A JSON-LD W3C VC JOSE EAAP shall be a JWS signature generated by the EAA subject.

EAAP-JSON-LD W3C VC JOSE-03: The payload of a JSON-LD W3C VC JOSE EAAP shall be an object meeting the requirements defined in clause 4.13 of "Verifiable Credentials Data Model v2.0" [1].

EAAP-JSON-LD W3C VC JOSE-04: The payload of a JSON-LD W3C VC JOSE EAAP shall have the `verifiableCredential` property.

NOTE 1: The `verifiableCredential` property is defined in clause 4.13 of [1] as an array. Each element of the `verifiableCredential` array encapsulates either a sequence of one or more JSON-LD W3C VC JOSE EAAs or a sequence of one or more JSON-LD W3C VC SD-JWT EAA as specified in clause 7 of ETSI TS 119 472-1 [5].

EAAP-JSON-LD W3C VC JOSE-05: Each element in the `verifiableCredential` array shall have the `type`, `@context`, and `id` properties.

EAAP-JSON-LD W3C VC JOSE-06: The `type` property child of each element in the `verifiableCredential` array shall have the value `EnvelopedVerifiableCredential`.

NOTE 2: This value signals that each element in the `verifiableCredential` array contains signed JSON-LD W3C VC JOSE EAAs and/or JSON-LD W3C VC SD-JWT EAAs as specified in clause 7 of ETSI TS 119 472-1 [5].

EAAP-JSON-LD W3C VC JOSE-07: The `id` property child of each element in the `verifiableCredential` array shall contain one or more data URIs as specified in IETF RFC 2397 [4].

EAAP-JSON-LD W3C VC JOSE-08: Each data URI within the `id` property shall be separated from the next one by the ';' character.

EAAP-JSON-LD W3C VC JOSE-09: If the URI encapsulates a sequence of one or more JSON-LD W3C VC JOSE EAAs then the media type of the `id` property shall be `application/vc+jwt`.

NOTE 3: These values declare that the data URL encapsulates a sequence of one or more JSON-LD W3C VC JOSE EAAs.

EAAP-JSON-LD W3C VC JOSE-10: If the URI encapsulates a sequence of one or more JSON-LD W3C VC SD-JWT EAAs then the media type of the `id` property shall be `application/vc+sd-jwt`.

NOTE 4: These values declare that the data URL encapsulates a sequence of one or more JSON-LD W3C VC SD-JWT EAAs.

EAAP-JSON-LD W3C VC JOSE-11: All the signed JSON-LD W3C VC EAAs in the data part of the data URL of the `id` property shall use the same Serialization, either the Compact Serialization or the base64 encoding of the Flattened JSON.

EAAP-JSON-LD W3C VC JOSE-12: If all the signed JSON-LD W3C VC EAAs in the data part of the data URL of the `id` property are base64 encoding of the Flattened JSON Serialization mentioned before, the string "`;base64,`" shall be inserted between the value of the media type and the data part.

NOTE 5: As a consequence of the former requirements JSON-LD W3C VC EAAs using Compact Serialization and JSON-LD W3C VC EAAs using Flattened JSON Serialization, are placed in different elements of the `verifiableCredential` array.

NOTE 6: Therefore, each object in the `verifiableCredential` array encapsulates a sequence of one or more JSON-LD W3C VC JOSE EAAs or a sequence of one or more JSON-LD W3C VC SD-JWT EAAs as specified in clause 7 of ETSI TS 119 472-1 [5].

Annex B (informative): EAAP implementation based on X.509 Attribute Certificates (X509-AC)

NOTE: To be completed in later versions. A possible solution would be to use an enveloping JAdES signature, specified in ETSI TS 119 182 [i.5], with the `srAttrs` header parameter, enclosing the `certified` member containing an array of X.509 Attribute certificates. Also other solutions need to be investigated.

Annex C (normative): Requirements for requesting presentation of JSON-LD W3C-VC EAAs

C.1 Introduction

The present annex defines requirements for requesting and retrieving JSON-LD W3C-VC EAAP using the profiles based on HAIP defined in the present document.

C.2 Authorization Request

OIDFVP-HAIP-JSON_LD_EAA-GEN-REQ-01: For requesting a JSON-LD W3C-VC EAA presentation as specified in Annex A of the present document, the `format` claim within the `dcql_query` shall have the value "`vp+jwt`".

Annex D (normative): Requirements for requesting presentation of X509-AC EAAs

D.1 Introduction

The present annex defines requirements for requesting and retrieving X509-AC EAA Presentations using the profiles based on HAIP defined in the present document.

D.2 Authorization Request

X509_AC_EAA-GEN-REQ-01: For requesting a X509-AC EAAP as specified in Annex B of the present document, the `format` claim within the `dcql_query` shall have the value "`x509_attr`".

Annex E (informative): Change history

Date	Version	Information about changes
28/12/2025	V1.1.2	Major changes performed based on comments on v1.1.1. Below follow details: 1) Major re-structuring of the material due to the incorporation of API-mediated transmission mechanisms for both profiles ISO/IEC-mdoc (built on ISO/IEC 18013) and HAIP. Now the structure for each profile contains: • Requirements on EUDI Wallet and RP support • Common requirements for all the transmission mechanisms • Specific requirements for non-API mediated transmission mechanisms. • Specific requirements for API-mediated transmission mechanisms 2) ISO/IEC-mdoc profile: • Deleted a number of wrong set of HAIP requirements that had been placed in clauses defining the profile built on ISO/IEC 18013). This was clause 5.2 of v1.1.1. • EAAP Request: Definition of new member requestInfo with RP's Registrar-provided information and optionally the registration certificate of the RP. • EAAP Response: o deviceSigned may now contain attributes IF the EAA Provider has explicitly authorized them. This is for covering certain use cases apparently requested by Germany (self-issued EAAs). o Suppressed mention to algorithms (wrong algorithms taken from HAIP): ISO specifies the usable algorithms. o Suppressed requirement of deviceAuth containing only deviceSignature (ISO also allows MAC: deviceMac allowed). • Dropped requirements that are repetition of ISO requirements • Added specific requirements for API-mediated transmission mechanism 3) HAIP profile. • Dropped requirement mandating that EUDI Wallet supports both A128GCM and A256GCM, as this could leave out some implementations. Note that RPs are mandated to support both algorithms. • Kept details only on how to requests presentations of EAA built on SD-JWT VC and ISO/IEC-mdoc. The details on how to requests presentations of EAA built on X.509-AC and JSON-LD W3C VC, moved to Annexes A and B. • Make verifier_info mandatory in RO JWT body. • verifier_info shall contain registrar-provided data. • If registration certificate available, then verifier_info shall contain it • Defined details for the element containing registrar-provided data. NO trade_name: assumed concerns mentioned for ETSI TS 119 472-3 also apply here. • Dropped requirements that repeat requirements either in HAIP or in OID4VP • Dropped requirements on iat and aud in RO. • Not mandatory that EAAPs are signed • Added specific requirements for API-mediated transmission mechanism
16/1/2026	V1.1.3	Implemented dispositions to comments to v1.1.2 agreed at the two ad hoc calls held on January 2026. See clause "Comments raised by EC experts team and iDAKTO" of document https://docbox.etsi.org/ESI/ESI/05-CONTRIBUTIONS/2026/ESI(26)000030r1_ESI_88a_Disp2Comms_iDAKTO_EC2_TS_119472-2v1_1_2.doc the dispositions implemented in this version.

Date	Version	Information about changes
30/1/2026	V1.1.4	Implemented dispositions to comments to v1.1.3 agreed at ESI#88a meeting. • Dropped the API-mediated mechanism from the ISO-mdoc profile. • Make support by Wallet and RPs of API-mediated mechanism in HAIP, optional. • Add a note reporting that this can change in the future depending on the evolution of DC-API. • Add an informative reference to DC-API. • Make optional the presence of credential parameter in the RO (OpenID4VC-HAIP profile). • Keep in clause 6.3.2.2 requirements that are not imposed by neither OpenID4VC nor by HAIP. • Add a note explaining that JAR actually mandates that all the parameters of the Authorization Request are placed in the RO. • Modify CDDL in clause 5.3.2 for adding to EUWrpRegistrarInfo, purpose (mandatory), policyUri (mandatory), and credential (optional).
4/2/2026	V1.1.5	Editorial changes: • Removed DM from Annex A Title (from "EAAP implementation based on JSON-LD W3C VC DM" to "EAAP implementation based on JSON-LD W3C VC", for coherence with notation used in TS 119 472-1 clause 7. • Removed DM abbreviation from clause 3.3 Abbreviations • Changed "W3C VC DM W3C Verifiable Credentials Data Model" to "W3C VC W3C Verifiable Credentials" in clause 3.3 Abbreviations

History

Version	Date	Status
V1.1.1	December 2025	Publication
V1.2.1	March 2026	Publication