ETSI TS 103 705 V1.3.1 (2025-05)

Lawful Interception (LI);
Data Structures for Lawful Disclosure

Reference
RTS/LI-00281

Keywords
lawful disclosure, lawful interception

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

Important notice

The present document can be downloaded from the
ETSI Search & Browse Standards application.

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format on ETSI deliver repository.

Users should be aware that the present document may be revised or have its status changed, this information is available in the Milestones listing.

If you find errors in the present document, please send your comments to the relevant service listed under Committee Support Staff.

If you find a security vulnerability in the present document, please report it through our Coordinated Vulnerability Disclosure (CVD) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2025.
All rights reserved.

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.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Lawful Interception (LI).

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 specifies flexible and extensible data structures for Lawful Disclosure for use in combination with existing handover interface standards, e.g. ETSI TS 103 120 [i.1] or ETSI TS 103 462 [i.2]. Data structures are not limited to telecommunication-specific data.

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] ETSI TS 103 280: "Lawful Interception (LI); Dictionary for common parameters".

[2] IETF RFC 8259: "The JavaScript Object Notation (JSON) Data Interchange Format".

[3] IETF Draft draft-bhutton-json-schema-01: "JSON Schema: A Media Type for Describing JSON Documents".

[4] IETF RFC 4122: "A Universally Unique (UUID) Identifier URN Namespace".

[5] IETF RFC 7515: "JSON Web Signature (JWS)".

[6] IETF RFC 7518: "JSON Web Algorithms (JWA)".

[7] IETF RFC 7797: "JSON Web Signature (JWS) Unencoded Payload Option".

[8] IETF RFC 8037: "CFRG Elliptic Curve Diffie-Hellman (ECDH) and Signatures in JSON Object Signing and Encryption (JOSE)".

[9] ISO 13616-1:2020: "Financial services — International bank account number (IBAN) — Part 1: Structure of the IBAN".

[10] ISO/IEC 7812: "Identification cards — Identification of issuers".

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 103 120: "Lawful Interception (LI); Interface for warrant information".

[i.2] ETSI TS 103 462: "Lawful Interception (LI); Inter LEMF Handover Interface".

3 Definition of terms, symbols and abbreviations

3.1 Terms

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

Communication Service Provider (CSP): service provider who may be required to disclose data

Law Enforcement Agency (LEA): organization entitled to request disclosure of data

3.2 Symbols

Void.

3.3 Abbreviations

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

CSP	Communication Service Provider
JSON	JavaScript Object Notation
JWA	JSON Web Algorithms
JWS	JSON Web Signature
LEA	Law Enforcement Agency
URL	Uniform Resource Locator
UTF	Unicode Transformation Format
UUID	Universally Unique Identifier

4 Overview

4.1 Lawful Disclosure

Lawful Disclosure is the process by which a law enforcement agency requests and receives data from an obligated service provider.

A formal definition of Lawful Disclosure (or the related terms "Retained Data" or "Stored Data") is not given in the present document but can be found in any relevant applicable regulations. The present document does not give a definition of what should be stored or disclosed in any particular jurisdiction. The present document describes how information shall be delivered but does not describe when or whether it should be delivered.

4.2 Nature of the data

The present document considers the handover of "snapshot" information i.e. information which is created relating to a given moment in time. Snapshot information is information that can be collected, delivered and is then complete. The present document does not apply to ongoing streams of information.

The type and nature of the data is not restricted by the present document. It may include customer information, network usage records, etc.

4.3 Structure of data

The present document provides a mechanism for formatting and encoding the results of a Lawful Disclosure request in JSON (see clause 5). The present document provides a means for the CSP to describe the structure of the data through the composition or extension of a catalogue of simple standardized data structures ("entities") into a set of result records ("records").

Such a mechanism provides the CSP with the flexibility to describe their data in the most appropriate way, while setting out requirements on the description of that data to ensure that the LEA can always understand it. Further, this mechanism sets out the principles by which both standardized and CSP-specific extensions (see clause 4.4) are defined to maximize the re-use of standardized components.

The present document does not consider the format or structure of the Lawful Disclosure request, or how it is presented to the CSP.

4.4 CSP-defined information

Where practical, it is helpful for the data to be structured using fully standardized structures. However, in some places the format of data can change, or it can be different for different providers. In general, where a service is based on heavily standardized practices (e.g. traditional telephony) then the data formats should also be stable. Where a service is rapidly changing and differs between different providers (which may be the case for services offered over-the-top of basic connectivity) then the data formats are unlikely to be fully stable and CSP-defined fields are likely to be necessary. However, standardized structures should still be used where possible (see clause 6.5).

4.5 Handover and reference model

The present document does not specify any handover interface. Instead, it only specifies data structures for use in other handover interfaces defined in other specifications, including but not limited to:

- ETSI TS 103 120 [i.1].
- ETSI TS 103 462 [i.2].

Normative instructions on how data structures from the present document can be used can be found in the respective specifications; an informative summary is given in Annex A.

The present document is intended to be used in situations where a Law Enforcement Agency (LEA) has an interface for sending/receiving information with a Communication Service Provider (CSP).

4.6 Assurance of material (e.g. for use in court)

The present document supports external techniques for assurance of material to help its use as part of legal proceedings.

Implementations may choose to digitally sign the JSON-encoded messages for security and assurance purposes. If used, the procedure described in Annex C shall be followed.

5 Response Model

5.1 Overview

The present document describes how a CSP may encode the results of a Lawful Disclosure request as a Response document.

A Response document consists of the elements shown in Table 5.1-1.

Table 5.1-1: Response elements

Field	Cardinality	Description	Reference
recordSetDescription	1	Describes the data being returned by the CSP.	Clause 5.2
recordSet	1	The data being returned by the CSP.	Clause 5.3

A Response document is encoded in JSON as per IETF RFC 8259 [2]. A JSON Schema [3] for validating the high-level structure of a Response is provided in clause 6.1.

5.2 recordSetDescription

5.2.1 Overview

The recordSetDescription provides information related to the Response that is necessary for the LEA to interpret the data being returned by the CSP. It consists of the elements shown in Table 5.2.1-1.

Table 5.2.1-1: recordSetDescription elements

Field	Cardinality	Description	Reference
etsiSchemaId	1	ID of the ETSI JSON schema used by the Response.	Clause 5.2.2
etsiSpecificationVersion	1	Version of the present document used by the Response.	Clause 5.2.3
cspName	1	String which identifies the CSP generating the Response.	Clause 5.2.4
cspSchemaId	1	ID of the CSP Schema used by the Response.	Clause 5.2.5
cspSchemaVersion	1	Version of the CSP Schema.	Clause 5.2.6
resultSetId	1	Unique identifier for this result set.	Clause 5.2.7
requestReference	1	Reference to the original request.	Clause 5.2.8
created	1	Timestamp showing the creation time of the Response.	Clause 5.2.9

5.2.2 etsiSchemaId

Contains the JSON schema ID of the ETSI schema against which the Response was generated i.e. the value of the "$id" field in the ETSI Response Schema (see clause 6.2).

This value shall take the following form, with the placeholder value `{version}` replaced with the version of the schema as determined by the rules in clause 6.3:

```
urn:etsi:li:103705:response_schema_id:v{version}
```

NOTE: Implementers should be aware that the etsiSchemaId value is distinct from the version number of the present document, which may vary independently and is given separately in the etsiSpecificationVersion field as described in clause 5.2.3.

5.2.3 etsiSpecificationVersion

Contains the version number of the present document against which the Response was generated.

NOTE: Implementers should be aware that the etsiSpecificationVersion value may change independently of the schemaId value e.g. if there are changes to the prose of the present document but no changes to the ETSI schema.

5.2.4 cspName

Contains a value that identifies the CSP that generated the Response. The choice of value is out of scope of the present document, but it is recommended to choose a value which is guaranteed to be sufficiently unique for the LEA to unambiguously identify the CSP (e.g. a domain name).

5.2.5 cspSchemaId

Contains a value which matches the "cspSchemaID" field in the CSP Record Schema which defines the valid set of records that may appear in the recordSet (see clause 6.2.4).

5.2.6 cspSchemaVersion

Contains the version number assigned by the CSP to the CSP Record Schema identified by the cspSchemaId (see clauses 5.2.5 and 6.2.4). Shall follow the rules and encoding set out in clause 6.3.

5.2.7 resultSetId

Contains a value which uniquely identifies the Response between a given CSP and LEA. Unless otherwise agreed between CSP and LEA, this shall be given as a UUIDv4 [4] chosen by the CSP at the point in time when the Response is created.

5.2.8 requestReference

Contains a value which uniquely and unambiguously identifies the original disclosure request against which the Response was created. If the Response is being carried according to ETSI TS 103 120 [i.1], then it is recommended that the format and value are as indicated in Annex B. In all other cases, the format and contents shall be agreed between the CSP and LEA in advance.

5.2.9 created

Contains a timestamp indicating when the Response was generated. Shall be given in QualifiedMicrosecondDateTime format according to ETSI TS 103 280 [1] clause 6.5.

5.3 recordSet

5.3.1 Overview

The recordSet element consists of a list of record structures. Each record has the minimum information in it as in Table 5.3.1-1.

Table 5.3.1-1: Record elements

Field	Cardinality	Description	Reference
id	1	Unique identifier of the record within the recordSet.	Clause 5.3.2
type	1	Identifies the type of the record.	Clause 5.3.3
NOTE: The record will contain further fields as defined by the relevant record type (see clause 5.3.3).			

5.3.2 id

Provides a unique identifier for the record within the recordSet. The format and value may be chosen by the CSP, subject to the requirement that each id value in a given recordSet shall occur once and only once.

The id value may be used by the Pointer entity as a way of relating two records in a recordSet together (see clause A.1).

5.3.3 type

The type field contains a value which indicates the type of the record. Shall be given as a JSON reference to the schema definition that defines the type. This in turn shall match one of the allowed values set out in the CSP Record Schema (see clause 6.2.4).

5.4 Pointers/relationships

To facilitate the definition of flat reusable structures, the present document provides a Pointer entity (see clause A.1) to allow a record in a recordSet to refer to data in another record in the same recordSet for extra information or to indicate a relationship.

6 Schema requirements

6.1 Overview

Clause 6 sets out the means by which the present document can be used to meet the requirements in clauses 4.3 and 4.4; namely that CSPs have the ability to describe their own data while also providing sufficient description of that data to allow the LEA to make sense of it.

To achieve this, each CSP is required to define one or more JSON Schema [3] documents (the "CSP Schema"). The CSP schema consists of a CSP Record Schema that describes the records they may include as part of a recordSet in a Response (see clause 5.3) by supplying the definition of the recordSet type required by the ETSI Response Schema. The CSP Schema may also contain one or more CSP Supplementary Schemas which describe CSP-specific definitions for records or types.

This structure is shown in Figure 6.1-1.

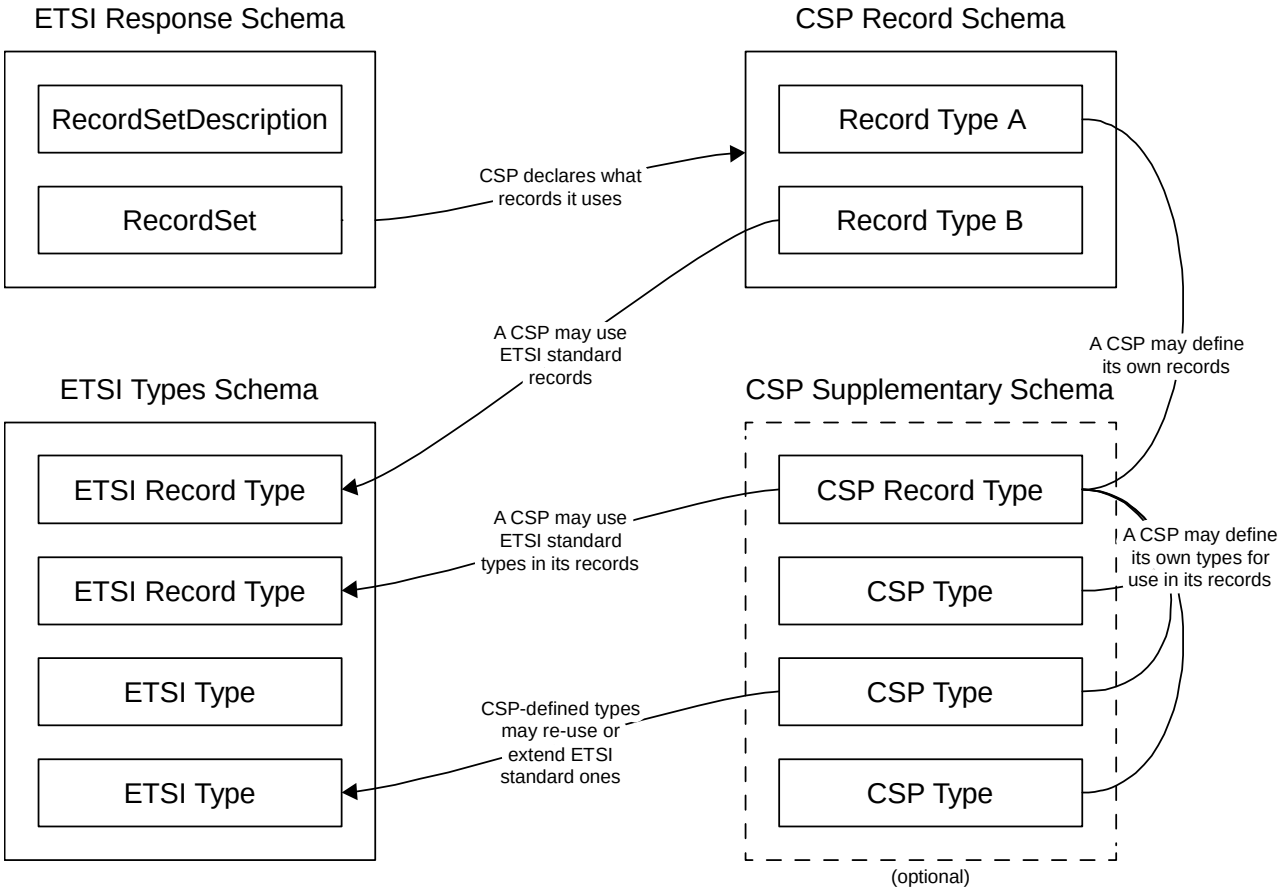

Figure 6.1-1: JSON schema structure

The requirements for the contents, structure and dissemination of the schema (including any other schemas referred to within it) are provided in clause 6. These requirements also apply recursively to any other schemas referenced by a schema compliant with the present document.

6.2 Schema structure

6.2.1 General

The following JSON schema files are supplied via the ETSI TC LI Forge repository at:

- https://forge.etsi.org/rep/li/schemas-definitions/-/tree/spec/103705/1.3.1/103705?ref_type=tags.

Table 6.2.1-1: Schema files provided with the present document

File	Provides	Description
response.schema.json	ETSI Response Schema	Verifies that the Response document meets the requirements in clause 5.
records.schema.json	CSP Record Schema (template only)	The file provided with the present document is a template; the CSP shall provide a file matching this template.
etsi_types.schema.json	ETSI Type Schema	Provides the definitions given in Annex A, which may be used or extended in forming a schema (see clause 6.5 and Annex D for examples).

6.2.2 ETSI Response Schema

The Response document shall conform to the ETSI Response Schema (which includes conformance with CSP Record Schema, see clause 6.2.4), which verifies that the structure is as described in clause 5.

Each schema document in the CSP schema shall follow the requirements set out in clause 6.

6.2.3 ETSI Types Schema

The ETSI Types Schema provides a set of standard definitions that may be used as either records (i.e. referred to directly by the CSP Record Schema) or as components of a CSP-defined Record or Type schema.

CSPs are encouraged to use types defined in the ETSI Types schema whenever possible (see clause 6.5).

6.2.4 CSP Record Schema

The CSP Schema shall contain a CSP Record Schema which defines all records which may appear as part of a recordSet from that CSP. The Record Schema shall contain the following mandatory fields.

Table 6.2.4-1: CSP Record Schema

Field	Description	Value
$id	JSON schema identifier, as set by the relevant ETSI Response schema.	As per the provided template.
$schema	JSON meta-schema identifier.	As per the provided template.
title	Schema title.	As per the provided template.
description	Schema description.	As per the provided template.
cspSchemaID	Identifies the CSP schema. Implementers should note that this is not the value used in the $id field (see above). To be used in the cspSchemaID field of instance documents following the CSP schema (see clause 5.2.5).	Assigned by the CSP.
cspName	String which identifies the CSP responsible for the CSP Schema definition, and which will appear in the cspName field of any Response messages generated against this schema (see clause 5.2.4).	Assigned by the CSP.
dateIssued	Gives the date that the CSP Schema was published or otherwise issued. Shall be given in QualifiedDateTime format (see ETSI TS 103 280 [1], clause 6.4).	Assigned by the CSP.
cspSchemaVersion	Gives a version number of the CSP Schema, to be used in the cspSchemaVersion of instance documents following the CSP Schema (see clause 5.2.6).	Assigned by the CSP. Shall follow the format and semantics given in clause 6.3.
$defs	Contains schema definitions.	Chosen by the CSP, subject to the constraints given in the rest of clause 6.2.4.

The first definition given in the "$defs" field shall be "record", which in turn shall consist of a single "oneOf" element. This element shall contain a list of definitions which completely describe the records that a CSP may return in a recordSet. Each definition shall be given as an "allOf" element containing the following definitions.

Table 6.2.4-2: allOf elements for CSP recordSet definition

Field	Description	Format
$ref	Reference to the type of the record (see rest of clause 6.2.4).	JSON pointer to a schema definition
properties/type/const	Set to the same value as the $ref field.	JSON pointer to a schema definition

The schema pointed to in the $ref field may either be a reference to a type defined in the ETSI Types Schema, or a type defined by the CSP. In the latter case, the definition may be given directly in the CSP Record Schema as a later definition in the "$defs" field or alternatively given in a supplementary CSP Types Schema.

6.2.5 CSP Supplementary Schema

The CSP Schema may contain one or more CSP Supplementary Schemas that provide additional definitions referred to from the CSP Record Schema.

All definitions in a CSP Supplementary Schema shall follow the requirements given clauses 6.4 and 6.5.

When choosing the value for the JSON schema ID ("$id") of a CSP Supplementary Schema, implementers shall not assume that the LEA is able to retrieve schema documents dynamically via the schema ID (e.g. by interpreting a schema ID as a URL and downloading it) without prior agreement between CSP and LEA. Similarly, LEA implementers shall not assume that a schema identifier is network addressable, nor attempt to automatically retrieve schemas based on their schema identifier value, without prior agreement between CSP and LEA.

6.3 Versioning

The ETSI Response Schema, ETSI Types schema and the CSP Schema(s) shall each have a version assigned to it of the form <major>.<minor>.<patch>. The version number of the ETSI Response Schema and ETSI Types schema are assigned by ETSI, while the CSP Schema version is assigned by the CSP responsible for the schema.

The major version shall be incremented when making a backwards-incompatible change.

The minor version shall be incremented when adding a backwards-compatible element.

The patch version shall be incremented when fixing a backwards-compatible bug.

Under this convention, a JSON fragment that is valid against version 1.0.0 of a given schema is not guaranteed to validate against version 2.0.0 of the schema, but is guaranteed to validate against version 1.1.0.

When encoding a version (see clauses 5.2.2, 5.2.3 and 5.2.6), the value shall be given as decimal digits without leading zeroes, separated by period characters (".").

6.4 Schema requirements

6.4.1 Format

Every CSP-defined schema in the CSP Schema shall be a valid JSON Schema [3] document.

6.4.2 Definitions

Each definition given in the CSP Schema shall have the following annotation elements populated.

Table 6.4.2-1: Record Set Description elements

Field	Cardinality	Description
description	1	Provides a description of the new element, including any appropriate indications of usage, format and any other pertinent details of the schema description of this element, sufficient for the LEA to understand the element.
examples	0..1	May be used to provide example values for simple types. If provided, shall be given as an array of one or more valid values that match the definition of the simple type.

6.5 Drafting principles

Schemas shall be drafted according to the principles in this clause 6.5.

The following principles shall be followed except where the principle makes it impractical to convey the relevant information accurately:

- ETSI-defined type shall be used in preference to custom-defined ones (see Annex D example 1 for an example of this).
- Extensions of ETSI-defined types shall be used in preference to new entities (see Annex D example 2 for an example of this).
- If internal structure is required, the new structure shall be factored out into another entity (preferably a standardized entity, or an extension of one, according to the above).
- Anonymous types shall be avoided.

For the purposes of interpreting the principles above, "ETSI-defined types" includes types defined by the ETSI Types Schema (see clause 6.2.3) and types defined in ETSI TS 103 280 [1].

Implementers are encouraged to use the examples given in Annex D as guidance in interpreting these principles.

6.6 Availability

The CSP shall ensure the CSP Schema has been received and accepted by the LEA prior to the first use of that schema in a Response. The means by which it is provided is out of scope of the present document.

Annex A (normative): Catalogue of ETSI-defined types

A.1 Overview

The ETSI Types Schema contains a list of standard types that can be used by CSP schemas for Response types, either by composition or extension (see clause 6.5). The schema is provided as a JSON schema file supplied via the ETSI TC LI Forge repository (see clause 6.2). A list of types provided in the schema is given in Table A.1-1.

Table A.1-1: ETSI standard types

Type	Description
Pointer	Allows one record in a recordSet to point to data in another record in the same recordSet. See clause 5.4.
CallRecord	Details about a record or event related to a call.
MessagingRecord	Details about a record or event related to a message.
EmailRecord	Details about a record or event related to an email.
DataAccessRecord	Information about a data access session (e.g. a mobile data session).
CallPartyInformation	Information about the participants and technologies related to the call.
MessagingPartyInformation	Information about the participants and technologies related to the messages.
EmailPartyInformation	Information about the participants and technologies related to the email.
DataAccessPartyInformation	Information about the participants and technologies related to the data service.
CallPartyId	Identifier of the party related to the call.
MessagingPartyId	Identifier of the party related to the message.
EmailPartyId	Identifier of the party related to the email.
DataAccessPartyId	Identifier of the party related to the data service.
DataVolume	Quantification of the bytes exchanged.
AccessTechnology	Access technology used.
Location	Location information.
RoamingInformation	Inbound / outbound roaming information.
ValidityPeriod	Time period specifying that the combination of values at the same level in the JSON structure has been valid in this time period.
Person	Identifies a natural person.
Organization	Identifies a juridical person.
Address	Identifies a postal address.
BankAccount	Information about a bank account, e.g. IBAN according to ISO 13616-1:2020 [9] and account holder.
PaymentCard	Information about a payment card, e.g. payment card number according to ISO/IEC 7812 [10], type, card holder and date of expiry of the payment card.
Contract	Information about a contract, e.g. number, begin and end date, nature of the contract.
ContactDetails	Information about contact details, e.g. phone number and email address for contacting the subscriber.
IdentityVerfication	Information about an identity verification, e.g. document number, type, date of expiry, issuing authority, country code.
Document	Document information, including e.g. the nature of the document, the content, the mime type and the checksum.

Annex B (informative): Working with other standards

B.1 Working with ETSI TS 103 120

B.1.1 How to include ETSI TS 103 705 (the present document) structures in ETSI TS 103 120

Response documents may be delivered using the Delivery Object as described in ETSI TS 103 120 [i.1], clause 10.

The requestReference field in the Response document (see clause 5.2.8) should be set to the HI1 Object Identifier of the LDTask Object that contains the request. This will also be present as an Associated Object of the Delivery Object (see ETSI TS 103 120 [i.1], clause 10.2.1).

Annex C (normative): Use of Digital Signatures

C.1 Using digital signatures

This annex describes how digital signatures may be used with Response Documents.

The present document defines one approach, given in clause C.2. Other approaches may be added in future.

C.2 Enveloped Signature

C.2.1 Overview

The signature mechanism is based on the JSON Web Signature (JWS) with Detached Content, as described in IETF RFC 7515 [5], Appendix F. It is modified to allow the JWS header and signature to be included as part of the Response Documents.

For signing documents two basic procedures need to be observed, which are the generation of a signature (as described in clause C.2.2) and the verification of a signature (as described in clause C.2.3).

C.2.2 Signing procedure

The signing procedure is as follows:

1) Start with JSON document containing the unsigned Response Document.

2) Add a JSON object named "signature". The "Signature" object has a structure based on the JWS JSON Serialization object in IETF RFC 7515 [5], section 3.2, and the following object members shall be present:

   a) "protected": An empty string when generating the signature, the base64url protected value once signed.

   b) "signature": An empty string when generating the signature, the JWS Signature once calculated.

3) Compute the JWS Protected Header from the JWS Header Parameters to use for the signing, per IETF RFC 7515 [5], section 4. The JWS Protected Header shall contain the "alg" JWS Header Parameter. Other JWS Header Parameters may be provided as required, such as those to support the JWS Unencoded Payload option as per IETF RFC 7797 [7], section 3.

4) Take the JWS Payload to be the octets of the UTF-8 encoding of the modified JSON document containing the Response Document.

5) Compute the JWS Signing Input from the JWS Protected Header and JWS Payload, using the algorithm in IETF RFC 7515 [5], section 5.1, and optionally IETF RFC 7797 [7], section 5.

6) Compute the JWS Signature on the JWS Signing Input per IETF RFC 7515 [5], section 5.1. The JWS Signature shall use an appropriate JSON Web Algorithm (JWA) key per IETF RFC 7518 [6] or IETF RFC 8037 [8].

7) Set the values of the following members of the "Signature" JSON object that was added in step 2:

   a) "protected": The base64url encoded JWS Protected Header from step 3.

   b) "signature": The base64url encoded JWS Signature from step 6.

C.2.3 Verification procedure

The verification procedure verifies the JWS Protected Header and it verifies the JWS Signature. The verification procedure is only considered successful if both verifications succeed.

The verification procedure, which included both of these steps, is as follows:

1) Start with JSON document containing the signed Response Document.
2) Modify the values of the following members of the "Signature" JSON object:
   a) "protected": Temporarily store the value as the encoded JWS Protected Header, and set the "protected" member to the empty string.
   b) "signature": Temporarily store the value as the JWS Signature, and set the "signature" member to the empty string.
3) Decode the JWS Protected Header JSON object from the encoded JWS Protected Header from step 2)a.
4) Verify the JWS Protected Header is supported by the implementation, per IETF RFC 7515 [5], section 5.2 step 5.
5) Take the JWS Payload to be the octets of the UTF-8 encoding of the modified JSON document containing the Response Document.
6) Compute the JWS Signing Input from the JWS Protected Header and JWS Payload, using the algorithm in IETF RFC 7515 [5], section 5.2 or optionally IETF RFC 7797 [7], section 5.
7) Compute the JWS Signature on the JWS Signing Input per IETF RFC 7515 [5], section 5.2. The JWS Signature uses an appropriate JSON Web Algorithm (JWA) key as per IETF RFC 7518 [6] or IETF RFC 8037 [8].
8) Verify the result of step 7 matches the JWS Signature stored in step 2)b.

Implementations shall ensure that the removal of the values in step 2 does not alter any other part of the JSON document. In particular, implementers should take care that the resulting JSON document is not reformatted, and that no changes are made to the indenting or whitespace. Such changes will result in a payload at step 5 different from the one that the signer signed in clause C.2.2 step 4, and will therefore result in a verification failure.

Annex D (informative): Examples

D.1 Example schemas and documents

A set of example documents are maintained on the ETSI TC LI Forge repository (see clause 6.2). The examples are split into directories. Each directory contains a set of files, including both schemas and instance documents, that are intended to illustrate a particular aspect of the present document.

Table D.1-1: Examples

Example directory	Description
example1	Demonstrates how a CSP can use only ETSI standard records in their responses. The CSP imports the ETSI-defined CallRecord and MessagingRecord in their CSP record schema.
example2	Demonstrates how a CSP can extend an ETSI standard record by taking the ETSI-defined MessagingRecord and adding a CSP-specific UUID to it.
example3	Demonstrates how a CSP can use both ETSI standard records and also define a completely novel record type to describe some novel service that is not well-approximated by any standard records.
example4	Demonstrates how the Pointer record can be used. In this example CSP wants to create a simple subscriber record structure where the subscriber can have a number of different addresses (e.g. contact, billing, delivery, installed) but they are often all set to the same value. To reduce the amount of repetition, the CSP uses the Pointer record.
example5	Demonstrates the use of the Signature object following the procedures in clause C.2, using the secret key "secret_key" and the algorithm "HS256".

Annex E (informative): Change history

Status of Technical Specification ETSI TS 103 705 Data Structures for Lawful Disclosure		
TC LI approval date	**Version**	**Remarks**
August 2024	1.1.1	First publication of the TS after approval by Remote Consensus following the final agreements after ETSI TC LI#66 (18-21 June 2024, Lucerne)
November 2024	1.2.1	Included Change Requests: CR001r5 (cat F) Signing JSON documents CR002 (cat F) Correction of table headings These CRs were approved by TC LI#67 (22-24 October 2024, Vancouver)
February 2025	1.3.1	Included Change Requests: CR004r2 (cat B) Adding Roaming Information object CR005r2 (cat B) Adding types for subscriber data These CRs were approved by TC LI#68 (25-27 February 2025, Dublin)

History

Document history		
V1.1.1	August 2024	Publication
V1.2.1	December 2024	Publication
V1.3.1	May 2025	Publication