ETSI TS 103 097 V1.4.1 (2020-10)

Intelligent Transport Systems (ITS);
Security;
Security header and certificate formats

Reference

RTS/ITS-00557

Keywords

ITS, privacy, protocol, security

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 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

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 at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, 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.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Intelligent Transport Systems (ITS).

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.

Introduction

Security policies require that data structures such as messages used in Intelligent Transport Systems are secured when stored or transferred. For interoperability reasons, a common format for secure data structures featuring security headers and public key certificates needs to be provided.

The present document provides these definitions as a profile of the base standard IEEE Std 1609.2 ™-2016 [1] and its amendments IEEE 1609.2a™-2017 and IEEE Std 1609.2b™-2019. A profile makes use of the definitions in the base standard and defines the use of particular subsets or options available in the base standard. This implies that the present document is to be read and interpreted together with that base standard.

From time to time, new versions of the present document may be published that extend IEEE Std 1609.2™ [1] data types using ASN.1 extension mechanisms to define ETSI originated extensions that are not necessarily endorsed by IEEE.

The present document contains material from IEEE Std 1609.2™-2016 [1] and its amendment(s), reprinted with permission from IEEE, and Copyright © 2016.

1 Scope

The present document specifies the secure data structure including header and certificate formats for Intelligent Transport Systems.

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 at https://docbox.etsi.org/Reference.

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] IEEE Std 1609.2™-2016 including amendments IEEE Std 1609.2a™-2017 and IEEE Std 1609.2b™-2019: "IEEE Standard for Wireless Access in Vehicular Environments -- Security Services for Applications and Management Messages".

[2] ETSI TS 102 965: "Intelligent Transport Systems (ITS); Application Object Identifier (ITS-AID); Registration".

[3] Recommendation ITU-T X.696 (08/2015): "Information technology - ASN.1 encoding rules: Specification of Octet Encoding Rules (OER)".

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 are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI TS 102 940: "Intelligent Transport Systems (ITS); Security; ITS communications security architecture and security management".

[i.2] ETSI TS 102 941: "Intelligent Transport Systems (ITS); Security; Trust and Privacy Management".

[i.3] ETSI TS 103 601: "Intelligent Transport Systems (ITS); Security; Security management messages communication requirements and distribution protocols".

[i.4] CPOC protocol.

NOTE: Available at https://cpoc.jrc.ec.europa.eu/Documentation.html.

3 Definition of terms, symbols and abbreviations

3.1 Terms

Void.

3.2 Symbols

Void.

3.3 Abbreviations

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

AA	Authorization Authority
ASN.1	Abstract Syntax Notation One
AT	Authorization Ticket
CA	Certification Authority
CAM	Cooperative Awareness Message
COER	Canonical Octet Encoding Rules
CRL	Certificate Revocation List
CTL	Certificate Trust List
DENM	Decentralized Environmental Notification Message
EA	Enrolment Authority
ECDSA	Elliptic Curve Digital Signature Algorithm
ECIES	Elliptic Curve Integrated Encryption Scheme
ITS	Intelligent Transport Systems
ITS-AID	ITS Application ID
ITS-S	Intelligent Transport Systems Station
OID	Object IDentifier
TLM	Trust List Manager

4 Basic format elements

4.1 Based on the published version of IEEE Std 1609.2

Data structures in the present document are defined using Abstract Syntax Notation 1 (ASN.1) and shall be encoded using the Canonical Octet Encoding Rules (COER) as defined in Recommendation ITU-T X.696 [3]. This includes some data structures in the present document for which a "canonical encoding" is used as defined in IEEE Std 1609.2 [1].

Clauses 5 and 6 specify and describe the data structures with reference to IEEE Std 1609.2 [1]. The corresponding ASN.1 module is defined in annex A.

The validity of a certificate shall be assessed as defined in IEEE Std 1609.2 [1] clause 5.1, using the Hash ID-based revocation method for EA and AA certificates, and no revocation method for authorization tickets and enrolment credentials.

NOTE 1: The CRL for EA and AA certificates is defined in ETSI TS 102 941 [i.2].

NOTE 2: The rules for verification of the Root CA certificate against the CTL are defined in ETSI TS 102 941 [i.2].

The validity of signed data shall be assessed as defined in IEEE Std 1609.2 [1], clause 5.2.

4.2 Extensions

4.2.1 General process

NOTE 1: This clause and the following clause outline approaches for maintaining and extending the present document and do not directly specify functionality to be implemented.

IEEE Std 1609.2 [1] structures are extensible using ASN.1 extension mechanisms.

For all extensible IEEE Std 1609.2 [1] data types other than `HeaderInfo`, extensions will be done by adding new fields after the extension marker in the underlying IEEE Std 1609.2 [1] data type. To avoid conflicts that might arise if multiple stakeholder groups want to extend the same IEEE Std 1609.2 [1] data type at the same time, the rapporteur of the present document will need to coordinate with the editor of IEEE Std 1609.2 [1] and ensure that different extension identifiers are associated with each different extension that is being simultaneously developed.

NOTE 2: In the above paragraph, "extension identifier" refers to the numeric identifier that the ASN.1 encoder automatically associates with an extension field in an ASN.1 structure. The numeric identifier is assigned automatically by the encoder based on the index of the extension field in the list of extension fields in that structure; it is not an identifier that is assigned through a registration process and visible to a human reader of the ASN.1.

4.2.2 HeaderInfo extensions

Background. The `HeaderInfo` type defined in IEEE Std 1609.2 [1] has a structure including an ASN.1 "…" extension marker. The fields after the extension marker are the extension fields. The reader is referred to IEEE Std 1609.2 [1] directly for the current IEEE definition.

In the present document, the type `ToBeSignedData` shall have the component `headerInfo` of type `HeaderInfo` as defined in IEEE Std 1609.2 [1], clause 6.3.9, with the addition of:

- the component `contributedExtensions` as specified in clause A.2.2:
  - within the component `contributedExtensions`, an optional sequence of components of type `EtsiOriginatingHeaderInfoExtension` as specified in clause A.2.2.

`HeaderInfo` extensions are included in the component `contributedExtensions`.

The component `contributedExtensions` is of type `ContributedExtensionBlocks` and is a sequence of single extension "blocks" of type `ContributedExtensionBlock`. Each extension block is defined by an identified contributing organization. The ETSI TC ITS WG5 extension block shall be identified by the integer `etsiHeaderInfoContributorId` (2). Within the ETSI TC ITS WG5 extension block, each extension shall be of type `EtsiOriginatingHeaderInfoExtension`. ASN.1 implementing these design principles is specified in clause A.2.2.

The type `EtsiOriginatingHeaderInfoExtension` is defined in the module `EtsiTs103097ExtensionModule` specified in clause A.1 and composed of the component `id` and the component `Extn`. The component `id` shall be of type `ExtId` and shall uniquely identify the extension within the set of `EtsiOriginatingHeaderInfoExtension`s. The component `content` shall be associated to the related `id` according to the information object set `EtsiTs103097HeaderInfoExtensions`. The ETSI originated extensions shall be defined as information objects of the class `Extension` and shall be listed in the information object set `EtsiTs103097HeaderInfoExtensions`.

NOTE: This approach allows ETSI to specify new extensions as necessary, using an identifier that is entirely under ETSI's control (the `EtsiTs103097HeaderInfoExtensionId`) to identify those extensions and a separate module called `EtsiTs103097ExtensionModule` that can be updated by ETSI without a need to change the module `IEEE1609dot2.`

The data type `ExtensionModuleVersion` in the module `EtsiTs103097ExtensionModule` shall indicate the version of the module `EtsiTs103097ExtensionModule` and shall be imported into the `EtsiTs103097Module.`

ASN.1 implementing these design principles is specified in clause A.1.

5 Specification of secure data structure

5.1 EtsiTs103097Data

A secure data structure shall be of type `EtsiTs103097Data` as defined in annex A, which corresponds to a `Ieee1609Dot2Data` as defined in IEEE Std 1609.2 [1], clause 6.3.2, with the constraints defined in this clause, in clause 5.2 and in clause 5.3.

The type `Ieee1609Dot2Data` shall support the following options in the component content:

- The option `unsecuredData` shall be used to encapsulate an unsecured data structure.
- The option `signedData`, corresponding to the type `SignedData` as defined in IEEE Std 1609.2 [1], clause 6.3.4, shall be used to transfer a data structure with a signature.
- The option `encryptedData`, corresponding to the type `EncryptedData` as defined in IEEE Std 1609.2 [1], clause 6.3.30, shall be used to transfer an encrypted data structure.

The following corresponding profiles of the type `EtsiTs103097Data` are defined in annex A:

- The parameterized type `EtsiTs103097Data-Unsecured` using the Ieee1609Dot2Data option `unsecuredData`.
- The parameterized type `EtsiTs103097Data-Signed` using the Ieee1609Dot2Data option `signedData` containing the data structure in the component `tbsData.payload.data`.
- The parameterized type `EtsiTs103097Data-SignedExternalPayload` using the `Ieee1609Dot2Data` option `signedData` containing the digest of the data structure in the component `tbsData.payload.extDataHash`.
- The parameterized type `EtsiTs103097Data-Encrypted`, using the `Ieee1609Dot2Data` option `encryptedData` containing the encrypted data structure in the component `ciphertext.aes128ccm.ccmCiphertext`.
- The parameterized type `EtsiTs103097Data- SignedAndEncrypted`, using the parameterized type `EtsiTs103097Data-Encrypted`, containing an encrypted `EtsiTs103097Data-Signed.`
- The parameterized type `EtsiTs103097Data-Encrypted-Unicast` using the parameterized type `EtsiTs103097Data-Encrypted` further constraint to have one entry in the component `recipients`.
- The parameterized type `EtsiTs103097Data-SignedAndEncrypted-Unicast` using the parameterized type `EtsiTs103097Data-Encrypted` containing an encrypted `EtsiTs103097Data-Signed` and further constraint to have one entry in the component `recipients`.

5.2 SignedData

The type `SignedData` shall have the following constraints:

- The component `hashId` of `SignedData` shall indicate the hash algorithm to be used to generate the hash of the message according to IEEE Std 1609.2 [1], clauses 6.3.5 and 5.3.3.
- The component `tbsData` of `SignedData` shall be of type `ToBeSignedData` as defined in IEEE Std 1609.2 [1] clause 6.3.6. The type `ToBeSignedData` shall have the component `payload` of type `SignedDataPayload` as defined in IEEE Std 1609.2 [1], clause 6.3.7, containing either:
  - the component `data`, containing the payload to be signed as an `Ieee1609Dot2Data`; or
  - the component `extDataHash`, containing the hash of data that is not explicitly transported within the structure.

The type `ToBeSignedData` shall have the component `headerInfo` of type `HeaderInfo` as defined in IEEE Std 1609.2 [1], clause 6.3.9, and constrained to have the following security headers:

- The component `psid` containing the ITS-AID corresponding to the contained message.
- The component `generationTime` as defined in IEEE Std 1609.2 [1], always present.
- The component `expiryTime`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of message profiles in clause 7.
- The component `generationLocation`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of message profiles in clause 7.
- The component `p2pcdLearningRequest` always absent.
- The component `missingCrlIdentifier` always absent.
- The component `encryptionKey`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of message profiles in clause 7.
- The component `inlineP2pcdRequest`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of message profiles in clause 7.
- The component `requestedCertificate`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of message profiles in clause 7.
- The component `pduFunctionalType`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of message profiles in clause 7.
- In the component `contributedExtensions`, any component of type `EtsiOriginatingHeaderInfoExtension` identified in the Information Object Set `EtsiTcItsHeaderInfoExtensions` present or absent according to the specification of the message profiles in clause 7 and according to the specification of the particular extension in the document that specifies it:
  - The extension EtsiTs102941CrlRequest, if present, indicates that the ITS-Station is requesting a CRL according to ETSI TS 103 601 [i.3], with format as defined in ETSI TS 102 941 [i.2]. The component issuerId shall indicate the issuer of the CRL and the component lastKnownUpdate, if present, shall indicate the value of the thisUpdate field of the latest CRL that the ITS-Station has available.
  - The extension EtsiTs102941DeltaCtlRequest, if present, indicates that the ITS-Station is requesting a delta CTL according to ETSI TS 103 601 [i.3], with format as defined in ETSI TS 102 941 [i.2], using the data structure EtsiTs102941CtlRequest. The component issuerId shall indicate the issuer of the CTL and the component lastKnownCtlSequence, if present, shall indicate the value of the ctlSequence field of the latest CTL that the ITS-Station has available.

- In the component `contributedExtensions`, any component of type other than `EtsiOriginatingHeaderInfoExtension` always absent.

NOTE: The present document does not specify contributedExtensions fields of type other than `EtsiOriginatingHeaderInfoExtension` and does not specify what an implementation that processes received secure data structures should do, based on such extensions. Anyhow, compliance to the present document requires an implementation to correctly parse received secure data structures that contain those extensions.

The component `signer` of `SignedData` shall be of type `SignerIdentifier` as defined in IEEE Std 1609.2 [1], clause 6.3.24, and constrained to one of the following choices:

- `digest`, containing the digest of the signing certificate as defined in IEEE Std 1609.2 [1], clause 6.3.26.
- `certificate`, constrained to only one entry in the `SequenceOfCertificate` list of type `TS103097Certificate`, containing the signing certificate as defined in clause 6 of the present document.

The component `signature` of `SignedData` shall be of type `Signature` as defined in IEEE Std 1609.2 [1], clause 6.3.28 and shall contain the ECDSA signature as defined in IEEE Std 1609.2 [1], clauses 6.3.29, 6.3.29a and 5.3.1.

5.3 EncryptedData

The type `EncryptedData` shall have the following constraints:

- The component `recipients` of `EncryptedData` shall be of type `SequenceOfRecipientInfo` as defined in IEEE Std 1609.2 [1], clause 6.3.31. Every entry shall be either of option `pskRecipInfo` as defined in IEEE Std 1609.2 [1], clause 6.3.32, of option `certRecipInfo`, or of option `signedDataRecipInfo`, as defined in IEEE Std 1609.2 [1], clause 6.3.34.
- The encryption scheme used shall be ECIES as defined in IEEE Std 1609.2 [1], clause 5.3.5. The component `ciphertext` of `EncryptedData` shall be of type `SymmetricCiphertext` as defined in IEEE Std 1609.2 [1] clause 6.3.37 and contain an `EtsiTs103097Data` encrypted according to IEEE Std 1609.2 [1], clauses 6.3.38 and 5.3.8.

6 Specification of certificate format

A certificate contained in a secure data structure shall be of type `EtsiTs103097Certificate` as defined in annex A, which corresponds to a single `ExplicitCertificate` as defined in IEEE Std 1609.2 [1], clause 6.4.6, with the constraints defined in this clause.

The component `toBeSigned` of the type `EtsiTs103097Certificate` shall be of type `ToBeSignedCertificate` as defined in IEEE Std 1609.2 [1], clause 6.4.8 and constrained as follows:

- The component `id` of type `CertificateId` constrained to choice type `name` or `none`.
- The component `cracaId` set to 000000'H.
- The component `crlSeries` set to 0'D.
- The component `validityPeriod` with no further constraints.
- The component `region` of type `GeographicRegion` as defined in IEEE Std 1609.2 [1], present or absent according to the specification of certificate profiles in clause 7.
- The component `assuranceLevel` of type `SubjectAssurance`, as defined in IEEE Std 1609.2 [1], present or absent according to the specification of certificate profiles in clause 7.