INTERNATIONAL STANDARD

ISO/IEC 18014-2

First edition
2002-12-15

Information technology — Security techniques — Time-stamping services —

Part 2: Mechanisms producing independent tokens

Technologies de l'information — Techniques de sécurité — Services d'horodatage —

Partie 2: Mécanismes produisant des jetons indépendants

Reference number
ISO/IEC 18014-2:2002(E)



© ISO/IEC 2002

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

© ISO/IEC 2002

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

--`,,,`-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

© ISO/IEC 2002 – All rights reserved

Not for Resale

Contents

Foreword iv

Introduction v

1 Scope 1

2 Normative References 1

3 Terms and Definitions 2

4 General Discussion 3

5 Entities of the Time-Stamping Process 4

6 Message Formats 4
6.1 Object Identifiers 6
6.2 Extension fields 6
6.2.1 ExtHash extension 6
6.2.2 ExtMethod extension 6
6.2.3 ExtRenewal extension 7

7 Time-stamps using digital signatures 7
7.1 TSA response 8
7.2 Token verification 9

8 Time-stamps using message authentication codes 10
8.1 TSA response 10
8.2 MAC generation 12
8.3 MAC verification 12
8.4 Token verification 12

9 Time-stamps using archiving 13
9.1 TSA response 13
9.2 Token verification 13

Annex A (normative) **ASN.1 Module for time-stamping** 15

Annex B (informative) **Data Structures** 25

Bibliography 28

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

ISO/IEC 18014-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *IT Security techniques*.

ISO/IEC 18014 consists of the following parts, under the general title *Information technology — Security techniques — Time-stamping services*:

— *Part 1: Framework*

— *Part 2: Mechanisms producing independent tokens*

— *Part 3: Mechanisms producing linked tokens*

Further parts may follow.

© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Introduction

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this International Standard may involve the use of patents.

The ISO and IEC take no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured the ISO and IEC that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with the ISO and IEC. Information may be obtained from:

ISO/IEC JTC 1/SC 27 Standing Document 8 (SD 8) "Patent Information"

SD 8 is publicly available at: http://www.din.de/ni/sc27

Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights other than those identified above. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

© ISO/IEC 2002 All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Information technology — Security techniques — Time-stamping services – Part 2: Mechanisms producing independent tokens

1 Scope

A time-stamping service provides evidence that a data item existed before a certain point in time. Time-stamp services produce time-stamp tokens, which are data structures containing a verifiable cryptographic binding between a data item's representation and a time-value. This part of ISO/IEC 18014 defines time-stamping mechanisms that produce independent tokens, which can be verified one by one.

2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 7498-2: 1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture*

ISO/IEC 8824-1: 1998 | ITU-T Recommendation X.680 (1997), *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*

ISO/IEC 8824-2: 1998 | ITU-T Recommendation X.681 (1997), *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*

ISO/IEC 8824-3: 1998 | ITU-T Recommendation X.682 (1997), *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification*

ISO/IEC 8824-4: 1998 | ITU-T Recommendation X.683 (1997), *Information technology – Abstract Syntax Notation One (ASN.1): Parameterisation of ASN.1 specifications*

ISO/IEC 8825-1: 1998 | ITU-T Recommendation X.690 (1997), *Information technology – ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*

ISO/IEC 9594-8: 2001 | ITU-T Recommendation X.509 (2000), *Information technology – Open Systems Interconnection – The Directory: Public key and attribute certificate frameworks*

ISO/IEC TR 14516: 2002 | ITU-T Recommendation X.842 (2000), *Information technology – Guidelines for the use and management of Trusted Third Party services*

ISO/IEC 9798-1: 1997, *Information technology – Security techniques – Entity authentication – Part 1: General*

ISO/IEC 10181-2: 1996, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Authentication framework*

ISO/IEC 11770-1: 1996, *Information technology – Security techniques – Key management – Part 1: Framework*

ISO/IEC 11770-3: 1999, *Information technology – Security techniques – Key management – Part 3: Mechanisms using asymmetric techniques*

ISO/IEC 13888-1: 1997, *Information technology – Security techniques – Non-repudiation – Part 1: General*

ISO/IEC 14888 (all parts), *Information technology – Security techniques – Digital signatures with appendix*

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

ISO/IEC 18014-1: 2002, *Information technology – Security techniques – Time-stamping services – Part 1: Framework*

3 Terms and Definitions

The following terms are used as defined in ISO/IEC 7498-2:

Cryptography: the discipline that embodies principles, means, and methods for the transformation of data in order to hide its information content, prevent its undetected modification and/or prevent its unauthorized use.

Data integrity: the property that data has not been altered or destroyed in an unauthorized manner.

Data origin authentication: the corroboration that the source of data received is as claimed.

Digital signature: data appended to, or a cryptographic transformation (see cryptography) of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery e.g. by the recipient.

The following term is used as defined in ISO/IEC 8825-1:

Distinguished Encoding Rules (DER): encoding rules that may be applied to values of types defined using the ASN.1 notation. Application of these encoding rules produces a transfer syntax for such values. It is implicit that the same rules are also to be used for decoding. The DER is more suitable if the encoded value is small enough to fit into the available memory and there is a need to rapidly skip over some nested values.

The following term is used as defined in ISO/IEC 9594-8:

Certification path: an ordered sequence of certificates of objects in the DIT (directory information tree) which, together with the public key of the initial object in the path, can be processed to obtain that of the final object in the path.

The following terms are used as defined in ISO/IEC 9798-1:

Asymmetric key pair: a pair of related keys where the private key defines the private transformation and the public key defines the public transformation.

Asymmetric signature system: a system based on asymmetric techniques whose private transformation is used for signing and whose public transformation is used for verification.

The following term is used as defined in ISO/IEC 10181-2:

Authentication: the provision of assurance of the claimed identity of an entity.

The following term is used as defined in ISO/IEC 11770-1:

Certification authority (CA): a centre trusted to create and assign public key certificates. Optionally, the certification authority may create and assign keys to the entities.

The following terms are used as defined in ISO/IEC 13888-1:

Message Authentication Code (MAC): a data item derived from a message using symmetric cryptographic techniques and a secret key. It is used to check the integrity and origin of a message by any entity holding the secret key.

Private key: that key of an entity's asymmetric key pair that is usable only by that entity. In the case of an asymmetric signature system, the private key and the associated algorithms define the signature transformation.

© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Public key: the key of an entity's asymmetric key pair that can be made public. In the case of an asymmetric signature system, the public key and the associated algorithms define the verification transformation.

Public key certificate: a security certificate which binds unforgeably the public key of an entity to the entity's distinguishing identifier, and which indicates the validity of the corresponding private key.

The following term is used as defined in ISO/IEC 14888-2:

Trusted third party: a security authority, or its agent, trusted by other entities with respect to security related activities.

The following terms are used as defined in ISO/IEC 18014-1:

Time-stamping authority (TSA): a trusted third party trusted to provide a time-stamping service.

Time-stamping service (TSS): a service providing evidence that a data item existed before a certain point in time.

Time-stamp requester: an entity which possesses data it wants to be time-stamped.

Time-stamp token: a data structure containing a verifiable cryptographic binding between a data items' representation and a time-value. A time-stamp token may also include additional data items in the binding.

Time-stamp verifier: an entity which possesses data and wants to verify that it has a valid time-stamp bound to it. The verification process may be performed by the verifier itself or by a Trusted Third Party.

4 General Discussion

ISO/IEC 18014-1 presents a general framework for the provision of time-stamping services.

Time-stamping services produce time-stamp tokens. These tokens are associations between data and points in time, and are created in a way that aims to provide evidence that the data existed at the associated date and time. In addition the evidence may be used by non-repudiation services.

This part of ISO/IEC 18014 presents mechanisms to produce time-stamp tokens that are independent: in order to verify a time-stamp token, verifiers do not need access to any other time-stamp token.

Independency means that a verifier just requires one time-stamp token to verify the point in time at which the document existed.

Three mechanisms are presented. The first one is based on digital signatures, and is backwards compatible with the time-stamping protocol defined by IETF [RFC3161]. The second mechanism uses a message authentication code (MAC) to authenticate the binding in a time-stamp token, and the third mechanism is based on TSA archiving of information.

The first mechanism asks the time-stamp service provider to digitally sign the binding of the time to the document so that signature verification sustains the evidence.

The second mechanism asks the time-stamp service provider to use a MAC to sign the binding. The same secret is needed for signature creation and for signature verification, and this secret is kept by the TSA. Therefore, the TSA is required for verification.

The third mechanism asks the time-stamp service provider to archive the evidence, and only publish a reference to the archive. Therefore, the TSA is required for archival and verification.

Time-stamping service users may select the mechanism to be used by means of the ExtMethod extension specified in ISO/IEC 18014-1. If no mechanism is explicitly selected in a time-stamp request, the digital signature mechanism is assumed.

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

NOTE: In order to guarantee full interoperability with IETF compliant time-stamping servers, this extension must be omitted in order to force the usage of the digital signature mechanism, which matches the protocol specified in [RFC3161].

5 Entities of the Time-Stamping Process

The following entities (from ISO/IEC 18014-1) may be involved when a time-stamp is requested or verified:

- the time-stamp **requester;**
- the time-stamp **verifier**; and
- the **Time-Stamping Authority (TSA).**

6 Message Formats

This section reproduces some ASN.1 definitions that appear in ISO/IEC 18014-1, and introduces new ones. The complete ASN.1 module can be found Annex A.

The TSA produces the time-stamp information that is defined in ISO/IEC 18014 part 1 as:

```
TSTInfo ::=  SEQUENCE {

      version       Version,

      policy        TSAPolicyId,

      messageImprint    MessageImprint,

      serialNumber      SerialNumber,

      genTime       GeneralizedTime,

      accuracy          Accuracy OPTIONAL,

      ordering          BOOLEAN DEFAULT FALSE,

      nonce             Nonce OPTIONAL,

      tsa           [0] EXPLICIT GeneralName OPTIONAL,

      extensions        [1] Extensions OPTIONAL

}
```

This TSTInfo is wrapped into a time-stamp response, DER-encoded into an OCTET STRING:

```
ETSTInfo ::=

      OCTET STRING (CONTAINING TSTInfo ENCODED BY der)
```

The wrapping depends on the mechanism used.

© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

The time-stamp response returned to the requester is defined as:

```
TimeStampResp ::= SEQUENCE {
      status         PKIStatusInfo,
      timeStampToken    TimeStampToken OPTIONAL
}

PKIStatusInfo ::= SEQUENCE {
      status         PKIStatus,
      statusString      PKIFreeText OPTIONAL,
      failInfo          PKIFailureInfo OPTIONAL
}

TimeStampToken ::= SEQUENCE {
      contentType       CONTENT.&id({Contents}),
      content    [0] EXPLICIT CONTENT.&Type({Contents}{@contentType})
}

Contents CONTENT ::= {
      time-stamp-mechanism-signature  |
      time-stamp-mechanism-MAC        |
      time-stamp-mechanism-archival,
       --
      ...  -- Expect additional time-stamp mechanisms --
}

time-stamp-mechanism-signature CONTENT ::=
      { SignedData IDENTIFIED BY id-signedData }

time-stamp-mechanism-MAC CONTENT ::=
      { AuthenticatedData IDENTIFIED BY id-ct-authData }
```

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
time-stamp-mechanism-archival CONTENT ::=

      { ETSTInfo IDENTIFIED BY id-data }
```

The value of the contentType component is an OBJECT IDENTIFIER. It is tightly bound to the type of the content component. The Contents information object set is referenced by both components of TimeStampToken. Each object in the Contents set specifies a valid pair of associated contentType and content values. There is one object in the set for each of the time-stamp mechanisms defined in this standard.

6.1 Object Identifiers

A number of OBJECT IDENTIFIERS are required to support the mechanisms defined below. These identifiers are used to uniquely identify specific time-stamping mechanisms. The following root identifier is defined to support allocation of further identifiers for independent token mechanisms:

```
tsp-itm  OBJECT IDENTIFIER::= { iso(1) standard(0) time-stamp(18014) itm(2) }
```

Identifiers defined in this document originate from this root identifier, except where declared otherwise.

6.2 Extension fields

The following extensions fields are identified to be carried either by the TimeStampReq or the TSTInfo. Further extensions may be identified in the future.

6.2.1 ExtHash extension

A requester of time-stamping services may wish to submit for time-stamping more than one hash value derived from a single data item. To enable the submission of multiple hash values the following extension is defined:

```
extHash EXTENSION ::= { SYNTAX ExtHash IDENTIFIED BY tsp-ext-hash }


ExtHash ::= SEQUENCE SIZE(1..MAX) OF MessageImprint


tsp-ext-hash OBJECT IDENTIFIER ::= { tsp-ext 1 }
```

This extension is carried in the "extensions" field of both the TimeStampReq message sent by a requester to a TSA and in the "extensions" field of the resulting TSTInfo structure formed by the TSA and returned to the requester. The TSA reproduces the contents of this extension unmodified.

6.2.2 ExtMethod extension

A requester of time-stamping services may wish to indicate to a specific TSA which time-stamping method to use when forming the eventual time-stamp token. To enable a requester to indicate to a specific TSA which time-stamping method to use in forming the resulting time-stamp token, the following extension is defined:

```
extMethod EXTENSION ::= { SYNTAX ExtMethod IDENTIFIED BY tsp-ext-meth }
```


© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
ExtMethod ::= SEQUENCE SIZE(1..MAX) OF Method


Method ::= METHOD.&id({Methods})


tsp-ext-meth OBJECT IDENTIFIER ::= { tsp-ext 2 }
```

This extension is carried in the "extensions" field of both the TimeStampReq message sent by a requester to a TSA and in the "extensions" field of the resulting TSTInfo structure formed by the TSA and returned to the requester.

If this extension is present and the TSA is able to process it, then the TSA must attempt to fulfil the request for the specified method, or return an error indicating the method is not available. If the requester specifies more than one possible method, the TSA may select one of the suggested methods for use in forming the time-stamp token. If this extension is not present, the TSA uses its default time-stamping mechanism.

6.2.3 ExtRenewal extension

A requester of time-stamping services may wish to indicate to the TSA that the current time-stamping request is a renewal of a time-stamp on data that were already time-stamped in the past, so that the validity period of the old time-stamp is effectively extended. For this purpose, the following extension is defined:

```
extRenewal EXTENSION ::= { SYNTAX ExtRenewal IDENTIFIED BY tsp-ext-renewal }


ExtRenewal ::= TimeStampToken


tsp-ext-renewal OBJECT IDENTIFIER ::= { tsp-ext 3 }
```

This extension is carried in the “extensions” field of both the TimeStampReq message sent by a requester to a TSA and in the “extensions” field of the resulting TSTInfo structure formed by the TSA and returned to the requester. The TSA reproduces the contents of this extension unmodified.

7 Time-stamps using digital signatures

In this mechanism the TSA has an asymmetric key pair, and uses the private key to digitally sign the time-stamp token. Signature verification will use the public key.

This mechanism may require the use of a PKI (Public Key Infrastructure) in order to authenticate the public key and associated usage.

The requester of the service may specify this mechanism using the ExtMethod extension that is specified in the extensions field of the Time-stamp request message, with the following object identifier:

```
tsp-itm-ds OBJECT IDENTIFIER::= { tsp-itm ds(1) }
```

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

If the service requester does not specify any time-stamp mechanism, the digital signature mechanism is assumed.

7.1 TSA response

The TimeStampToken is a sequence type that contains two components (i.e., contentType and content), which are defined in terms of the &id and &Type fields of the CONTENT information object class. Both components are constrained by the information object set Contents, which contains one object for each time-stamp mechanism, such as the object

```
{ SignedData IDENTIFIED BY id-signedData }
```

For the digital signature time-stamp mechanism, this constraint on the components of TimeStampToken requires that the contentType component contains the value id-signedData, and that the content component contains a value of type SignedData.

The object identifier value id-signedData is defined as:

```
id-signedData OBJECT IDENTIFIER ::= {

  iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs7(7) 2

}
```

The content is SignedData as described in ISO/IEC 18014-1 annex B:

```
SignedData::= SEQUENCE {

      version    CMSVersion,

      digestAlgorithms  DigestAlgorithmIdentifiers,

      encapContentInfo  EncapsulatedContentInfo,

      certificates   [0] CertificateSet OPTIONAL,

      crls       [1] CertificateRevocationLists OPTIONAL,

      signerInfos      SignerInfos

}


EncapsulatedContentInfo::= SEQUENCE {

      eContentType   CONTENT.&id({EContents}),

      eContent       [0] EXPLICIT CONTENT.&Type({EContents} {@eContentType})

}
```


© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
EContents CONTENT ::= {

  { ETSTInfo IDENTIFIED BY id-ct-TSTInfo },

}


id-ct-TSTInfo OBJECT IDENTIFIER::= {

  iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)

  smime(16) ct(1) tstinfo(4) }
```

eContent is the DER-encoded value of a TSTInfo structure.

The TSA must sign all time-stamp tokens with a key reserved specifically for that purpose. When a PKI is being used that complies to X.509 v3 certificates [ISO/IEC 9594-8: 2000], the corresponding certificate for the TSA must contain only one instance of the extended key usage field extension with KeyPurposeID having value id-kp-timeStamping [RFC2459]. This extension must be critical.

```
id-kp-timeStamping OBJECT IDENTIFIER::= {

  iso(1) identified-organization(3) dod(6) internet(1)

  security(5) mechanisms(5) pkix(7) kp (3) timestamping (8) }
```

7.2 Token verification

Let Ti be the date and time recorded in the time-stamp token (that is, the issue time). Let Tv be the moment when the time-stamp token is verified. The validity of a time-stamp token may be verified by checking that:

- the time-stamp token is syntactically well-formed;
- the value of the messageImprint component of the time-stamp token matches the value of a messageImprint evaluated at Tv over the document subject to scrutiny;
- the value of the messageImprint component of every ExtHash extension in the time-stamp token matches the value of a messageImprint evaluated at Tv over the document subject to scrutiny;
- there is a valid certification path at Ti
  C_0 ... C_i C_{i+1} ... C_R
  such that
  - for i= 1 .. R; C_i certifies C_{i-1};
  - C_0 is a valid certificate for the signing TSA;
  - C_0 verifies the signature of the time-stamp token;
  - C_0 contains one extended key usage extension indicating that its purpose is id-kp-timeStamping;
  - C_R is a valid certificate for a trusted CA;

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

- every certificate applies at Ti; that is, Ti is within certificate validity, and the applicable CRL at Ti does not revoke nor suspend the certificate, and
- the certification policies of each certificate satisfy the intended purpose of the validation being carried on;

- there is a valid certification path at Tv
  C_0 ... C_i C_{i+1} ... C_R
  such that
  - for i= 1 .. R; C_i certifies C_{i-1};
  - C_0 is a valid certificate for the signing TSA;
  - C_0 verifies the signature of the time-stamp token;
  - C_0 contains one extended key usage extension indicating that its purpose is id-kp-timeStamping;
  - C_R is a valid certificate for a trusted CA;
  - every certificate applies at Tv; that is, Ti is within certificate validity time, and the applicable CRL at Tv does not mention “key compromise” as the termination reason, and
  - the certification policies of each certificate satisfy the intended purpose of the validation being carried on.

Further evidence may be required for specific purposes of the verifier, and obtained by other means that are out of the scope of this standard.

8 Time-stamps using message authentication codes

In this mechanism the TSA uses a secret key to digitally bind the time association. The time-stamp token is authenticated using a Message Authentication Code (MAC).

When using this mechanism, the TSA is needed to carry out the verification, and the TSA has to be thoroughly trusted since there is no external evidence that might detect fraud.

Every exchange of information between the different actors (requester, verifier, and TSA) requires data integrity and data origin authentication protection. This protection may be provided by any means, for example over a secured channel, or using public keys that do not need to last longer than the transaction lapse.

In order to use this mechanism, the requester will use the ExtMethod extension that is specified in the extensions field of the time-stamp request message, with the following object identifier:

```
tsp-itm-mac OBJECT IDENTIFIER::= { tsp-itm mac(2) }
```

MAC evaluation is carried on by the TSA as described below. The secret key used to evaluate the MAC is kept secret. The secret key used for each token must be available for later verification.

The secret key used to seal a given token may be specific to that token, or common for a range of tokens.

8.1 TSA response

The TimeStampToken is a sequence type that contains two components, contentType and content, which are defined in terms of the &id and &Type fields of the CONTENT information object class. Both components are

© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

constrained by the information object set Contents, which contains one object for each time-stamp mechanism, such as the object

```
{ AuthenticatedData IDENTIFIED BY id-ct-authData }
```

For the MAC time-stamp mechanism, this constraint on the components of TimeStampToken requires that the contentType component contain the value id-ct-authData, and that the content component contain a value of type AuthenticatedData.

The object identifier value id-ct-authData is defined as:

```
id-ct-authData OBJECT IDENTIFIER ::= {

  iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)

  smime(16) ct(1) 2 }
```

The content is AuthenticatedData, defined as:

```
AuthenticatedData::= SEQUENCE {

      version    CMSVersion,

      recipientInfos    RecipientInfos,

      macAlgorithm  MessageAuthenticationCodeAlgorithm,

      encapContentInfo  EncapsulatedContentInfo,

      mac MessageAuthenticationCode

}


MessageAuthenticationCode::= OCTET STRING
```

The value of the encapContentInfo component is described in clause 7.1 above.

The recipientInfos component is present in a value of type AuthenticatedData, but it contains an empty set of values of type RecipientInfo:

```
 RecipientInfos ::= SET SIZE(0) OF RecipientInfo
```

The tsa and extensions fields of TSTInfo shall be present. The tsa field is required for token verification. The extensions field is required to carry the identification of the mechanism used.

© ISO/IEC 2002 All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

8.2 MAC generation

The MAC calculation process computes a message authentication code (MAC) on the content being authenticated. The input to the MAC calculation process is the value of the encapContentInfo eContent OCTET STRING, that contains the DER-encoded value of a TSTInfo structure. Only the octets comprising the value of the eContent are input to the MAC algorithm; the tag and the length are omitted.

The input to the MAC calculation process includes the MAC input data defined above, and an authentication key that is secret. The details of the MAC calculation depend on the MAC algorithm employed by the TSA. The object identifier, along with any parameters, that specifies the MAC algorithm employed is carried on the macAlgorithm field:

```
MessageAuthenticationCodeAlgorithm::= MACAlgorithmIdentifier


MACAlgorithmIdentifier ::= AlgorithmIdentifier {{ MACAlgorithms }}


MACAlgorithms ALGORITHM ::= {
       --
   ... -- Expect additional MAC algorithms -
}
```

8.3 MAC verification

The input to the MAC verification process includes the input data (as described in the previous section) and the secret used by the TSA to authenticate the time-stamp token. The details of the MAC verification process depend on the MAC algorithm employed.

8.4 Token verification

The validity of a time-stamp token may be verified by checking that:

- the time-stamp token is syntactically well-formed;
- the value of the messageImprint component of the time-stamp token matches the value of a messageImprint evaluated at Tv over the document subject to scrutiny;
- the value of the messageImprint component of every ExtHash extension in the time-stamp token matches the value of a messageImprint evaluated at Tv over the document subject to scrutiny, and
- the policy under which the token is issued is acceptable for the intended usage.

Furthermore, the verifier shall carry out a verification protocol exchange with the issuing TSA (see ISO/IEC 18014-1) using a data integrity and data origin authentication protected channel.

The TSA must be ready to be audited by third parties, under prior agreement, to certify appropriate security of activity logs and secret keys.

© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

9 Time-stamps using archiving

In this mechanism the TSA returns a time-stamp token that only has reference information to bind the time-stamp to the messageImprint in the time-stamp token. The TSA archives locally enough information to verify that the time-stamp is correct.

The TSA archives may consist of time-stamp tokens using any mechanism, activity logs, transmission logs, etc.

When using this mechanism, the TSA is needed to carry out the verification, and the TSA has to be thoroughly trusted since there is no external evidence that might detect fraud.

NOTE: In this scenario the TSA is playing the role of an Electronic Notary. Guidance on the use and management of this kind of Trusted Third Party may be found in ISO/IEC TR 14516.

Every exchange of information between the different actors (requester, verifier, and TSA) requires data integrity and data origin authentication protection. This protection may be provided by any means, for example over a secured channel, or using public keys that do not need to last longer than the transaction lapse.

In order to use this mechanism, the requester will use the ExtMethod extension that is specified in the extensions field of the time-stamp request, with the following object identifier:

```
tsp-itm-archive OBJECT IDENTIFIER::= { tsp-itm archive(3) }
```

9.1 TSA response

The TimeStampToken is a sequence type that contains two components, contentType and content, which are defined in terms of the &id and &Type fields of the CONTENT information object class. Both components are constrained by the information object set Contents, which contains one object for each time-stamp mechanism, such as the object

```
  { ETSTInfo IDENTIFIED BY id-data }
```

For the archival time-stamp mechanism, this constraint on the components of TimeStampToken requires that the contentType component contain the value id-data, and that the content component contain a value of type ETSTInfo.

The object identifier value id-data is defined as:

```
id-data OBJECT IDENTIFIER ::= {

    iso(1) member-body(2) us(840)

    rsadsi(113549) pkcs(1) pkcs7(7) 1 }
```

The content contains the DER-encoded value of a TSTInfo structure.

The tsa and extensions fields of TSTInfo shall be present. The tsa field is required for token verification. The extensions field is required to carry the identification of the mechanism used.

9.2 Token verification

The validity of a time-stamp token may be verified by checking that:

© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

- the time-stamp token is syntactically well-formed;
- the value of the messageImprint component of the time-stamp token matches the value of a messageImprint evaluated at Tv over the document subject to scrutiny;
- the value of the messageImprint component of every ExtHash extension in the time-stamp token matches the value of a messageImprint evaluated at Tv over the document subject to scrutiny, and
- the policy under which the token is issued is acceptable for the intended usage.

Furthermore, the verifier shall carry out a verification protocol exchange with the issuing TSA (see ISO/IEC 18014-1) using a data integrity and data origin authentication protected channel.

The TSA must be ready to be audited by third parties, under prior agreement, to certify appropriate security of activity logs.

© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Annex A
(normative)
ASN.1 Module for time-stamping

This module contains correct ASN.1 based on the current ASN.1 standards that has passed successfully through a reliable syntax checker used by the ITU-T ASN.1 Project.

The syntax is according to ISO/IEC 8824-1: 1998 | X.680: ITU-T Recommendation X.680 (1997) and is bits-on-the-line compatible with [RFC3161].

```
TimeStampProtocol-2 {

      iso(1) standard(0) time-stamp(18014) modules(0) part2(2) }

      DEFINITIONS IMPLICIT TAGS ::= BEGIN


-- EXPORTS All; --


IMPORTS


-- ISO/IEC 9594-8 | ITU-T Rec. X.509 AuthenticationFramework --


      EXTENSION, Extensions

            FROM AuthenticationFramework {

                  joint-iso-itu-t ds(5) module(1) authenticationFramework(7) 4 }


-- ISO/IEC 9594-8 | ITU-T Rec. X.509 CertificateExtensions --


      GeneralName

            FROM CertificateExtensions {

                  joint-iso-itu-t ds(5) module(1) certificateExtensions(26) 4 }


      AuthenticatedData, SignedData

            FROM CryptographicMessageSyntax {

                  iso(1)  member-body(2)  us(840)  rsadsi(113549)  pkcs(1)  pkcs-9(9)
                  smime(16) modules(0) cms(1) };
```

© ISO/IEC 2002 All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
TimeStampReq ::= SEQUENCE {
      version       Version,
      messageImprint    MessageImprint,
      reqPolicy         TSAPolicyId OPTIONAL,
      nonce             Nonce OPTIONAL,
      certReq       BOOLEAN DEFAULT FALSE,
      extensions        [0] Extensions OPTIONAL
}

MessageImprint ::= SEQUENCE {
      hashAlgorithm     DigestAlgorithmIdentifier,
      hashedMessage     OCTET STRING
}

DigestAlgorithmIdentifier ::=
      AlgorithmIdentifier {{ DigestAlgorithms }}

DigestAlgorithms ALGORITHM ::= {
      { OID sha1 PARMS NULL },
       --
      ... -- Expect additional digest algorithms --
}

TSAPolicyId ::= POLICY.&id({TSAPolicies})

TSAPolicies POLICY ::= {
        --
      ...  -- Any supported TSA policy --
}
```


--`,,`-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO/IEC 2002 – All rights reserved
Not for Resale

```
TimeStampResp ::= SEQUENCE {
      status        PKIStatusInfo,
      timeStampToken    TimeStampToken OPTIONAL
}

PKIStatusInfo ::= SEQUENCE {
      status        PKIStatus,
      statusString      PKIFreeText OPTIONAL,
      failInfo          PKIFailureInfo OPTIONAL
}

PKIStatus ::= INTEGER {
      -- the request is completely granted:
      granted                 (0),

      -- modifications were necessary,
      -- the requester is responsible for asserting the differences:
      grantWithMods           (1),

      -- the request could not be fulfilled,
      -- the failure code delivers additional information:
      rejection               (2),

      -- the request is not processed,
      -- the requester receives a receipt that the request has been received:
      waiting                 (3),

      -- a revocation is imminent:
      revocationWarning       (4),
```

© ISO/IEC 2002 All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
      -- notification that a revocation has occurred
      revocationNotification  (5)
}

PKIFreeText ::= SEQUENCE SIZE(1..MAX) OF UTF8String

PKIFailureInfo ::= BIT STRING {
      badAlg              (0),   -- unrecognized or unsupported Algorithm Identifier
      badRequest          (2),   -- transaction not permitted or supported
      badDataFormat       (5),   -- data submitted has the wrong format
      timeNotAvailable    (14),  -- the TSA's time source is not available
      unacceptedPolicy    (15),  -- the requested TSA policy is not supported
      unacceptedExtension (16),  -- the requested extension is not supported,
      addInfoNotAvailable (17),  -- the additional information requested
                                 -- could not be understood or is not available,
      systemFailure       (25)   -- system failure
}

TimeStampToken ::= SEQUENCE {
      contentType       CONTENT.&id({Contents}),
      content    [0] EXPLICIT CONTENT.&Type({Contents}{@contentType})
}

Contents CONTENT ::= {
      time-stamp-mechanism-signature  |
      time-stamp-mechanism-MAC        |
      time-stamp-mechanism-archival,
       --
      ...  -- Expect additional time-stamp mechanisms --
```


--`,,`-`-`,,`,,`,`,,`---
© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
}

-- Time-stamp mechanism information objects --

time-stamp-mechanism-signature CONTENT ::=
      { SignedData IDENTIFIED BY id-signedData }

time-stamp-mechanism-MAC CONTENT ::=
      { AuthenticatedData IDENTIFIED BY id-ct-authData }

time-stamp-mechanism-archival CONTENT ::=
      { ETSTInfo IDENTIFIED BY id-data }

ETSTInfo ::=
      OCTET STRING (CONTAINING TSTInfo ENCODED BY der)

TSTInfo ::=  SEQUENCE {
      version       Version,
      policy        TSAPolicyId,
      messageImprint    MessageImprint,
      serialNumber      SerialNumber,
      genTime       GeneralizedTime,
      accuracy          Accuracy OPTIONAL,
      ordering          BOOLEAN DEFAULT FALSE,
      nonce             Nonce OPTIONAL,
      tsa           [0] EXPLICIT GeneralName OPTIONAL,
      extensions        [1] Extensions OPTIONAL
}

Version ::= INTEGER { v1(1) }
```

--`,,`-`-`,,`,,`,`,,`---

© ISO/IEC 2002 All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
SerialNumber ::= INTEGER  -- Expect large values

Accuracy ::= SEQUENCE {
      seconds    INTEGER  OPTIONAL,
      millis     [0] INTEGER(1..999) OPTIONAL,
      micros     [1] INTEGER(1..999) OPTIONAL
}
 (ALL EXCEPT({ -- no components present -- }))

Ordering ::= BOOLEAN

Nonce ::= INTEGER  -- Expect large values

-- Time-stamping extensions --

TSExtensions EXTENSION ::= {
      extHash    |
      extMethod  |
      extRenewal,
       --
      ...  -- Expect additional extensions --
}

extHash EXTENSION ::= {
      SYNTAX ExtHash IDENTIFIED BY tsp-ext-hash
}

ExtHash ::= SEQUENCE SIZE(1..MAX) OF MessageImprint

extMethod EXTENSION ::= {
```


© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
      SYNTAX ExtMethod IDENTIFIED BY tsp-ext-meth
}

ExtMethod ::= SEQUENCE SIZE(1..MAX) OF Method

Method ::= METHOD.&id({Methods})

Methods METHOD ::= {
      --
      ...  -- Any time-stamping method  --
}

extRenewal EXTENSION ::= {
      SYNTAX ExtRenewal IDENTIFIED BY tsp-ext-renewal
}

ExtRenewal ::= TimeStampToken

EncapsulatedContentInfo::= SEQUENCE {
      eContentType  CONTENT.&id({EContents}),
      eContent      [0] EXPLICIT CONTENT.&Type({EContents} {@eContentType})
}

EContents CONTENT ::= {
      { ETSTInfo IDENTIFIED BY id-ct-TSTInfo },
      --
      ...     -- Expect additional content types --
}

-- Supporting definitions
```

--`,,`-`-`,,`,,`,`,,`---
© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
AlgorithmIdentifier { ALGORITHM:IOSet } ::= SEQUENCE {
      algorithm     ALGORITHM.&id({IOSet}),
      parameters    ALGORITHM.&Type({IOSet}{@algorithm})  OPTIONAL
}

ALGORITHM ::= CLASS {
      &id    OBJECT IDENTIFIER UNIQUE,
      &Type      OPTIONAL
}
      WITH SYNTAX { OID &id [PARMS &Type] }

CONTENT ::= TYPE-IDENTIFIER  -- ISO/IEC 8824-2, Annex A

OIDS ::= CLASS {
      &id OBJECT IDENTIFIER UNIQUE
}
      WITH SYNTAX { OID &id }

tsp OBJECT IDENTIFIER ::= { iso(1) standard(0) time-stamp(18014) }

POLICY ::= OIDS  -- Supported TSA policies

METHOD ::= OIDS  -- TSA Methods

tsp-itm OBJECT IDENTIFIER ::= { tsp itm(2) }
tsp-itm-ds OBJECT IDENTIFIER::= { tsp-itm ds(1) }
tsp-itm-mac OBJECT IDENTIFIER::= { tsp-itm mac(2) }
tsp-itm-archive OBJECT IDENTIFIER::= { tsp-itm archive(3) }
```


© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
tsp-ext OBJECT IDENTIFIER ::= { tsp ext(1) }

tsp-ext-hash OBJECT IDENTIFIER ::= { tsp-ext 1 }

tsp-ext-meth OBJECT IDENTIFIER ::= { tsp-ext 2 }

tsp-ext-renewal OBJECT IDENTIFIER ::= { tsp-ext 3 }


der OBJECT IDENTIFIER ::= {

      joint-iso-itu-t asn1(1) ber-derived(2) distinguished-encoding(1) }


sha1 OBJECT IDENTIFIER ::= {

      iso(1) identified-organization(3) oiw(14) secsig(3) 2 26 }


pkcs OBJECT IDENTIFIER ::= {

      iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)

}


pkcs-7 OBJECT IDENTIFIER ::= { pkcs pkcs7(7) }

pkcs-9 OBJECT IDENTIFIER ::= { pkcs pkcs9(9) }


id-data OBJECT IDENTIFIER ::= { pkcs-7 data(1) }

id-signedData OBJECT IDENTIFIER ::= { pkcs-7 signedData(2) }


id-ct-authData OBJECT IDENTIFIER ::= { pkcs-9 smime(16) ct(1) 2 }

id-ct-TSTInfo OBJECT IDENTIFIER ::= { pkcs-9 smime(16) ct(1) 4 }


id-kp-timeStamping OBJECT IDENTIFIER ::= {
      iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5)
      pkix(7) kp(3) timestamping(8)
}


-- verification of a timestamp token --


VerifyReq ::=    SEQUENCE {
```

© ISO/IEC 2002 All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

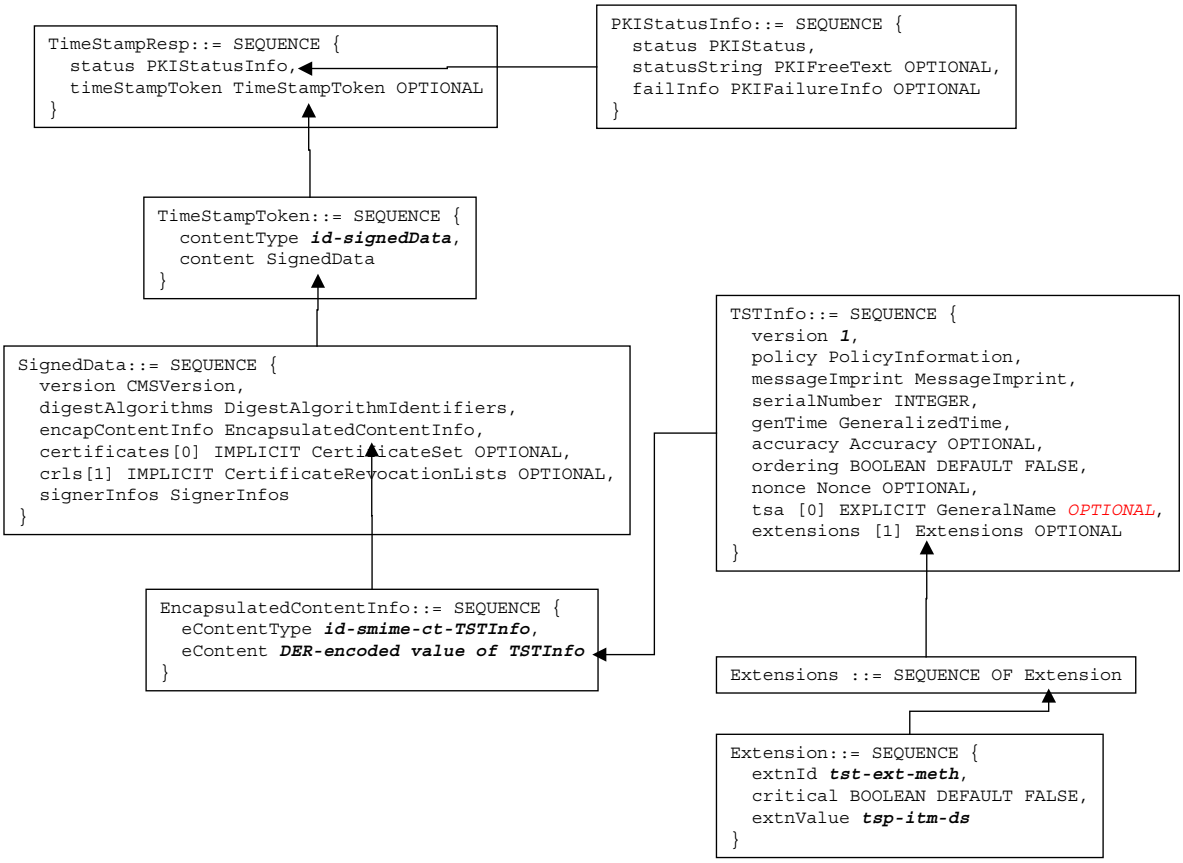
```
      version    Version,
      tst        TimeStampToken,
      requestID     RequestID OPTIONAL
}

VerifyResp ::= SEQUENCE {
      version    Version,
      status     PKIStatusInfo,
      tst        TimeStampToken,
      requestID     RequestID OPTIONAL
}

RequestID ::= OCTET STRING (SIZE(1..MAX))

END  -- TimeStampProtocol-2 --
```


© ISO/IEC 2002 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Annex B
(informative)
Data Structures

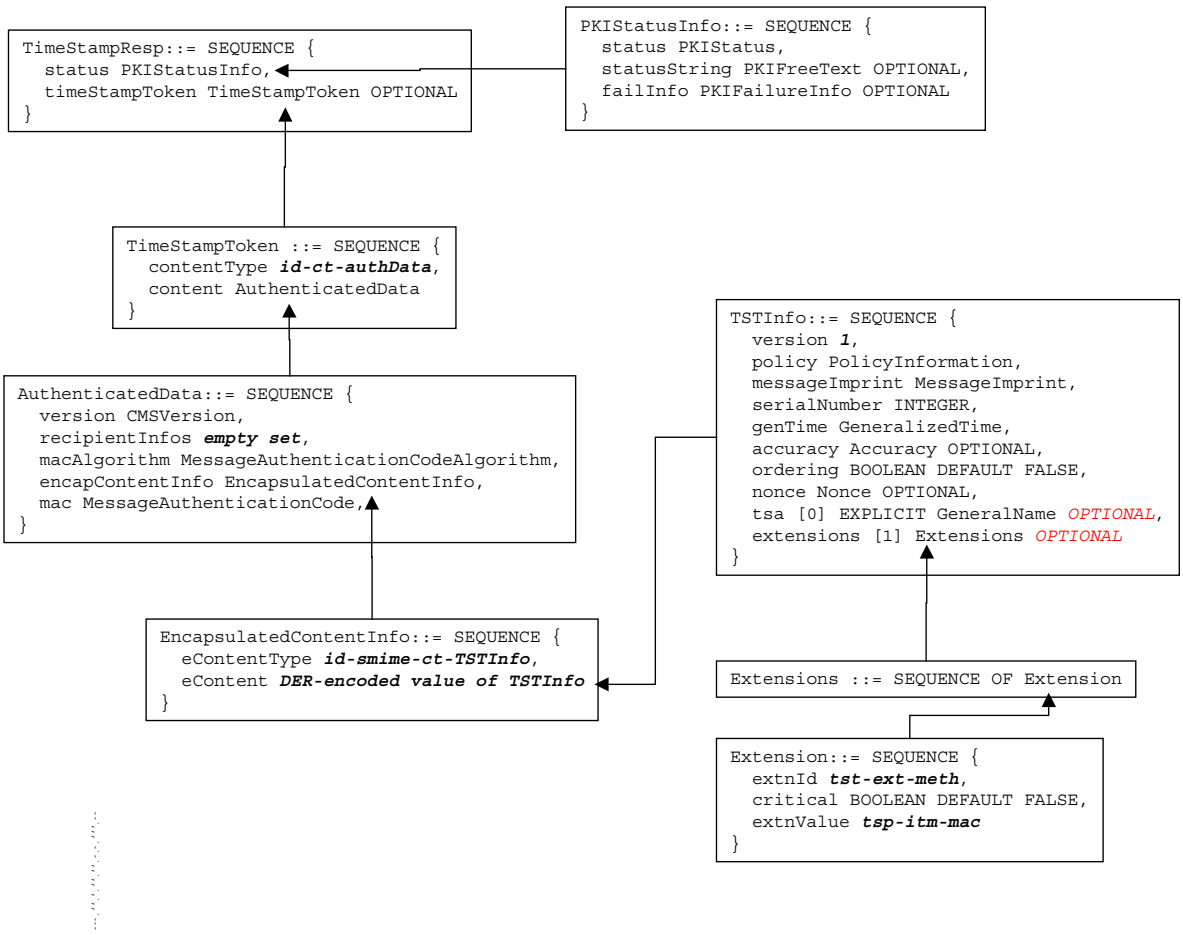
The following diagrams complement the ASN.1 descriptions in the normative text.

Time-stamps using digital signatures



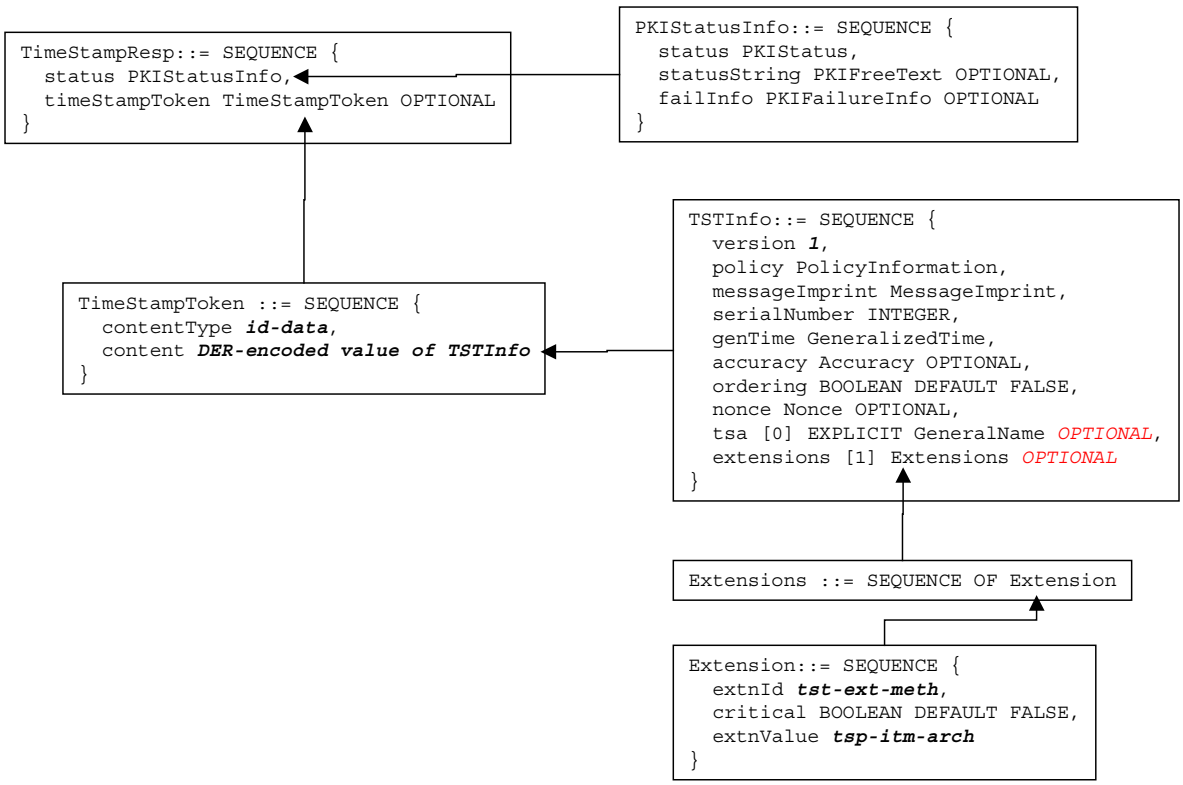
© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Time-stamps using message authentication codes



Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO/IEC 2002 – All rights reserved
Not for Resale

Time-stamps using archiving



© ISO/IEC 2002 All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Bibliography

[RFC2459] IETF RFC 2459: Internet X.509 Public Key Infrastructure Certificate and CRL Profile

[RFC3161] IETF RFC 3161: Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)

© ISO/IEC 2002 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

ICS 35.040

Price based on 28 pages

--`,,`-`-`,,`,,`,`,,`---

© ISO/IEC 2002 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale