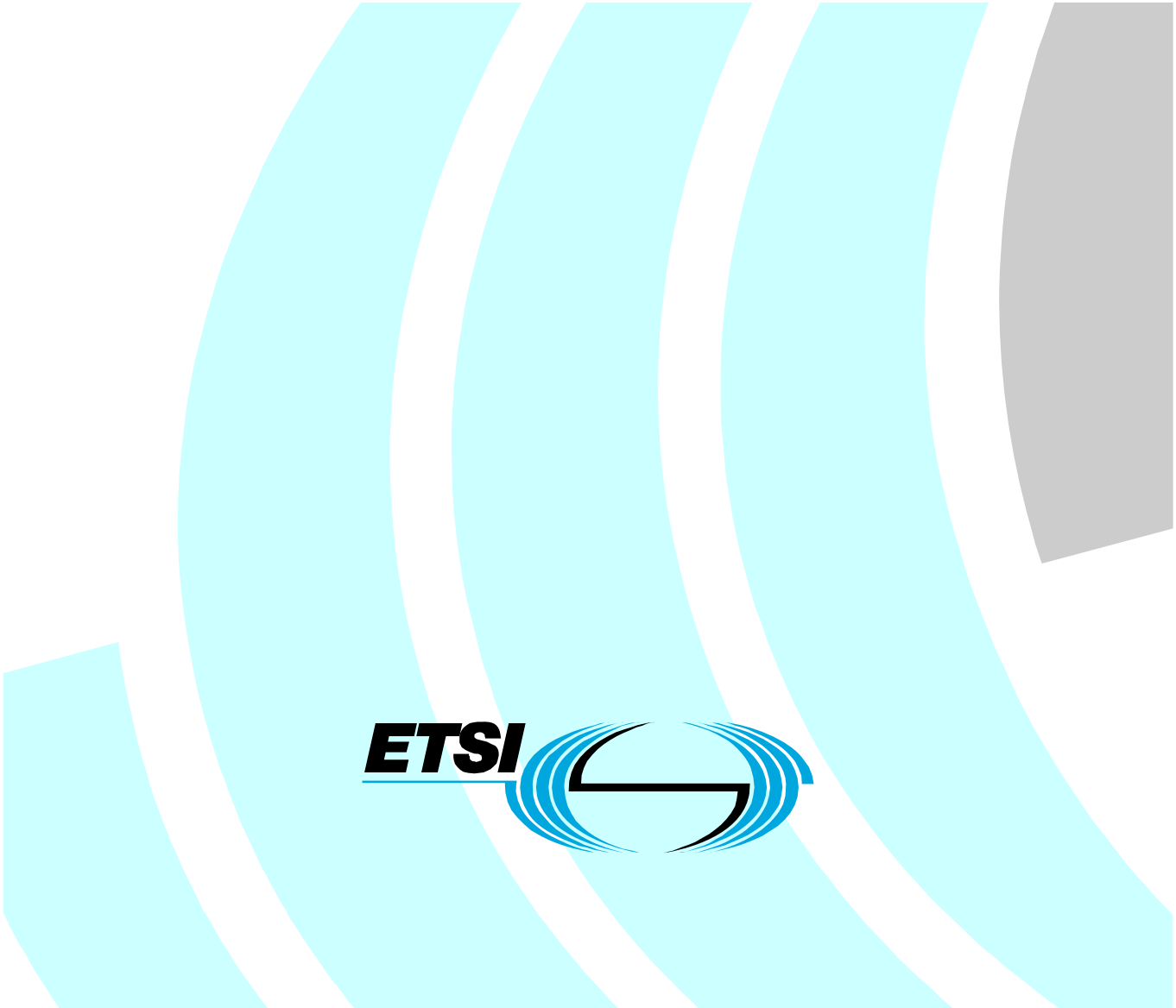
ETSI TS 102 231 V3.1.2 (2009-12)

Technical Specification

Electronic Signatures and Infrastructures (ESI); Provision of harmonized Trust-service status information



Reference
RTS/ESI-000083

Keywords
e-commerce, electronic signature, security, trust services

ETSI

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

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

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Content

Intellectual Property Rights 9
Foreword 9
Acknowledgement 9
Introduction 9
1 Scope 11
2 References 11
2.1 Normative references 11
2.2 Informative references 13
3 Definitions and abbreviations 13
3.1 Definitions 13
3.2 Abbreviations 14
4 Trust-service status information 15
5 Trust-service Status List structure 16
5.1 Structure of the Trust-service Status List 16
5.1.1 Trust-service Status List information 16
5.1.2 Logical model 17
5.1.3 Language support 19
5.1.4 Date-time indication 19
5.1.5 Use of Uniform Resource Identifiers 19
5.1.6 Value of Country Code fields 20
5.1.7 TSL Format 20
5.2 Trust-service Status List tag 20
5.2.1 TSL Tag 20
5.3 Scheme information 20
5.3.1 TSL version identifier 20
5.3.2 TSL sequence number 20
5.3.3 TSL type 21
5.3.4 Scheme operator name 21
5.3.5 Scheme operator address 21
5.3.5.1 Scheme operator postal address 21
5.3.5.2 Scheme operator electronic address 22
5.3.6 Scheme name 22
5.3.7 Scheme information URI 22
5.3.8 Status determination approach 23
5.3.9 Scheme type/community/rules 23
5.3.10 Scheme territory 23
5.3.11 TSL policy/legal notice 24
5.3.12 Historical information period 24
5.3.13 Pointers to other TSLs 24
5.3.13.1 Additional information field 25
5.3.14 List issue date and time 25
5.3.15 Next update 26
5.3.16 Distribution points 26
5.3.17 Scheme extensions 26
5.3.18 List of Trust Service Providers 26
5.4 TSP information 27
5.4.1 TSP name 27
5.4.2 TSP trade name 27
5.4.3 TSP address 27
5.4.3.1 TSP postal address 27
5.4.3.2 TSP electronic address 28
5.4.4 TSP information URI 28

5.4.5 TSP information extensions 28
5.4.6 List of services 28
5.5 Service information 29
5.5.1 Service type identifier 29
5.5.2 Service name 30
5.5.3 Service digital identity 30
5.5.4 Service current status 30
5.5.5 Current status starting date and time 32
5.5.6 Scheme service definition URI 32
5.5.7 Service supply points 33
5.5.8 TSP service definition URI 33
5.5.9 Service information extensions 33
5.5.10 Service approval history 33
5.6 Service approval history information 33
5.6.1 Service type identifier 33
5.6.2 Service name 34
5.6.3 Service digital identity 34
5.6.4 Service previous status 34
5.6.5 Previous status starting date and time 34
5.6.6 Service information extensions 34
5.7 Signature 34
5.7.1 Signed TSL 34
5.7.2 Scheme identification 35
5.7.3 Signature algorithm identifier 35
5.7.4 Signature value 35
5.8 TSL extensions 35
5.8.1 expiredCertsRevocationInfo Extension 36
5.8.2 additionalServiceInformation Extension 36

6 Operations 37
6.1 TSL publication 37
6.1.1 Transport Protocols 37
6.1.1.1 LDAP transport 37
6.1.1.1.1 Attributes and Object class definition 37
6.1.1.2 HTTP-Transport 39
6.1.1.2.1 HTTP-Media Type 39
6.1.1.3 FTP-Transport 39
6.1.1.4 Email Transport 39
6.1.1.4.1 Content-Types 39
6.1.1.4.2 Encoding considerations 39
6.1.1.5 MIME registrations 39
6.2 TSL Signer Certificate 40
6.3 TSL Distribution Points 40

Annex A (normative): Implementation in ASN.1 41

A.1 Structure of the Trust-service Status List 41
A.1.1 ASN.1 versioning 41
A.1.2 Basic types 41
A.1.2.1 The `NonEmptyURI` type 41
A.1.2.2 The `LanguageTag` type 42
A.1.2.3 The `CountryCode` type 42
A.1.2.4 The `MultiLangPointer` type 42
A.1.2.5 The `MultiLangString` type 42
A.1.2.6 The `PhysicalAndElectronicAddresses` type 42
A.1.3 General Structure 43

A.2 Scheme information fields 43
A.2.1 The tSLTag field 43
A.2.2 The version field 43
A.2.3 The sequenceNumber field 44
A.2.4 The tSLType field 44

A.2.5 The schemeOperatorName field 44
A.2.6 The schemeOperatorAddress field 44
A.2.7 The schemeName field 44
A.2.8 The schemeInformationURI field 44
A.2.9 The statusDeterminationApproach field 44
A.2.10 The schemeTypeCommunityRules field 45
A.2.11 The schemeTerritory field 45
A.2.12 The tSLpolicy field 45
A.2.13 The historicalInformationPeriod field 45
A.2.14 The pointersToOtherTSLs field 45
A.2.14.1 otherQualifier 46
A.2.15 The listIssueDateTime field 46
A.2.16 The nextUpdate field 47
A.2.17 The distributionPoints field 47
A.2.18 The schemeExtensions field 47
A.2.19 The tSPlist field 47

A.3 TSP information fields 48
A.3.1 The tSPname field 48
A.3.2 The tradeName field 48
A.3.3 The tSPaddress field 48
A.3.4 The tSPinformationURI field 48
A.3.5 The tSPextensions field 48
A.3.6 The listOfServices field 48

A.4 TSP service information fields 49
A.4.1 The serviceType field 49
A.4.2 The serviceName field 49
A.4.3 The serviceDigitalIdentity field 49
A.4.4 The currentServiceStatus field 50
A.4.5 The currentStatusStartingTime field 50
A.4.6 The schemeURI field 50
A.4.7 The tspURI field 50
A.4.8 The serviceSupplyPoints field 50
A.4.9 The srvcExtensions field 51
A.4.10 The serviceApprovalHistory field 51

A.5 Service history information fields 51
A.5.1 The serviceType field 51
A.5.2 The serviceName field 51
A.5.3 The serviceDigitalIdentity field 51
A.5.4 The previousServiceStatus field 51
A.5.5 The previousStatusStartingTime field 51
A.5.6 The srvcExtensions field 52

A.6 TSL signature fields 52
A.6.1 The signedTSL field 52
A.6.2 The scheme operator identifier 52
A.6.2.1 ESS signing certificate attribute 53
A.6.3 Algorithms and parameters 53

A.7 Extensions defined in the present document 53
A.7.1 TSL Extensions OID 54
A.7.2 TSLDistributionPoints 54
A.7.3 ExpiredCertsRevocationInfo 54
A.7.4 AdditionalServiceInformation 54

Annex B (normative): Implementation in XML 55

B.1 Structure of the Trust-service Status List 55
B.1.1 General Rules 55
B.1.2 XML-namespace and basic types 55
B.1.2.1 The `InternationalNamesType` and `MultiLangString` Types 55
B.1.2.2 The `AddressType` Type 56

B.1.2.3 The PostalAddresses Element 56
B.1.2.4 The ElectronicAddressType Type 57
B.1.2.5 Types for managing the extensions 57
B.1.2.6 Types for URIs 58
B.1.3 The TrustServiceStatusList element 58
B.1.3.1 The TSLTag attribute 58

B.2 The SchemeInformation element 59
B.2.1 The TSLVersionIdentifier element 59
B.2.2 The TSLSequenceNumber element 59
B.2.3 The TSLType element 59
B.2.4 The SchemeOperatorName element 59
B.2.5 The SchemeOperatorAddress element 59
B.2.6 The SchemeName element 60
B.2.7 The SchemeInformationURI element 60
B.2.8 The StatusDeterminationApproach element 60
B.2.9 The SchemeTypeCommunityRules element 60
B.2.10 The SchemeTerritory element 60
B.2.11 The PolicyOrLegalNotice element 60
B.2.12 The HistoricalInformationPeriod element 61
B.2.13 The PointersToOtherTSL element 61
B.2.13.1 Already identified contents of OtherInformation element 61
B.2.14 The ListIssueDateTime element 62
B.2.15 The NextUpdate element 62
B.2.16 The SchemeExtensions element 62
B.2.17 The DistributionPoints element 62
B.2.18 The TrustServiceProviderList element 62

B.3 The TSPInformation element 63
B.3.1 The TSPName element 63
B.3.2 The TSPTradeName element 63
B.3.3 The TSPAddress element 63
B.3.4 The TSPInformationURI element 63
B.3.5 The TSPInformationExtensions element 63
B.3.6 The TSPServices element 63

B.4 The ServiceInformation element 64
B.4.1 The ServiceTypeIdentifier element 64
B.4.2 The ServiceName element 64
B.4.3 The ServiceDigitalIdentity element 64
B.4.4 The ServiceStatus element 65
B.4.5 The StatusStartingTime element 65
B.4.6 The SchemeServiceDefinitionURI element 65
B.4.7 The ServiceSupplyPoints element 65
B.4.8 The TSPServiceDefinitionURI element 65
B.4.9 The ServiceInformationExtensions element 65
B.4.10 The ServiceHistory element 66

B.5 The ServiceHistory type 66

B.6 The Signature element 66
B.6.1 The scheme identification 67
B.6.1.1 The scheme operator identifier in XAdES signatures 67
B.6.2 Algorithm and parameters 67

B.7 Elements and types for TSL extensions 67
B.7.1 The ExpiredCertsRevocationInfo element 68
B.7.2 The AdditionalServiceInformation element 68

Annex C (normative): ASN.1 and XML files 69

C.1 Electronic attachment 69
C.2 ASN.1 module 69
C.3 XML schema 69
C.4 LDAP schema 70
Annex D (normative): Registered Uniform Resource Identifiers 71
D.1 URIs registered within the present document 71
D.2 ETSI Common Domain URIs 72
D.3 Registering additional URIs 74
Annex E (normative): Implementation notes for multilingual support 75
E.1 Multilingual character string 75
E.2 Multilingual pointer 75
E.3 Overall requirements 76
Annex F (informative): TSL Signing considerations 77
F.1 Signing application maturity 77
F.2 CMS/ESS and CAdES 77
F.3 XML 77
F.4 PDF/A 78
Annex G (informative): Management and Policy considerations 79
G.1 Change of scheme administrative information 79
G.2 Trust-service identification 79
G.3 Change of trust-service status 79
G.4 Amendment response times 80
G.5 On-going verification of authenticity 80
G.6 Upon a scheme's cessation of operations 80
G.7 User reference to TSL 81
G.8 Reliance upon hard-copy TSL information 81
G.9 TSL size 81
Annex H (informative): Locating and Authenticating a TSL 82
H.1 Introduction 82
H.2 Locating a TSL 82
H.2.1 TSL location models 82
H.2.1.1 Bound information 82
H.2.1.2 Linked information 83
H.2.1.3 De-coupled information 83
H.2.2 Searching for a TSL 83
H.2.2.1 Same-scheme searching 83
H.2.2.2 Known scheme searching 84
H.2.2.3 "Blind" (unknown) scheme searching 84
H.2.2.3.1 Structure of the HTML-Page 84
H.2.2.3.2 Example 85
H.3 Authenticating a TSL 86

H.4 Trusting a TSL 86
H.5 Replicating TSLs 87
H.6 Security issues 87
H.7 Implications for authentication of Trust Service Tokens 88

Annex I (informative): General TSL usage 89

I.1 Introduction 89
I.2 Generic TSL usage 89
I.2.1 Trusted Lists 89
I.2.2 Trust service status as legal evidence 89
I.2.3 Checking for anomalous status before accepting a credential 90
I.2.4 Cross-certification status confirmation 90
I.3 TSLs used to list other schemes 90
I.3.1 Hierarchical relationships 90
I.3.2 A collection of TSLs 91
I.3.3 Schemes applying common rules 91
I.3.4 Schemes trusted by a vendor community 92
I.3.5 Industrial trading consortium 92

Annex J (informative): TSL manual/auto field usage 93

Annex K: Void 94

Annex L (normative): URIs and extensions used for the EU Member States' national Trusted List of supervised/accredited Certification Service Providers 95

L.1 Introduction 95
L.2 eSig Directive URIs 95
L.3 eSig Directive Extensions 101
L.3.1 Qualifications Extension 101
L.3.1.1 QualificationElement 101
L.3.1.2 CriteriaList 101
L.3.1.2.1 KeyUsage 102
L.3.1.2.2 PolicySet 102
L.3.1.2.3 OtherCriteria 102
L.3.1.3 Qualifier 103
L.3.2 TakenOverBy Extension 103

Annex M (informative): Changes since the last major version 104

M.1 Changes from v2.1.1 to v3.1.1 104
M.2 Changes from v3.1.1 to v3.1.2 104
History 106

Intellectual Property Rights

IPRs essential or potentially essential to the present document 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 (http://webapp.etsi.org/IPR/home.asp).

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.

Foreword

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

Acknowledgement

The building blocks of the Localization, Access and Authentication Mechanisms described in this Technical Specification were submitted by UNINFO, the Italian standardization body for ICT, federated to UNI, Italian member body of CEN and ISO.

Members of the team working on the ***CROBIES study*** *(Cross-Border Interoperability of eSignature)* provided valuable input for version 3.1.1.

Introduction

The purpose of a Trust-service Status List (TSL), and hence of the present document, is to provide a harmonized way in which assessment schemes having an oversight role with regards to trust services and their providers (trust service providers - TSPs) can publish information about the services and TSPs which they currently oversee, or indeed (through the provision of historical information) have overseen. An assessment scheme operator may also use the TSL to only refer to other assessment schemes, in which case the services of these assessment scheme operators are considered as a specific type of trust service (see clause I.3).

The present document is based upon the reasoning that it will enhance the confidence of parties relying on certificates or other services related to electronic signatures if they had access to information that would allow them to know whether a given TSP was operating under the approval of any recognized scheme at the time of providing their services and of any dependent transaction that took place.

The assurance provided by information available within a TSL is intended to serve as a secondary source of trust, rather than a primary source of trust which might be derived by parsing a certificate chain. The present document is not intended to be a replacement for certificate chains and the assurance which may be obtained from parsing them to establish the validity of certificates (or other forms of trust service tokens) associated with providers of trust services of any kind.

The information should be available for a wide range of services and schemes, including the use of Qualified Certificates. The importance of this information is especially significant for cross-domain and international transactions. This information should preferably be accessible using an on-line protocol, although accessibility both off-line and on-line should be possible.

Entities having such an oversight role could be supervisory systems or voluntary approval schemes as defined in Directive 1999/93/EC [1] (see note), similar schemes established by other sovereign states or economies (e.g. certain government e-authentication frameworks), and those established by specific industry sectors or for international promotion of trust services.

NOTE: This refers in particular to the Trusted Lists to be established, published and maintained by every European Union Member State and that consist in the Member State's "Supervision/Accreditation Status List of certification services from Certification Service Providers, which are supervised/accredited by the referenced Member State for compliance with the relevant provisions laid down in Directive 1999/93/EC". Those Trusted Lists (one single list per Member State) will comply with the present document requirements while making use of the URIs and extensions described in annex L.

All previous versions of the present document (as listed below) are to be considered as "historical", with effect from the publication date of this present version. Although there may remain in existence for some time TSLs which were created compliant to previous versions all future TSL's published should be conformant to the specifications set out in the present document. Parsers should be upgraded to accommodate the version defined herein whilst retaining their ability to parse previous versions where they continue to be used.

Changes to previous major version of the present document have been listed in annex M.

In any case, all TSLs that have been previously created according to version 2.1.1 and above of the present document remain compatible to this version (V3.1.2).

This version renders these previous versions **historical**:

- Version 1.1.1, downloadable from ETSI as file "ts_102231v010101p".
- Version 2.1.1, downloadable from ETSI as file "ts_102231v020101p".
- Version 3.1.1, downloadable from ETSI as file "ts_102231v030101p".

Any external reference listed in clause 2.1 is intended to be the last available version of the referenced document that is applicable taking into account the present document prescriptions.

1 Scope

The present document specifies a standard for a Trust-service Status List (TSL) which makes available trust service status information such that interested parties may determine whether a trust service is **or was** operating under the approval of any recognized scheme at either the time the service was provided, or **the time at which a transaction reliant on that service took place**.

The normative specification defines the structure and meaning of a TSL which fulfils these requirements and specifies the mechanisms to be used for locating, accessing and authenticating TSLs. In addition, the present document gives informative guidance for the management of and access to TSLs and the use of status information held within them. Within the present document the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [5].

The present document is applicable to assessment scheme operators responsible for the approval of trust services and to those who wish to rely on such information.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
  - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at http://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.

2.1 Normative references

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

[1] Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures.

[2] ETSI TS 101 733: "Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES)".

[3] IETF RFC 959: "File Transfer Protocol".

[4] IETF RFC 2045: "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies".

[5] IETF RFC 2119: "Key words for use in RFCs to indicate Requirement Levels".

[6] IETF RFC 2141: "URN Syntax".

[7] IETF RFC 4511: "Lightweight Directory Access Protocol (LDAP): The Protocol".

[8] IETF RFC 4517: "Lightweight Directory Access Protocol (LDAP): Syntaxes and Matching Rules".

[9] Void.

[10] IETF RFC 4519: "Lightweight Directory Access Protocol (LDAP): Schema for User Applications".

[11] IETF RFC 2368: "The mailto URL scheme".

[12] IETF RFC 2616: "Hypertext Transfer Protocol - HTTP/1.1".

[13] IETF RFC 2634: "Enhanced Security Services for S/MIME".

[14] IETF RFC 5322: "Internet Message Format".

[15] IETF RFC 3023: "XML Media Types".

[16] IETF RFC 5646: "Tags for Identifying Languages".

[17] IETF RFC 5280: "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[18] IETF RFC 3305: "Report from the Joint W3C/IETF URI Planning Interest Group: Uniform Resource Identifiers (URIs), URLs, and Uniform Resource Names (URNs): Clarifications and Recommendations".

[19] IETF RFC 3986: "Uniform Resource Identifiers (URI): Generic Syntax".

[20] IETF RFC 4050: "Using the Elliptic Curve Signature Algorithm (ECDSA) for XML Digital Signatures".

[21] ISO 3166-1: "Codes for the representation of names of countries and their subdivisions - Part 1: Country codes".

[22] ISO 8601: "Data elements and interchange formats - Information interchange - Representation of dates and times".

[23] ISO 10646: "Information technology - Universal Multiple-Octet Coded Character Set (UCS)".

[24] ITU-T Recommendation X.208: "Specification of Abstract Syntax Notation One (ASN.1)".

[25] ITU-R Recommendation TF.460-5: "Standard-frequency and time-signal emissions".

[26] ISO/IEC 9594-8:2005: "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks".

[27] Void.

[28] ITU-T Recommendation X.690: "Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)".

[29] W3C Recommendation (2002): "XHTMLTM 1.0 - The Extensible HyperText Markup Language (Second Edition) - A Reformulation of HTML 4 in XML 1.0".

[30] W3C Recommendation (2001): "XHTMLTM 1.1 - Module-based XHTML".

[31] W3C Recommendation (1999): "HTML 4.01 Specification".

[32] W3C Recommendation (2004): "XML Schema Part 2: Data types Second Edition".

[33] W3C Technical Report #20 Revision 7: "Unicode in XML and other Markup Languages".

[34] W3C Recommendation Second edition (2008): "XML-Signature Syntax and Processing".

[35] ETSI TS 101 903: "XML Advanced Electronic Signatures (XAdES)".

[36] IETF RFC 4055: "Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[37] IETF RFC 5652: "Cryptographic Message Syntax (CMS)".

[38] IETF RFC 5035: "Enhanced Security Services (ESS) Update: Adding CertID Algorithm Agility".

[39] ISO/IEC 6429: "Information technology -- Control functions for coded character sets".

[40] ISO/IEC 2022: "Information technology -- Character code structure and extension techniques".

[41] IETF RFC 3279: "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[42] IETF RFC 3370: "Cryptographic Message Syntax (CMS) Algorithms".

[43] ITU-T Recommendation X.509: "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks".

[44] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[45] ISO 19005-1: "Document management -- Electronic document file format for long-term preservation -- Part 1: Use of PDF 1.4 (PDF/A-1)".

[46] ETSI TS 102 778-3: "Electronic Signatures and Infrastructures (ESI);PDF Advanced Electronic Signature Profiles;Part 3: PAdES Enhanced - PAdES-BES and PAdES-EPES Profiles".

2.2 Informative references

The following referenced documents are not essential to the use of the present document but they assist the user with regard to a particular subject area. For non-specific references, the latest version of the referenced document (including any amendments) applies.

[i.1] ITU-T Recommendation X.680: "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation".

3 Definitions and abbreviations

3.1 Definitions

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

approval: assertion that a(n electronic trust) service, falling within the oversight of a particular scheme, has been either positively endorsed (active approval) or has received no explicit restriction since the time at which the scheme was aware of the existence of the said service (passive approval)

assessment scheme: any organized process of supervision, monitoring, approval or such practices that are intended to apply oversight with the objective of ensuring adherence to specific criteria in order to maintain confidence in the services under the scope of the scheme

(electronic) Trust Service (TS): service which enhances trust and confidence in electronic transactions (typically but not necessarily using cryptographic techniques or involving confidential material)

implementation specific: used throughout the present document and refers principally to the annexes A and B implementation specifications for ASN.1 and XML

NOTE: It does not mean that implementers of TSL applications have a free choice.

Qualified Certificate: public key certificate issued in accordance with the requirements of Directive 1999/93/EC [1]

scheme operator: body responsible for the operation and/or management of any kind of assessment scheme, whether they are governmental, industry or private, etc.

Trust Service Provider (TSP): body operating one or more **(electronic) Trust Services**

NOTE: This term is used in preference to - and with a broader application than - the term Certification-Service-Provider (CSP) used in Directive 1999/93/EC [1]. The term "Trust Service Provider" can also encompass TSL issuers, in which case the TSL can even lists only other TSL issuers.

Trust Service Token (TrST): physical or binary (logical) object generated or issued as a result of the use of a Trust Service

NOTE: Examples of binary Trust Service Tokens are: certificates, CRLs, Time Stamp Tokens, OCSP responses. Where the TSP is a scheme the TrSTs are the TSLs it issues. Physical tokens may be devices on which binary objects (tokens or credentials) are stored. Equally, a token may be the performance of an act and the generation of an electronic record, e.g. an insurance policy or share certificate.

Trusted List (TL): refers to a European Union Member State's "Supervision/Accreditation Status List of certification services from Certification Service Providers, which are supervised/accredited by the referenced Member State for compliance with the relevant provisions laid down in Directive 1999/93/EC"

3.2 Abbreviations

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

ASN.1	Abstract Syntax Notation One
BMP	Basic Multilingual Plane
CA	Certification Authority
CAdES	CMS Advanced Electronic Signature
CMS	Cryptographic Message Syntax
CR	Carriage Return
CRL	Certificate Revocation List
CSP	Certification Service Provider
DE	Germany
DER	Distinguished Encoding Rules
DIT	Directory Information Tree
DN	Distinguished Name
DSA	Digital Signature Algorithm
ECDSA	Elliptic Curve Signature Algorithm
EFTA	European Free Trade Association
ESS	Enhanced Security Services
EU	European Union
EUMS	European Member States
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
HU	Hungary
LDAP	Lightweight Directory Access Protocol
LF	Line Feed
NBCA	National PKI Bridge CA
OCSP	Online Certificate Status Protocol
OID	Object Identifier
PAdES	PDF Advanced Electronic Signature
PKC	Public Key Certificate
PKI	Public Key Infrastructure
QC	Qualified Certificate
RA	Registration Authority
REM	Registered Electronic Mail
RGS	Le Référentiel Général de Sécurité
RKA	Root Key Authority
RSA	Rivest, Shamir and Adleman (cryptographic algorithm)
SSCD	Secure Signature Creation Device
TAB	Tabulator

TDP	TSL Distribution Point
TL	Trusted List
TLS	Transport Layer Security
TOC	Trans-Oceanic Consortium
ToSch	TSL "of Schemes"
TrST	Trust Service Token
TSL	Trust-service Status List
TSP	Trust Service Provider
TST	Trust Service Token
UCS	Universal Character Set
URI	Uniform Resource Identifier
URN	Uniform Resource Name
UTC	Coordinated Universal Time
WWW	World Wide Web
XAdES	XML Advanced Electronic Signature
XML	eXtensible Markup Language

4 Trust-service status information

The present document specifies a standard for the provision of trust service status information and mechanisms for locating, accessing and authenticating that information. In recognition of the selection of a form of signed list as the basis for presentation of this information, the term Trust-service Status List (TSL) is adopted. Each assessment scheme (scheme operator) which maintains a TSL in accordance with the present document MUST comply with the format and semantics specified in clause 5. Each such assessment scheme MUST operate against specific criteria for determining the status of trust services which it recognizes: an assessment scheme operator could, therefore, operate more than one discrete scheme, according to different criteria it might apply for different purposes.

In addition to this "generic" type, TSLs can be issued also with the purpose of listing all schema operators belonging to a community or a federation (we indicate this type as "schemes"). We refer also to the schemes TSL issuer with the term "scheme operator" also in case it simply has the role to "list" all schema operators belonging to the community and not providing any status information on each schema operator. In this case the schema operator has no liability other than to provide a comprehensive list of all schema operators belonging to the community.

With regard to the information provided within a TSL, it should be noted that the present document addresses only the type, format and meaning of information which MAY be presented in a TSL and does not define how that information should be sourced, i.e. what steps the scheme operator takes to collect that information. Nor does it specify the criteria which assessment schemes should use to determine the status of any trust services falling within their remit - such criteria remain the responsibility of the scheme operators. Furthermore, it does not specify how any status or scheme-related information should be presented outside the context of a TSL, e.g. on schemes' websites.

Each assessment scheme adopting this TSL standard MUST be able to support the provision of status information in each of the following forms:

- Human readable in a format readily down-loadable and printable.
- Machine processable to allow automatic verification of status information.

The TSL specified by the present document enables any interested party to determine whether a trust service is or was operating under the approval of any recognized scheme at either the time the service was provided, or the time at which a transaction reliant on that service took place. In order to fulfil this requirement, Trust-service Status Lists MUST necessarily contain information from which it can be established whether the TSP's service was, at the time of the transaction, known by the assessment scheme operator and if so the status of the service, i.e. whether it was approved, suspended, cancelled, revoked, etc. The Trust-service Status List MUST therefore contain not only the service's current status, but also the history of its status. Because of this requirement upon it, the TSL MUST therefore be specified in a manner which can support both "positive approval" lists and "delinquents" lists, including historical information.

The TSL specified by the present document therefore has four major components, in a structured relationship. These components:

- provide information on the issuing scheme;
- identify the TSPs recognized by the scheme;

- indicate the service(s) provided by these TSPs and the current status of those service(s);
- indicate for each service the status history of that service.

The logic of the list is that, once the assessment scheme operator has become aware of the existence of the TSP (whether by some pro-active action on the part of the TSP or by the scheme's own supervision of the marketplace), the particular status as determined according to the scheme rules is either the present status of the TSP's service (i.e. only current status, no history) or is seamlessly followed by a sequence of one or more statuses (current status and history). Note that if a trust service was approved until a certain date/time and there was a period in between the expiry of the approval and the start of the re-approval, then a status identifier would provide the information for that interim period. The "interim status" would either be expired (i.e. voluntarily, by the TSP) or revoked (by the scheme, with reasons).

5 Trust-service Status List structure

This clause specifies the Trust-service Status List structure. Each of the fields within the TSL is described to a level of detail sufficient to permit any assessment scheme operator to implement a standardized TSL, consistent with any other TSL conformant to the present document, with specified values, meanings and interpretations given for each field. Whether the inclusion of a field is REQUIRED or OPTIONAL is indicated.

5.1 Structure of the Trust-service Status List

The logical model of the Trust-service Status List is shown in figure 1. It has four logical component parts, all but the first of which MAY be replicated as required.

The list commences with key information about the list itself and the nature of the scheme which has determined the information found in, and through, the list (component 1). The specified set of information MUST include a pointer (URI) to details of the scheme and how its operator MAY be contacted. Whilst the objective has been to keep the size of the TSL to the minimum consistent with its purpose and the requirements placed upon it, certain key information which one would expect to be found in the scheme details MUST be provided directly within the TSL itself so as to facilitate either easy recognition and contact with the scheme or machine processing.

Following this scheme-related information there comes information relating to the Trust Service Providers (TSPs) whose services are within the scope of the scheme (component 2), and for each of those TSPs, the details of their specific trust services whose current status is recorded within the TSL (component 3). For each service, any available historical status information is recorded (component 4). The number of TSPs, of services per TSP, and of history sections per service is unbounded.

The TSL is a signed list for authentication purposes and is tagged to facilitate identification for electronic searches. The structure of the TSL is described in the following clauses by each component part and its fields.

Where fields are defined as being of type URI, implementers MAY in future use the URN (a particular subset of URIs that provides with persistent names and whose syntax is specified by RFC 2141 [6]) once such names become technically resolvable. Until such time implementers should use other URI types whose general syntax is specified by RFC 3986 [19]. See RFC 3305 [18] for clarification about URI and URN.

5.1.1 Trust-service Status List information

Description:

This field represents all the structured information and SHALL contain the following:

a) A Trust-service Status List tag to facilitate identification of the TSL for electronic searches. The contents of the tag are specified in clause TSL Tag5.2.1.

b) Scheme information, as specified in clause 5.3.

c) A sequence of fields holding information on the TSPs that the scheme oversees. This sequence is OPTIONAL. The contents of the TSP information field are specified in clause 5.4.

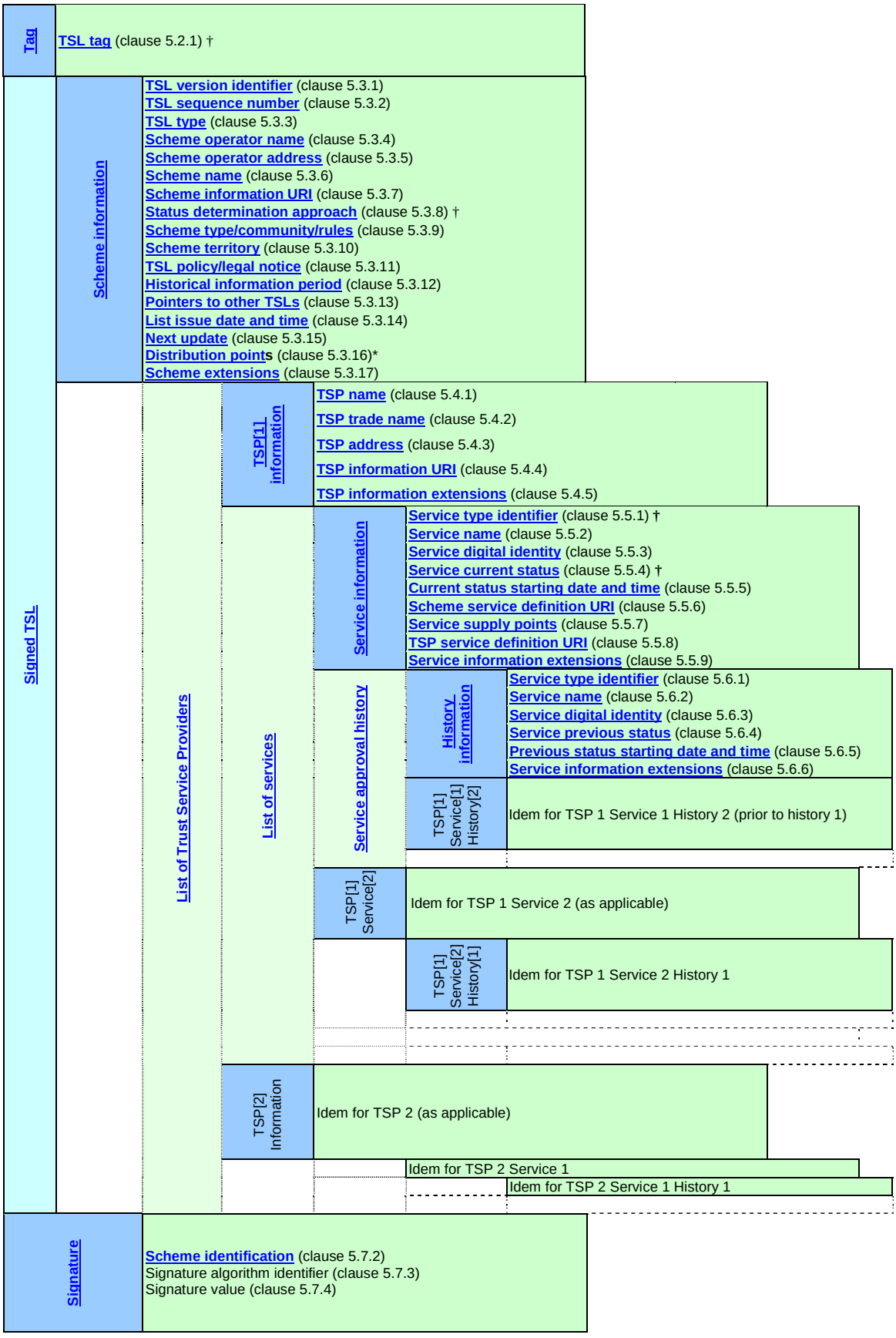
d) For each TSP, a sequence of fields holding information on the service(s) provided by that TSP. This sequence is REQUIRED and MUST have a minimum of one entry. The contents of the service information field are specified in clause 5.5.

e) For each service, a sequence of fields holding information on the status history of that service. This sequence is REQUIRED when the scheme declares that history information is held. The contents of the history information field are specified in clause 5.6.

f) An optional signature computed over all fields of the TSL except the signature value specified in clause 5.7.4. The contents of the signature field are specified in clause 5.7.

5.1.2 Logical model

Figure 1 should be used as a manual index to the TSL field definitions when using a printed copy of the present document.



"*" indicates the field is new in this version of the specification.
"†" indicates that the field's definition has changed since the previous version of the present document.

Figure 1: Logical model of the TSP Status List

5.1.3 Language support

Trust Status Lists MAY be issued supporting multiple (natural) languages. For all fields, where multiple language versions are possible, the following general rules apply:

1) A **multilingual character string** is an ISO 10646 [23] character string encoded in UTF-8. Each **multilingual character string** consists of two parts: a tag, conformant to RFC 5646 [16], that identifies the language in which the string is expressed, and the text in that language. The same content MAY be represented in multiple languages by a sequence of **multilingual character strings**.

2) A **multilingual pointer** is a URI that identifies a resource expressed in a particular language. Each **multilingual pointer** consists of two parts: a tag, conformant to RFC 5646 [16], that identifies the language in which the content pointed-to by the URI is expressed, and the URI expressed as a character string with the syntax specified by RFC 3986 [19], in the given language. The same content MAY be represented in multiple languages by a sequence of **multilingual pointers**.

When the present document is used to implement an EU-wide or beyond scheme, multiple languages SHALL be used: at least the official language(s) of the EUMS of issue and UK English. Exceptions to this requirement are acceptable only if the translation would be inadequate, and in any case only in the following clauses:

- 5.3.4 Scheme operator name;
- 5.3.5.1 Scheme operator postal address;
- 5.3.6 Scheme name;
- 5.4.1 TSP name;
- 5.4.2 TSP trade name;
- 5.5.2 Service name.

For these clauses, however, whenever the native terms cannot be represented using the Latin alphabet, as defined in Unicode, one issue of the term in the native language plus one issue with a transliteration to the Latin alphabet SHALL be used.

Further detail requirements regarding multilingual implementation are given in annex E.

5.1.4 Date-time indication

All fields carrying date-time values SHALL comply with the following rules:

1) the date-time values SHALL be a character string formatted according to ISO 8601 [22];

2) the date-time value SHALL be expressed as "Zulu" (Coordinated Universal Time or UTC): its value MUST contain year (four digits are RECOMMENDED), month, day, hour, minute and second and SHALL NOT include fractional seconds. The time scale MUST be based on the second as defined in ITU-R Recommendation TF.460-5 [25].

3) The actual format to be used is implementation specific.

5.1.5 Use of Uniform Resource Identifiers

In the definitions of TSL fields given in this clause, many use uniform resource identifiers (URIs) to indicate the meaning of the field concerned. Within these definitions a "common name" is used to broadly and simply describe the specific values or meanings of the field. These common names are linked to their declaration in annex D, which formally states all URIs used in the present document, with their meanings.

Many fields allow to use different URIs, which have the same purpose, are registered and described by the scheme operator or another entity and are recognized by the intended user community. Such URIs may be registered with ETSI. Information on URI registration can be found in clause D.3.

5.1.6 Value of Country Code fields

All fields carrying Country Codes values SHALL be in accordance with ISO 3166-1 [21] Alpha 2 codes with the following exceptions:

1) The Country Code for United Kingdom SHALL be "UK".

2) The Country Code for Greece SHALL be "EL".

3) When the scope of the field is the European Union and/or the European Commission the code "EU" SHALL be used.

5.1.7 TSL Format

A TSL can be issued in an human readable or machine processable format. If the scheme operator publishes more than one TSL in different formats, they MUST contain exactly the same information, with the only exception of the time strictly required to the update operation, where the information can be not aligned for a short period of time.

The human readable format MUST be in PDF/A format [45].

The machine processable format MUST be in ASN.1 DER or XML format as specified in annexes A and B respectively.

5.2 Trust-service Status List tag

5.2.1 TSL Tag

Description: This field is REQUIRED. The TSL SHALL be tagged to facilitate its identification during electronic searches and also to confirm its purposes when in human-readable form.

Format: A character string which indicates that the data structure is a TSL. This SHALL be the character representation of the TSLTag URI.

Meaning: A unique value enabling a web-searching tool to establish during a WWW-wide search for TSLs that a resource it has located is indeed a TSL. Only the characters required to fully represent the URI SHALL be present.

5.3 Scheme information

5.3.1 TSL version identifier

Description: This field is REQUIRED. It SHALL specify the version of the TSL format.

Format: Integer.

Meaning: The value of the identifier for TSLs conforming to this version of the present document, which SHALL be "**3**".

NOTE: This field will only be incremented when the rules for parsing the TSL change, e.g. through addition/removal of a field or a change to the values or meaning of an existing field. Revisions to the specification which do not change the parsing rules of the TSL MAY be made without revision to this field - there should be no reliance placed upon the continuing alignment of the TSL version and the specification issue after the initial publication of the present document at version 01.01.01 which defined TSL version "**1**".

5.3.2 TSL sequence number

Description: This field is REQUIRED. It SHALL specify the sequence number of the TSL.

Format: Integer.

Meaning: At the first release of the TSL, the value of the sequence number SHALL be 1. The value SHALL beincrementedat each subsequent release of the TSL and SHALL NOT, under any circumstance, be re-cycled to "1" or to any value lower than the one of the TSL currently in force.

5.3.3 TSL type

Description: This field is REQUIRED. It SHALL specify the type of the TSL.

Format: A TSL type indicator expressed as one of the URIs defined in clause D.2 or another URI having the same purpose, registered and described by the scheme operator or another entity, such as a community or federation of schemes, a standards body, etc.

Meaning: This field SHALL indicate the type of the TSL which will permit a parser to determine which form of any following fields to expect, where those fields have alternative meanings according to the type of TSL represented. The present document specifies in clause D.2 the following TSL Types:

"Generic" when the TSL contains a list of trust services which are approved or recognized by the scheme operator owning the TSL through a process of direct oversight (whether voluntary or regulatory);

"Schemes" when the TSL exclusively contains a list of TSL Issuers which are independently responsible for the approval or recognition by a community of trust services through a process of direct oversight (whether voluntary or regulatory).

5.3.4 Scheme operator name

Description: This field is REQUIRED. It SHALL specify the formal name under which the scheme operator does business or is given its mandate (e.g. for governmental administrative agencies).

Format: A sequence of multilingual character strings (see clause 5.1.3).

Meaning: The name of the scheme operator MUST be the name which is used in formal legal registrations or authorizations and to which any formal communication, whether physical or electronic, should be addressed.

Local language and cross-border (international) trading considerations MAY require that this information be provided both in a national language (and script) and in a commonly accepted internationally-used language.

5.3.5 Scheme operator address

Description: This field is REQUIRED. It SHALL specify the address of the legal identity identified in clause 5.3.3, for both postal and electronic communications. Users (subscribers, relying parties) should use this address as the contact point for enquiries, complaints, etc. to the scheme operator.

This is a multi-part field consisting of the scheme operator physical address specified in clause 5.3.5.1 and the scheme operator electronic address specified in clause 5.3.5.2.

5.3.5.1 Scheme operator postal address

Description: This field is REQUIRED. It SHALL specify the postal address of the legal entity identified in clause 5.3.3, with the provision for the inclusion of the address in multiple languages.

Format: A sequence of multilingual character strings (see clause 5.1.3).

Each sequence of character strings SHALL give the following attributes pertaining to the legal entity:

- street address (sub-components internally delimited by ";");
- locality (town/city);

- optionally, if applicable, State or Province name;
- postal code, if applicable;
- country name as a two-character code in accordance with clause 5.1.6.

Meaning: This MUST be a postal address at which the scheme operator provides a regularly-serviced capability for conventional (physical) mail.

5.3.5.2 Scheme operator electronic address

Description: This field is REQUIRED. It SHALL specify the address of the legal entity identified in clause 5.3.3 for electronic communications.

Format: Sequence of character strings giving: e-mail address as a URI, in the form specified by RFC 3986 [19] and with the URI scheme defined in RFC 2368 [11], and; web-site as a URI, in the form specified by RFC 3986 [19].

At least one such character string MUST be present.

Meaning: In the case of an e-mail address, this MUST be an address at which the scheme operator provides a regularly serviced help line capability. In the case of a web-site URI, this MUST lead to a capability whereby the user MAY communicate with a regularly serviced help line capability.

5.3.6 Scheme name

Description: This field is REQUIRED. It SHALL specify the name under which the scheme operates.

Format: A sequence of multilingual character strings (see clause 5.1.3).

Meaning: The name of the scheme MUST be the name which is used in formal references to the scheme in question, and MUST be unique and MUST NOT be used by any other scheme operated by the same entity.

Local language and cross-border (international) trading considerations MAY require that this information be provided both in a national language (and script) and in a commonly accepted internationally-used language.

NOTE: The scheme name is required to uniquely identify by name the scheme referred to by the "Scheme information URI", and also to ensure that in the event that a scheme operator operates more than one scheme, there is a distinct name given to each of them. Thus if a scheme name is the same as the scheme operator's name that name may only be used for one scheme.

5.3.7 Scheme information URI

Description: This field is REQUIRED. It SHALL specify the URI(s) where users (subscribers, relying parties) can obtain scheme-specific information.

Format: A sequence of multilingual pointers (see clause 5.1.3).

Meaning: The referenced URI(s) MUST provide a path to information describing the general terms and conditions of the scheme, its criteria for TSP and service approval and other generic information which applies to the scheme operations.

NOTE: The URI(s) could differ from the URI(s) provided in clause 5.3.5.2, e.g. if the scheme operator wanted to have a different service or facility for handling e-mails.

5.3.8 Status determination approach

Description: This field is REQUIRED. It SHALL specify the identifier of the status determination approach.

Format: A status determination approach indicator expressed as an URI with one of the following purposes:

- Active;
- Passive;
- Delinquent

for both "Generic" and "Schemes" TSL types, and:

- List for "Schemes" TSL types only.

The field content SHALL be represented by one of the URIs listed in clause D.2, pertaining to this field, or another URI having the same purpose, registered and described by the scheme operator or another entity, such as a community or federation of schemes, a standards body, etc. and which is recognized by the intended user community.

Meaning: The meaning of this field is specified in clause 5.5.4.

For TSLs of type "Schemes", setting the status determination approach to Active, Passive, Delinquent or any URI having the same purpose means, that all Schemes the TSL points to MUST be following the given status determination approach. Otherwise, *List* MUST be used.

5.3.9 Scheme type/community/rules

Description: This field is OPTIONAL. If present, it SHALL contain one or more registered URIs.

Format: A sequence of strings each one compliant with RFC 3986 [19].

Meaning: This field MAY be used by any community of users which establishes and registers a URI by which to denote participation within that community. Such communities MAY be legislative, inter-governmental, industry or other, which have registered a URI for the purposes of identifying themselves. The referenced URI(s) MUST identify the specific policy/rules against which services included in the list SHALL be assessed and from which the type of scheme or community MAY be determined. Where more than one URI is provided each MUST be a complete subset of the policy defined by its predecessor (e.g. a corporate policy might be over-arching; separate divisions MAY have their own implementations which are fully within the corporate high-level policy).

NOTE: By permitting a string of hierarchical URIs the scheme MAY indicate a broad set of rules within which it operates and a specific set of detailed implementation rules. E.g. consider two URIs, the first of which confirms adherence to the supervision requirements relating to Certificates as defined by Directive 1999/93/EC [1], the second of which specifies the particular rules of an individual Member State's scheme. The hierarchy of the URIs is only a logical one: the URIs themselves need not directly represent that structure.

5.3.10 Scheme territory

Description: This field is OPTIONAL. If present, it SHALL specify the country or territory in which the scheme is established.

Format: Character string giving either:

a) a Country name, as a two-character code in accordance with clause 5.1.6;

b) the two-character code "EU" indicating the European Union.

Meaning: A two-letter code which specifies the country or territory in which the scheme is established.

5.3.11 TSL policy/legal notice

Description: This field is OPTIONAL. If present, it SHALL specify the scheme's policy or provide a notice concerning the legal status of the scheme or legal requirements met by the scheme for the jurisdiction in which the scheme is established and/or any constraints and conditions under which the TSL is maintained and offered.

Format: Either:

a) A sequence of multilingual pointers (see clause 5.1.3) for specific use as a pointer to the policy or notice; or

b) the actual text of any such policy or notice, as a multilingual character string (see clause 5.1.3).

Meaning: Any referenced URI MUST provide a path to information describing the policy under which the TSP operates or any relevant legal notices with which users of the TSL should be aware. If plain text is provided, this MUST serve the same purpose.

In either case, local language and cross-border (international) trading considerations MAY require that this information be provided both in a national language and in a commonly accepted internationally-used language.

NOTE: If this field is implemented using format (a) then TAB, CR and LF control characters MAY be used, irrespective of the requirements of annex E.

5.3.12 Historical information period

Description: This field is REQUIRED. It SHALL specify the duration over which historical information in the TSL is provided.

Format: Integer.

Meaning: a) 0 (zero) SHALL signify that the scheme does not retain history information;

b) 1 through 65 534 SHALL signify the number of days over which historical information in the TSL is provided;

c) 65 535 or greater SHALL signify an indefinite duration.

In case "Status determination approach" is of type "List" this field SHALL be set to 0 (zero).

NOTE: The period chosen should take due account of the legal requirements for data retention in the host jurisdiction. A range of values 1 through 65 534 allows for a specific duration of up to at least 179 years, which is considered to be sufficient for most foreseen purposes.

5.3.13 Pointers to other TSLs

Description: This field is OPTIONAL. It MAY be used to indicate other TSLs.

Format: Sequence of one or more tuples, each tuple giving:

a) a string containing the URI of another TSL (in human readable or machine processable format);

b) optional digital identities, representing the issuer of the TSL pointed to, formatted as specified in clause 5.5.3; and

c) optional additional information in a scheme-specific format as a set of TSL Qualifiers.

Meaning: A series of pointers to the location of other TSLs, with additional information whose meaning is scheme-specific as further specified in clause 5.3.13.1. Such TSLs MAY be maintained by other parties or by the operator of the TSL in question. If digital identities are given, they MUST be usable for verifying the signature on the TSL pointed to.

In case "Status determination approach" is of type "List" this field MUST be present and contains the list of all applicable TSLs issued by the schema operators belonging to the community or federation for which the "List" TSL is issued.

In case a TSL belongs to a community or federation and a TSL with "Status determination approach" of type "List" is issued for such a community/federation to list its members, it is RECOMMENDED that the TSL issued by the schema operator contains a pointer to the TSL that lists the community/federation members. It is RECOMMENDED that the mentioned pointer references one or more digital identities to verify the pointed TSL.

If digital identities field is used in a TSL with "Status determination approach" of type "List" for a community/federation, it is responsibility of each schema operator to send to the issuer of this TSL the set of digital identities that can be validly used by TSL users to authenticate the TSL and maintain them updated. Use of more than one digital identity can help the management of the TSL signing process (e.g. in case of expiration/substitution of TSL signing keys or more than a single signing key is allowed to sign the TSL). An URI pointing to the location where the public key or digital certificate currently in force can be added.

If one or more digital identity is present for a given TSL, it MUST be successfully authenticated with one of such digital identities before its use.

NOTE: If an entry does not contain the digital identities field, the TSL user MUST verify the authenticity of the TSL pointed by this field before relying on its content, having received in another secure way the information necessary to reliably perform such a verification.

5.3.13.1 Additional information field

This field is RECOMMENDED if more than one Pointer to other TSL is present. When this field is present, it MUST be unique within the TSL (i.e. no couple of Additional information field attributes can containin exactly the same set of TSL Qualifiers). This field contains important additional information about the scope of the TSL pointed to, which may be used as hints by applications using TSLs.

A typical scenario could be a federation of schema operators, each one of them with a territorial scope or issuing more than one TSL for different scopes (e.g. voluntary accreditation schemas, REM, ID certificates, other CSP services). To avoid the need for an application to download every pointed TSL to find the one that fits the current need, some field of the TSL pointed to can be replicated as TSL Qualifiers in the "Pointers to other TSLs" field.

Possible values for TSL Qualifiers, without any limitation to define new schema specific ones by schema operators, are:

a) TSL type, as defined in clasue 5.3.3.

b) Scheme operator, as defined in clause 5.3.4.

c) Scheme name, as defined in clause 5.3.6.

d) Scheme information URI, as defined in clasue 5.3.7.

e) Scheme type/community/rules, as defined in clause 5.3.9.

f) Scheme territory, as defined in clause 5.3.10.

g) Mime type, as one of the media types defined in clause 6.1.1.2.1

5.3.14 List issue date and time

Description: This field is REQUIRED. It SHALL specify the date and time on which the list was issued.

Format: Date-time value (see clause 5.1.4).

Meaning: Coordinated Universal Time (UTC) at which the TSL was issued.

5.3.15 Next update

Description: This field is REQUIRED. It SHALL specify the latest date and time by which the next planned update of the TSL will be made available by the schema operator or be null to indicate a closed TSL. Applications should consider, in the event they implement some caching mechanism, that other TSLs could be issued and published before the next planned update.

Format: Date-time value (see clause 5.1.4).

Meaning: Coordinated Universal Time (UTC) by which an updated TSL SHALL be issued, expressed as Zulu. The schema operator MAY issue other TSL before the next planned TSL and Next Update MUST always be the same or greater than the next update of the previous TSL. If a scheme ceases operations or halts publication of its TSL a final version SHALL be published with all services' status shown as "expired" (see Service current status) and this field set null.

In the event of no interim status changes to any TSP or service covered by the scheme, the TSL MUST be re-issued by the time of expiration of the last TSL issued. TSL with a Next update occurring in the past MUST be discarded as expired as a measure to reduce the risk of a substitution by an attacker with an old TSL.

5.3.16 Distribution points

Description: This field is OPTIONAL. If used, it SHALL specify locations where the current TSL is published and where updates to the current TSL can be found. If multiple distribution points are specified, they all must provide identical copies of the current TSL or its updated versions.

Format: A nonempty sequence of strings, each of them compliant with RFC 3986 [19].

Meaning: Dereferencing the given URI will always deliver the latest update of this TSL.

5.3.17 Scheme extensions

Description: This field is OPTIONAL. It MAY be used by scheme operators (or communities thereof) to provide specific service-related information and enhancements to the present document that do not require a change in the version number, which MAY be interpreted by all accessing parties according to the specific scheme's rules.

Format: Sequence of scheme extensions, each of which MUST be selected by the scheme operator according to the meaning and information it wishes to convey within its TSL. Each extension MUST have an indication of its criticality.

Meaning: The meaning of each extension is defined by its source specification, that specification being either the scheme operator's own definition or any other extension definition produced by another entity, such as a community or federation of schemes, a standards body, etc. The criticality indication will have the same semantics as with extensions in X.509-certificates [43]. A system using TSLs MUST reject the TSL if it encounters a critical extension it does not recognize, while a non-critical extension MAY be ignored if it is not recognized.

5.3.18 List of Trust Service Providers

Description: This field is OPTIONAL. In the case where no TSPs are or were recognized by the scheme (according to the scheme type and criteria), this field SHALL be absent. If one or more Trust services are or were recognized by the scheme then the field SHALL contain a sequence identifying each TSP providing one or more of those services, with details on the approval status and (where provided - see clause 5.3.12) history of each of the TSP's services. In case "Status determination approach" is of type "List" this field SHALL NOT be present.

Format: Sequence of TSP information (see clause 5.4).

Meaning: The presence or absence of TSPs within this list can only have meaning when taken in the context of the scheme's status determination approach (see clause 5.3.8). E.g. absence of any listed TSPs under a scheme working solely on a delinquent list principle suggests that there are no known TSPs which are also known to be not operating within the permissible or acknowledged bounds, whereas a similar absence of TSPs in a positive approval-list driven scheme would suggest that no TSPs are approved by the scheme.

NOTE: The term "TSP" is used liberally in the above text, since service providers whose services are listed under a "delinquency" scheme MAY not be deserving of the term "trusted" in the context of the scheme's rules.

5.4 TSP information

5.4.1 TSP name

Description: This field is REQUIRED. It SHALL specify the name of the legal entity responsible for the TSP's services that are or were recognized by the scheme.

Format: A sequence of multilingual character strings (see clause 5.1.3).

Meaning: The name of the legal entity responsible for the TSP MUST be the name which is used in formal legal registrations and to which any formal communication, whether physical or electronic, should be addressed.

NOTE: Local language and cross-border (international) trading considerations MAY require that this information be provided both in a national language (and script) and in a commonly accepted internationally-used language.

5.4.2 TSP trade name

Description: This field is OPTIONAL. If present, it SHALL specify an alternative name under which the TSP identifies itself in the provision of its services.

Format: A sequence of multilingual character strings (see clause 5.1.3).

Meaning: Any name under which the legal entity responsible for the TSP operates, in the specific context of the delivery of those of its services which are to be found in this TSL.

NOTE: Local language and cross-border (international) trading considerations MAY require that this information be provided both in a national language (and script) and in a commonly accepted internationally-used language.

5.4.3 TSP address

Description: This field is REQUIRED. It SHALL specify the address of the legal entity identified in clause 5.4.1, for both physical and electronic communications. Users (subscribers, relying parties) should use this address as the single contact point for enquiries, complaints, etc. to the TSP.

This is a multi-part field consisting of the TSP physical address specified in clause 5.4.3.1 and the TSP electronic address specified in clause 5.4.3.2.

5.4.3.1 TSP postal address

Description: This field is REQUIRED. It SHALL specify the postal address of the legal entity identified in clause 5.4.1, with the provision for the inclusion of the address in multiple languages.

Format: The format SHALL be the same as that specified in clause 5.3.5.1.

Meaning: This MUST be a postal address at which the TSP provides a regularly serviced capability for conventional (physical) mail.

5.4.3.2 TSP electronic address

Description: This field is REQUIRED. It SHALL specify the address of the legal entity identified in clause 5.4.1, to be used for electronic communications.

Format: The format SHALL be the same as that specified in clause 5.3.5.2.

Meaning: In the case of an e-mail address, this MUST be an address at which the TSP provides a regularly serviced customer care or help line capability. In the case of a web-site URI, this MUST lead to a capability whereby the user MAY communicate with a regularly serviced customer care or help line capability.

5.4.4 TSP information URI

Description: This field is REQUIRED. It SHALL specify the URI(s) where users (subscribers, relying parties) can obtain TSP-specific information.

Format: Multilingual pointer (see clause 5.1.3).

Meaning: The referenced URI(s) MUST provide a path to information describing the general terms and conditions of the TSP, legal issues, its customer care policies and other generic information which applies to all of its services.

NOTE: The URI(s) could differ from the URI provided in clause 5.4.3.2, e.g. if the scheme operator wanted to have a different service or facility for handling e-mails.

5.4.5 TSP information extensions

Description: This field is OPTIONAL. It MAY be used by scheme operators to provide specific TSP-related information, to be interpreted according to the specific scheme's rules.

Format: Sequence of TSP extensions, each of which MUST be selected by the scheme operator according to the meaning and information it wishes to convey within its TSL. Each extension MUST have an indication of its criticality.

Meaning: The meaning of each extension is defined by its source specification, that specification being either the scheme operator's own definition or any other extension definition produced by another entity, such as a community or federation of schemes, a standards body, etc. The criticality indication will have the same semantics as with extensions in X.509-certificates ITU-T Recommendation X.509 [43]. A system using TSLs MUST reject the TSL if it encounters a critical extension it does not recognize, while a non-critical extension MAY be ignored if it is not recognized.

5.4.6 List of services

Description: This field is REQUIRED. It SHALL contain a sequence identifying each of the TSP's recognized services and the approval status of that service. At least one service MUST be listed, even if the information held is entirely historical.

Format: Sequence of service information (see clause 5.5).

Meaning: The presence or absence of services within this list can only have meaning when taken in the context of the scheme's status determination approach (see clause 5.3.8). E.g. no services under a scheme working solely on a delinquency list principle suggests that there are ***no*** known services which are ***not*** operating within the permissible or acknowledged bounds, whereas a similar absence of services in a positive approval list driven scheme would suggest that no services meet the scheme's criteria.

If a scheme retains historical information then that information MUST be retained even if the service's present status would not normally require it to be listed (e.g. in a positive list, the service is withdrawn; in a delinquency list, the service conforms to the required standards). Thus a TSP MUST be included even when its only listed service is in such a state, so as to preserve the history. However, if the scheme does not retain historical information then in such a situation, again as the only service related to the TSP in question, when that service needs no longer to be listed then the TSP MUST be removed as well.

5.5 Service information

5.5.1 Service type identifier

Description: This field is REQUIRED. It SHALL specify the identifier of the service type, according to the type of TSL being presented.

Format: An identifier expressed as an URI specifying one of the following service types:

- CA (PKC);
- CA (QC);
- Time-stamp Authority;
- Certificate status (OCSP);
- Certificate status (CRL);
- RA;
- Id verification;
- Certificate generation;
- Attribute CA;
- Archive;
- Key escrow;
- Pin/password credential authority;
- Registered Electronic Mail;
- Signature Policy Authority;
- unspecified.

for TSL type "Generic", and:

- supervisory systems;
- voluntary approval scheme;
- TSL Issuer
- unspecified.

for TSL type "Schemes". For any other TSL type:

- unspecified.

The field content SHALL be represented by one of the URIs listed in clause D.2, pertaining to this field, or another URI having the same purpose, registered and described by the scheme operator or another entity, such as a community or federation of schemes, a standards body, etc. and which is recognized by the intended user community.

Meaning: This field identifies the type of a service. In case of "unspecified" Service Type identifier, it is RECOMMENDED that this information is provided in other ways such a service level Extension.

Additional URIs may be added related to entirely new services types. To assure interoperability among different schemas, a new URI SHALL NOT be added simply to identify local "higher quality services", because this distinction would hinder interoperability. To meet such local requirements different solutions MUST be used. For example, this could be specified in a "Service information extensions" (see clause 5.5.9).

5.5.2 Service name

Description: This field is REQUIRED. It SHALL specify the name under which the TSP provides the service identified in clause 5.5.1.

Format: A sequence of multilingual character strings (see clause 5.1.3).

Meaning: The name under which the TSP provides the service.

Local language and cross-border (international) trading considerations MAY require that this information be provided both in a mother language (and script) and in a commonly accepted internationally-used natural language.

5.5.3 Service digital identity

Description: This field is REQUIRED. It SHALL be either null or SHALL specify at least one representation of a digital identifier unique to the service specified in clause 5.5.1 by which the service can be unambiguously identified. The digital identifier MAY be present more than once and in different formats. If the digital identifier is present more than once, all variants MUST refer to the same identity.

Format: Character string or bit string or data structure specifying for each occurrence of the digital identifier the type of format and the data representing the digital identity. When using public-key technology (i.e. PKI), this field MUST be a representation of the public key(s) the TSP uses for providing its services; e.g. the key used for signing certificates or OCSP responses. Implementation dependent - see annexes A and B.

Meaning: The digital identifier can be of different types depending on the service. It could be a Distinguished Name (DN), a certificate which can be used to verify electronic signatures of the service provider, a public-key or a subject key identifier. If the field is null the scheme responsible for publishing the specific TSL SHALL determine and publish the meaning and significance of a null value.

NOTE: It is RECOMMENDED that, in order to avoid unnecessary processing overhead of parsing a public key certificate, where a DN is available it is stated before any other forms of service digital-identity (e.g. before a public key certificate, which would require parsing to extract include the DN).

5.5.4 Service current status

Description: This field is REQUIRED. It SHALL specify the identifier of the status of the service.

Format: An identifier expressed as an URI specifying one of the following TSP statuses:

- in accordance;
- expired;
- suspended;
- revoked;

- not in accordance

The field content SHALL be represented by one of the URIs listed in clause D.2, pertaining to this field, or another URI having the same purpose, registered and described by the scheme operator or another entity, such as a community or federation of schemes, a standards body, etc. and which is recognized by the intended user community.

Meaning: This is the fundamental aspect of the TSL - i.e. the service's status. That status,

- when having one of the five distinct values as specified above, needs to be interpreted with regard to the scheme's status determination approach (see clause 5.3.8) which indicates the general types of criteria being applied.

  Table 1 is intended to assist in that understanding. The meanings given apply to a status given in either the current or historical part of the TSL, for a scheme which is known still to be operational;

- when having one value of a set of other values as specified in the above "Meaning" paragraph, needs to be interpreted with regard to:

  i) the scheme's status determination approach (see clause 5.3.8) which indicates the general types of criteria being applied; and

  ii) the information having the same purpose as the one provided in table 1;

in order to assist that understanding. Such information can be provided through the use of the "Scheme type/community/rules", see clause 5.3.9, or through the use of a profile of the present specifications, or through any appropriate means."

Should the scheme no longer be operational (which MAY be determined by all the current statuses indicating "expired", or any other status value that implies the service being no longer operational, or implied by the "next update" time having been exceeded or set to null) only the historic information should be relied upon. This is because either the status will have been set to "expired" when the scheme ceased operations and hence no subsequent status information will have been maintained, or the scheme ceased operations before it could affect a re-issue of the TSL in which case it could be uncertain the extent to which the indicated current status remained valid after the publication of the list.

In table 1, grey shading indicates an unlikely combination of approach vs. status, black indicates such a combination is not possible.

Table 1: Meaning of Service status in relation to the Status determination approach

		Status determination approach		
		positive assessment (active approval)	**nomination/observation (passive approval)**	**delinquent**
Service current or previous status	**in accordance**	An assessment has been performed on behalf of the scheme operator and the TSP and its service found to be in compliance.	The service is known to be operational and has not been found to be non-compliant with the scheme's criteria.	This combination cannot exist (since only those non-compliant with the scheme's criteria are listed).
	expired, not renewed	The validity of the assessment has lapsed without the service being re-assessed.	The service is understood to have ceased operations.	This combination cannot exist (since only those TSPs and services non-compliant with the scheme's criteria are listed).
	suspended	No specific conclusion should be drawn - it could be because the service's validity is being verified (for reasons which are likely to be specific to the scheme) or there could be a delay in renewal.	Although no explicit approval is granted under these schemes, such a status could be used if a scheme's possible non-compliance was under investigation.	This combination unlikely to exist (since only those which are non-compliant are listed), although a scheme could, at its own discretion, use such a status if it was investigating a scheme's possible flagging as "non-compliant".
	revoked	Having once been found to be in conformance with the scheme's criteria, the TSP and/or the service have failed to continue to fulfil the criteria set by the scheme.	Essentially as per "not in accordance" (below), except that this combination is unlikely to exist since a scheme applying passive observation is not generally likely to have granted any right or recognition to explicitly revoke, and would there apply the status "not in accordance".	This combination cannot exist, since no positive recognition is granted, hence it cannot be withdrawn (revoked).
	not in accordance	Essentially as per "revoked" (above), except that this combination is unlikely to exist since a scheme exercising positive assessment is more likely to want to remove a positive assertion in the TSP or scheme when there has been a failure to continue to fulfil the criteria set by the scheme, and would therefore apply the status "revoked'.	The TSP and/or the service have been found to be non-compliant with the criteria required by the scheme.	The TSP and/or the service have been found to be non-compliant with the criteria required by the scheme for the TSPs/services listed.

It should be understood that few schemes could state with absolute certitude that all services which potentially fall within their scope are actually listed within the TSL, irrespective e of their status determination approach.

5.5.5 Current status starting date and time

Description: This field is REQUIRED. It SHALL specify the date and time on which the current approval status is effective.

Format: Date-time value (see clause 5.1.4).

Meaning: Coordinated Universal Time (UTC) at which the current approval status became effective.

NOTE: The user (subscribers, relying parties) could apply this information by comparing it with other available information, e.g. the date and time on which a certificate or a time stamp was issued. From the comparison, the user could determine whether the specific service of the TSP had the desired approval status under the scheme at the date and time of provision of the service.

5.5.6 Scheme service definition URI

Description: This field is OPTIONAL. If present, it SHALL specify the URI(s) where users (subscribers, relying parties) can obtain service-specific information provided by the scheme operator.

Format: A sequence of multilingual pointers (see clause 5.1.3).

Meaning: The referenced URI(s) MUST provide a path to information describing the service as specified by the scheme.

5.5.7 Service supply points

Description: This field is OPTIONAL. If present, it SHALL specify one or more URIs where users (subscribers, relying parties) can access the service.

Format: A sequence of character strings whose syntax MUST be compliant with RFC 3986 [19].

Meaning: The referenced URI(s) MUST specify where and how the service can be accessed.

5.5.8 TSP service definition URI

Description: This field is OPTIONAL. If present, it SHALL specify the URI(s) where users (subscribers, relying parties) can obtain service-specific information provided by the TSP.

Format: A sequence of multilingual pointers (see clause 5.1.3).

Meaning: The referenced URI(s) MUST provide a path to information describing the service as specified by the TSP.

5.5.9 Service information extensions

Description: This field is OPTIONAL. It MAY be used by scheme operators (or communities thereof) to provide specific service-related information and enhancements to the present document that do not require a change in the version number, to be interpreted by all accessing parties according to the specific scheme's rules.

Format: Sequence of service information extensions, each of which MUST be selected by the scheme operator according to the meaning and information it wishes to convey within its TSL. Each extension MUST have an indication of its criticality.

Meaning: The meaning of each extension is defined by its source specification, that specification being either the scheme operator's own definition or any other extension definition produced by another entity, such as a community or federation of schemes, a standards body, etc. The criticality indication will have the same semantics as with extensions in X.509-certificates ITU-T Recommendation X.509 [43]. A system using TSLs MUST reject the TSL if it encounters a critical extension it does not recognize; while a non-critical extension MAY be ignored if it is not recognized.

5.5.10 Service approval history

Description: This field is OPTIONAL but MUST be present if Historical information period is non-zero (i.e. the scheme retains or intends to retain historical information at all). In the case where historical information is intended to be retained but the service has no history prior to the current status (i.e. a first recorded status or history information not retained by the scheme operator) this field SHALL be empty. Otherwise, for each change in TSP service approval status which occurred within in the historical information period as specified in clause 5.3.12, information on the now previous approval status SHALL be provided in descending order of status change date and time (i.e. the date and time on which the subsequent approval status became effective).

Format: Sequence of History information (see clause 5.6).

Meaning: When present, a sequence of all previous status entries which the scheme has recorded for the given TSP and service, within the period over which historical information is retained.

5.6 Service approval history information

5.6.1 Service type identifier

Description: This field is REQUIRED. It SHALL specify the identifier of the service type, with the Format and Meaning used in clause 5.5.1.

5.6.2 Service name

Description: This field is REQUIRED. It SHALL specify the name under which the TSP provided the service identified in clause 5.5.1, with the Format and Meaning used in clause 5.5.2.

NOTE: This clause does not require that the name be the same as that specified in clause 5.5.2. A change of name MAY be one of the circumstances requiring a new status.

5.6.3 Service digital identity

Description: This field is REQUIRED. It SHALL specify at least one representation of a digital identifier unique to the service specified in clause 5.5.1, with the Format and Meaning used in clause 5.5.3.

5.6.4 Service previous status

Description: This field is REQUIRED. It SHALL specify the identifier of the previous status of the service, with the Format and Meaning used in clause 5.5.4.

5.6.5 Previous status starting date and time

Description: This field is REQUIRED. It SHALL specify the date and time on which the previous status in question became effective, with the Format and Meaning used in clause 5.5.5.

5.6.6 Service information extensions

Description: This field is OPTIONAL. It MAY be used by scheme operators to provide specific service-related information, to be interpreted according to the specific scheme's rules, with the Format and Meaning used in clause 5.5.9.

5.7 Signature

5.7.1 Signed TSL

It is RECCOMENDED that the TSL is signed by the scheme operator to ensure its authenticity and integrity. No signature liability is implied other than what provided by regulations/agreements between scheme operators and/or other entities, such as a community or federation of schemes. This clause does not prescribe the format of the signature but refers to normative annexes A and B for implementations using ASN.1 and XML respectively and to PAdES part 3 (TS 102 778-3 [46]) in case for the PDF/A implementation, and additional informative guidance given in annex F.

This clause does not mandate a specific format but specifies the content of some fields that MUST be present in the signature, so they reperesent a requirement on the specific format chosen. For the signature implementations described in the present document it is specified what are the signature fields that map to the fields defined in this clause. In case a different implementation is chosen by the schema operator, the implementation MUST support this requirement. In order to accommodate implementation dependent issues, they need not necessarily appear in any specific order. The present document REQUIRES that scheme operators acquire and use to sign their TSL a public-key cryptography signing key which is bound into a public-key certificate conformant with ITU-T Recommendation X.509 [43]. It is RECOMMENDED that all the algorithms related to the signature are at least as strong as the ones used for the services digital identities listed in the TSL.

In case the TSL is not signed, its authenticity and integrity MUST be guaranteed by an appropriate communication channel with an equivalent security level. Use of TLS [44] is RECOMMENDED for this purpose and the fingerprint of the certificate of the TLS channel MUST be made available out of band to the TSL users by the schema operator.

5.7.2 Scheme identification

Description: This field is REQUIRED. It SHALL specify a reference assigned by the scheme operator which uniquely identifies the specific scheme and this TSL, and MUST be included in the calculation of the signature.

Format: Character string or Bit string is suggested, depending on the implementation.

Meaning: MUST represent one of the following:

- an X.509-certificate conformant to ITU-T Recommendation X.509 [43];
- a value of a SubjectKeyIdentifier extension conformant to ITU-T Recommendation X.509 [43];
- an implementation-specific X.509-certificate identifier;
- a public key.

The actual choice is implementation dependent and will depend on constraints imposed by the signature implementation framework.

NOTE: If the scheme operator operates more than one scheme for which it publishes a TSL they should use a unique reference in this field for **each** TSL they publish.

5.7.3 Signature algorithm identifier

Description: This field is REQUIRED. It SHALL specify the cryptographic algorithm that has been used to create the signature and MUST be included in the calculation of the signature.

Format: Character string or Bit string is suggested, depending on the implementation.

Meaning: Depending on the algorithm used, this field MAY require additional parameters. This field MUST be included in the calculation of the signature.

5.7.4 Signature value

Description: This field is REQUIRED. It SHALL contain the actual value of the digital signature. Since the signature protects the signed information from undetected manipulation, all fields of the TSL except the signature value itself MUST be included in the calculation of the signature. The calculation of the digital signature SHALL cover all fields described in clauses 5.2 to 5.6 as well as clauses 5.7.2 and 5.7.3.

Format: Character string or Bit string is suggested, depending on the implementation .

Meaning: Contains the actual value of the digital signature.

5.8 TSL extensions

This clause defines general use TSL extensions. The extension definition MUST specify if the extension can be used at Scheme, TSP and/or Service level.

Scheme level extensions MUST only be used in the field defined in clause 5.3.17.

TSP level extensions MUST only be used in the field defined in clause 5.4.5.

Service level extensions MUST only be used in the field defined in clause 5.5.9.

5.8.1 expiredCertsRevocationInfo Extension

Description: This extension is optional and, if present, MUST be used at Service level only and can be applied only to the following service types (as defined in clause 5.5.1):

- CA (PKC);
- CA (QC);
- Certificate status (OCSP);
- Certificate status (CRL);

This extension MUST NOT be set critical.

Format: Date-time value (see clause 5.1.4).

Meaning: This TSL extension indicates the time from which on the service issues CRL and/or OCSP responses that keep revocation notices for revoked certificates also after they have expired. This extension addresses the same issue addressed in ISO/IEC 9594-8:2005 [26], clause 8.5.2.12. expiredCertsRevocationInfo is expressed as GeneralizedTime and indicates that the scope of each CRL and OCSP response, issued by the service to which this extension applies, is extended to include the revocation status of certificates that expired at the exact time specified in the extension or after that time. If limitations in the CRL's scope are specified (by either reason codes or by distribution points), that applies to expired certificates as well. The revocation status of a certificate SHALL NOT be updated once the certificate has expired. This behaviour is openly allowed by ISO/IEC 9594-8:2005 [26] and RFC 5280 [17].

If a CRL contains the extension expiredCertsOnCRL defined in [26] it prevails over the TSL extension value but only for that specific CRL.

5.8.2 additionalServiceInformation Extension

Description: This extension is optional and, if present, MUST be used at Service level only. It is used to provide additional information on a service. Examples are: qualified timestamps, blue certificates or fat-free ocsp-responses.

Format: A tuple providing the information detailed below. If required, a TSL MAY have more than one AdditionalServiceInformation extension in the same service entry, each extension giving:

a) an URI identifying the additional information. Possible values, not limited to the following list:

- an URI indicating some nationally defined specific qualification for a supervised/accredited Trust Service Token provisioning service, e.g.:
  - a specific security/quality granularity level with regard to national supervision/accreditation scheme for CSPs not issuing QCs (e.g. RGS */**/*** in France, specific "supervision" status set by national legislation for specific CSPs issuing QCs in Germany), see note(4) of "Service current status" - clause 5.5.4;
  - or a specific legal status for a supervised/accredited Trust Service Token provisioning (e.g. nationally defined "qualified TST" as in Germany, Hungary or Italy);
  - or meaning of a specific Policy identifier present in a X.509v3 certificate provided in "Sdi" field;
- a registered URI as specified in "Service type identifier", clause 5.5.1, in order to further specify the participation of the "Sti" identified service as being a component service of a certification service provider issuing QC (e.g OCSP-QC, CRL-QC, and RootCA-QC);

b) an optional string containing the serviceInformation classification, meaning as specified in the scheme (e.g. in France services are classified with *, ** or ***);

c) any optional additional information provided in a scheme-specific format.

Meaning: This TSL extension can be used to provide additional information about a certain service that may help to verify the applicability of the given service for a certain purpose. E.g., in France time stamping services are categorized in three groups: *, ** and ***. The service type identifier alone does not (and should not) allow differentiating between these three quality levels. This extension allows doing so in a very generic way.

Dereferencing the URI SHOULD lead to human readable documents containing all the details required to understand the extension, and in particular explaining the meaning of the given URIs, specifying the possible values for serviceInformation and the meaning for each value.

6 Operations

6.1 TSL publication

Scheme operators will likely make TSLs available to TSL-users by publishing them in a Directory or a Web server. Lightweight Directory Access Protocol (LDAP) defined in RFC 4511 [7], Hypertext Transfer Protocol (HTTP) defined in RFC 2616 [12] and the File Transfer Protocol (FTP) defined in RFC 959 [3] offer methods for certificate and TSL distribution. The transport protocols specified below allow end entities to access TSLs. Repository providers MUST support at least HTTP transport. They may also support LDAP and FTP. An application processing TSLs MUST support at least HTTP transport and may support LDAP and FTP.

To avoid possible attacks, the use of secure channels, like TLS [44], is RECOMMENDED. Otherwise, there is no requirement for specific security mechanisms to be applied at this level, if the TSLs is signed. An application processing TSLs MUST support at least TLS over HTTP (HTTPS).

Any file containing a machine readable TSL must either only contain a DER-encoded ASN.1 representation or an XML representation of the TSL as specified in the present document. There MUST be no extraneous header or trailer information in the file.

The same provisions apply to human readable TSLs published in PDF/A format.

6.1.1 Transport Protocols

6.1.1.1 LDAP transport

This text following in this clause refers explicitly to LDAP v3.

6.1.1.1.1 Attributes and Object class definition

In order to use an LDAP-server-like repository to publish the TSLs in compliance with the present document, these servers MUST be compliant with LDAP version 3: therefore they MUST support the syntax notation defined by RFC 4517 [8] and they must be also compliant with RFC 4511 [7] and RFC 4519 [10].

1) cn: this attribute MUST be present and the value MUST be the Relative Distinguished Name (RDN) of the entry, in form of Common Name; this attribute is defined by RFC 4519 [10]. It is RECOMMENDED to use the Scheme name field of the TSL as the value or as part of the value for the CN. This helps to search the directory for TSLs more efficiently.

2) tslTag: this attribute MUST be present and the value MUST be the OID 0.4.0.2231.1.1; in order to speed-up the search operations, the indexing of this attribute is RECOMMENDED; the attribute is defined according to the RFC 4517 [8] syntax as:

```
(  0.4.0.2231.5.2
NAME 'tslTag'
DESC 'Indexed. Indicates that the entry contains a TSL (the value of the OID is
0.4.0.2231.1.1)'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
EQUALITY objectIdentifierMatch
SINGLE-VALUE
)
```

3) tslDer: this attribute MAY be present; in this case the value must be the sequence of bytes that represents the DER-encoded TSL; the attribute is defined according to the RFC 4517 [8] syntax as:

```
(  0.4.0.2231.5.3
NAME 'tslDer'
DESC 'DER-encoded TSL'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
EQUALITY octetStringMatch
SINGLE-VALUE
)
```

4) tslXml: this attribute MAY be present; in this case the value must be the sequence of bytes that represents the XML-encoded TSL; the attribute is defined according to the RFC 4517 [8] syntax as:

```
(  0.4.0.2231.5.4
NAME 'tslXml'
DESC ' XML-encoded TSL'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
EQUALITY octetStringMatch
SINGLE-VALUE
)
```

5) tslPdf: this attribute MAY be present; in this case the value must be the sequence of bytes that represents the TSL in PDF/A human readable format; the attribute is defined according to the RFC 4517 [8] syntax as:

```
(  0.4.0.2231.5.5
NAME 'tslPdf'
DESC ' PDF/A-encoded TSL'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.40
EQUALITY octetStringMatch
SINGLE-VALUE
)
```

At least one of the optional attributes tslDer, tslXml and tslPdf MUST contain a value.

A TSL published on an LDAP server MUST be stored within a dedicated entry. The structural Object Class of such an entry MUST be tslDistributionPoint and MUST use the attributes previously defined. This Object Class is defined according to RFC 4517 [8] syntax as:

```
(0.4.0.2231.5.1
NAME 'tslDistributionPoint'
DESC 'OC containing the TSL'
STRUCTURAL
SUP top
MUST ( cn $ tslTag )
MAY  ( tslDer $ tslXml $ tslPdf)
)
```

Each TSL is stored within a specific entry of the LDAP server and this entry MAY be located in any point of the Directory Information Tree (DIT). Multiple TSLs can be stored within the DIT. In this case, each TSL MUST be stored in a different entry so as to be uniquely identified by the Distinguished Name (DN) of the entry that contains it.

For each TSL it is possible to store both the DER-encoded and the XML-encoded TSL, but at least one of the two formats MUST be present (i.e. the corresponding attribute MUST have a value). If both formats are published, they MUST be stored in the same entry. Each entry constitutes a TSL Distribution Point (TDP).

Within the DIT, the tslDistributionPoint SHOULD be hierarchically located under an entry whose class is one of the following:

- domain
- locality
- organization
- organizationalUnit
- organizationalPerson
- organizationalRole
- applicationProcess

6.1.1.2 HTTP-Transport

This clause specifies a means for transport of TSLs via the Internet using HTTP.

6.1.1.2.1 HTTP-Media Type

TSL payloads MUST be sent using one of the following media types, depending on the format of the TSL (PDF, ASN.1 or XML):

- application/pdf
- application/vnd.etsi.tsl.der
- application/vnd.etsi.tsl.xml

The client MAY, when sending requests, provide an HTTP Accept header field. This header field SHOULD indicate an ability to accept, as a minimum "*application/pdf*", "*application/vnd.etsi.tsl.der*" OR "*application/vnd.etsi.tsl.xml*".

6.1.1.3 FTP-Transport

TSL-repository-providers may also offer FTP as a way to access TSLs similar to the HTTP transport. Since FTP does not support media types, as does HTTP, it is RECOMMENDED that the file extension defined in clause 6.1.1.5 be used, to enable media type recognition by filename.

6.1.1.4 Email Transport

This clause specifies the message format required for transport of TSLs via Internet mail. A scheme or another service provider may want to "push" automatically newly-published TSLs to its users, using email as the transport mechanism.

The email containing the TSL payloads MUST be compliant to RFC 5322 [14] and the RFC 2045 [4] Message.

6.1.1.4.1 Content-Types

TSL payloads MUST be sent with one of the following two content types, depending on the representation of the TSL (ASN.1 or XML):

- application/vnd.etsi.tsl.der
- application/vnd.etsi.tsl.xml

6.1.1.4.2 Encoding considerations

For the DER version it is RECOMMENDED to use base64-transfer encoding. For the XML version, the encoding considerations of clause 3.2 of RFC 3023 [15] as well as clause 6.1.1.5 of the present document are applicable.

6.1.1.5 MIME registrations

Two MIME-Types and file-extensions support the transfer of TSLs:

NOTE: At the time of publication the MIME-Types are undergoing registration procedure with IANA and users are advised to make their own checks for completion of these formalities (the list of Directories of Content Types and Subtypes can be found here: http://www.iana.org/assignments/media-types/application/).

MIME media type name:	Application
MIME subtype name:	vnd.etsi.tsl.der
Required parameters:	none
encoding considerations:	binary
File extension:	dtsl or dts

MIME media type name:	Application
MIME subtype name:	vnd.etsi.tsl+xml
Required parameters:	none
encoding considerations:	binary
File extension:	xtsl or xts
Security considerations:	TSLs do not contain any active code or invoke any automated processing by itself. It is expected that clients only parse the TSL and that there is no security risk. TSLs are signed; no additional integrity protection is required. TSLs typically are meant to be public, no confidentiality is required.
Published specification:	The TSL format as defined in the present document.

6.2 TSL Signer Certificate

It is RECOMMENDED that the TSL Signer Certificates, in order to improve interoperability and to simplify the verification software, is conformant to the following restrictions:

- "Country code" and "Organization" fields in Subject Distinguished Name match respectively the "Scheme Territory" and one of the "Scheme operator name". Fot the latter, it is RECOMMENDED to use as available the value in English language (preferred) or local language (transliterated to Latin script)
- KeyUsage extension set only and exclusively digitalSignature or nonRepudiation (contentCommitment).
- ExtendedKeyUsage extension present containing exclusively id-tsl-kp-tslSigning (see below)
- SubjectKeyIdentifier extension present (as per RFC5280, one of the first 2 methods specified in clause 4.2.1.2)
- BasicConstraints extension CA=false

In order to indicate thatthe use of key-pairs is restricted to sign TSLs only, an X.509 v3 certificate can include the following key purpose id OID in the extended key usage extension:

```
-- OID for TSL signing KeyPurposeID for ExtKeyUsageSyntax

   id-tsl OBJECT IDENTIFIER { itu-t(0) identified-organization(4)
                    etsi(0) tsl-specification (2231) }
   id-tsl-kp OBJECT IDENTIFIER ::= { id-tsl kp(3) }
   id-tsl-kp-tslSigning OBJECT IDENTIFIER ::= { id-tsl-kp tsl-signing(0) }
```

6.3 TSL Distribution Points

Trust Service Providers may wish to give information on how to locate a TSL of the scheme they operate under. To do so, they MAY include the following extension in their trust service tokens (certificates, CRLs, time stamp tokens, OCSP responses and other). If the extension mechanism allows for the expression of criticality, this extension MUST NOT be marked critical. A distribution point must remain accessible until all certificates it is referenced in have expired. The value of this extension will be a sequence of URIs and is specified in clause A.7.

Annex A (normative): Implementation in ASN.1

A.1 Structure of the Trust-service Status List

A.1.1 ASN.1 versioning

This clause specifies the ASN.1 structures to be used when implementing an ASN.1-version of the present document. The field names used reflect those assigned to fields in clause 5.

The ASN.1 syntax used in this annex is the 1988 version, as defined by ITU-T Recommendation X.208 [24] with the addition of "UTF8String" type imported from the hybrid ASN.1 module of RFC 5280 [17]. These additions are imported so as to enhance interoperability by avoiding ambiguity concerning signature algorithms and digest calculation. The following schema requires the use of a "relaxed compiler" to accommodate these two special types.

The ASN.1 in this annex may be converted into the 1997 ([i.1]) syntax by using the Information Object Classes introduced by that version to replace the type "ANY DEFINED BY" (this type not being supported by the 1997 version) and removing the importation of "UTF8String" type, plus amending the module header appropriately.

The ASN.1 implementation of the TSL must be encoded by using the Distinguished Encoding Rules defined by ITU-T Recommendation X.690 [28].

The header of the ASN.1 module is specified as follows:

```
ETSI-TSL-v2-88syntax { itu-t(0) identified-organization(4) etsi(0)
  tsl-specification (2231) id-mod(0) v2-88syntax (1)}
DEFINITIONS EXPLICIT TAGS ::=

BEGIN

-- EXPORTS All
IMPORTS
  -- Internet X.509 Public Key Infrastructure - Certificate and CRL Profile: RFC 5280
  Extensions, Certificate, CertificateSerialNumber, AlgorithmIdentifier,
    UTF8String, SubjectPublicKeyInfo, Name
    FROM PKIX1Explicit88 {iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-explicit(18)}
  KeyIdentifier
    FROM PKIX1Implicit88 {iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-implicit(19)}
  -- Cryptographic Message Syntax (CMS): RFC 5652
  ContentInfo, ContentType, id-signedData, SignedData, EncapsulatedContentInfo,
    SignerInfo
    FROM CryptographicMessageSyntax2004 {iso(1) member-body(2) us(840) rsadsi(113549)
      pkcs(1) pkcs-9(9) smime(16) modules(0) cms-2004(24) };
```

A.1.2 Basic types

The following are basic types used more than once within the ASN.1 module.

A.1.2.1 The `NonEmptyURI` type

The following type is used to carry a non-empty URI.

```
NonEmptyURI ::= IA5String (SIZE (1..MAX))
```

A.1.2.2 The LanguageTag type

The following type is used to carry a language tag according to RFC 5646 [16].

```
LanguageTag ::= PrintableString (SIZE (1..MAX))
```

A.1.2.3 The CountryCode type

The following type is used to carry the country code according to clause 5.1.6.

```
CountryCode ::= PrintableString (SIZE (2))
```

A.1.2.4 The MultiLangPointer type

This definition specifies a format for giving alternative pointers (URIs) to the same text translated in different languages and scripts. The value of the languageTag field MUST be a language tag as specified by RFC 5646 [16] and indicates the language of the text pointed by the URI contained within the companion uRI field. The text pointed by the URI can be expressed by using any format or language (plain text, HTML, XML, etc.).

```
MultiLangPointer ::= SEQUENCE SIZE (1..MAX) OF LangPointer

LangPointer ::= SEQUENCE {
  languageTag LanguageTag,
  uRI         NonEmptyURI
  }
```

A.1.2.5 The MultiLangString type

This definition specifies a format for giving alternative text strings in different languages and scripts. The text field contains plain text, with characters from the ISO 10646 [23] character set without any escape sequence and UTF-8 encoded. The value of the languageTag field MUST be a language tag as specified by RFC 5646 [16] and indicates the language of the text contained within the companion text field.

```
MultiLangString ::= SEQUENCE SIZE (1..MAX) OF LangString

LangString ::= SEQUENCE {
  languageTag LanguageTag,
  string      UTF8String (SIZE (1..MAX))
}
```

A.1.2.6 The PhysicalAndElectronicAddresses type

This definition specifies a format for giving physical addresses in different languages and scripts and for giving the electronic addresses.

The streetAddress, locality, stateOrProvince, postalCode, countryName fields contain plain text, with characters from the ISO 10646 [23] character set without any escape sequence and UTF-8 encoded. The value of the languageTag field MUST be a language tag as specified by RFC 5646 [16] and indicates the language of the text contained within the companion streetAddress, locality, stateOrProvince, postalCode, countryName fields within the same sequence.

The electronicAddresses field MUST include at least one electronic address and MAY include more than one. Each electronic address is a non-empty URI that MUST represent either:

- a RFC 5322 e-mail address, expressed by using the "mailto:" URI scheme as defined by RFC 2368 [11]; or
- a web-site.

```
PhysicalAndElectronicAddresses ::= SEQUENCE {
  physicalDeliveryAddress  MultiLangAddress,
  electronicAddresses      ElectronicAddresses
  }

MultiLangAddress ::= SEQUENCE SIZE (1..MAX) OF LangAddress

LangAddress ::= SEQUENCE {
  languageTag        LanguageTag,
  streetAddress      UTF8String(SIZE (1..MAX)),
  locality           UTF8String(SIZE (1..MAX)),
  stateOrProvince    UTF8String (SIZE (1..MAX)) OPTIONAL,
  postalCode         UTF8String(SIZE (1..MAX)) OPTIONAL,
  countryName        CountryCode
  }

ElectronicAddresses ::= SEQUENCE SIZE (1..MAX) OF NonEmptyURI
```

A.1.3 General Structure

The main structure of the ASN.1 implementation of a TSL is defined as follows:

```
TSL ::= ContentInfo

ToBeSignedTSL ::=SEQUENCE {
  tSLTag                       TSLTag,
  version                      Version,
  sequenceNumber               SequenceNumber,
  tSLType                      TSLType,
  schemeOperatorName           SchemeOperatorName,
  schemeOperatorAddress        SchemeOperatorAddress,
  schemeName                   SchemeName,
  schemeInformationURI         SchemeInformationURI,
  statusDeterminationApproach  StatusDeterminationApproach,
  schemeTypeCommunityRules     [0] SchemeTypeCommunityRules OPTIONAL,
  schemeTerritory              [1] SchemeTerritory OPTIONAL,
  tSLpolicy                    [2] TSLpolicy OPTIONAL,
  historicalInformationPeriod  HistoricalInformationPeriod,
  pointersToOtherTSLs          [3] PointersToOtherTSLs OPTIONAL,
  listIssueDateTime            ListIssueDateTime,
  nextUpdate                   NextUpdate,
  schemeExtensions             [4] Extensions OPTIONAL,
  distributionPoint            [5] DistributionPoints OPTIONAL,
  tSPlist                      TSPlist OPTIONAL
  }
```

A.2 Scheme information fields

A.2.1 The tSLTag field

This field is REQUIRED. It shall facilitate the identification of the TSL as such, when electronic searches are conducted across the Internet. The type of this field is TSLtag, defined as follows:

```
TSLTag ::= NonEmptyURI
```

The tag is implemented as a string (with an embedded URI) whose unique value MUST be:

```
tslTag-value NonEmptyURI ::="http://uri.etsi.org/02231/TSLTag"
```

A.2.2 The version field

This REQUIRED field specifies the version of the TSL format. In this version of the TSL it must have the value "3". The type of this field is Version, defined as follows:

```
Version ::= INTEGER {  v3(3)  }
```

A.2.3 The sequenceNumber field

This REQUIRED field specifies the sequence number of the TSL. At the first release of the TSL, the value of the sequence number shall be "1". The value shall be increased at each subsequent release of the TSL. The type of this field is SequenceNumber, defined as follows:

```
SequenceNumber ::= INTEGER (1..MAX)
```

A.2.4 The tSLType field

This REQUIRED field specifies the type of the TSL. The value SHALL be one of the URIs listed in clause D.2 or another registered URI having the same purpose. The type of this field is tSLType, defined as follows:

```
TSLType ::= NonEmptyURI
```

A.2.5 The schemeOperatorName field

This REQUIRED field specifies the name(s) of the scheme operator. The type of this field is SchemeOperatorName, defined as follows:

```
SchemeOperatorName ::= MultiLangString
```

A.2.6 The schemeOperatorAddress field

This REQUIRED field includes the scheme operator postal address (see clause 5.3.5.1) and the scheme operator electronic address (see clause 5.4.3.2). The type of this field is SchemeOperatorAddress, defined as follows:

```
SchemeOperatorAddress ::= PhysicalAndElectronicAddress
```

A.2.7 The schemeName field

This REQUIRED field specifies the name(s) under which the scheme operates. The type of this field is SchemeName, defined as follows:

```
SchemeName ::= MultiLangString
```

A.2.8 The schemeInformationURI field

This REQUIRED field specifies the URI where users can obtain scheme-specific information. The type of this field is SchemeInformationURI, defined as follows:

```
SchemeInformationURI ::= MultiLangPointer
```

A.2.9 The statusDeterminationApproach field

This REQUIRED field specifies the status determination approach. The value SHALL be one of the URIs listed in clause D.2 or another registered URI having the same purpose. The type of this field is StatusDeterminationApproach, defined as follows:

```
StatusDeterminationApproach  ::= NonEmptyURI
```

A.2.10 The schemeTypeCommunityRules field

This OPTIONAL field is a sequence of registered Uniform Resource Identifiers (URIs), used as unique identifiers when required to indicate one or more sets of rules/policies under which the TSL has been issued. If this field is present, at least one URI MUST be present. The type of this field is SchemeTypeCommunityRules, defined as follows:

```
SchemeTypeCommunityRules ::= SEQUENCE SIZE (1..MAX) OF NonEmptyURI
```

A.2.11 The schemeTerritory field

This OPTIONAL field specifies the country in which the scheme is established. The type of this field is SchemeTerritory, defined as follows:

```
SchemeTerritory ::= CountryCode
```

A.2.12 The tSLpolicy field

This OPTIONAL field can be used to specify the scheme's policy or provide a notice concerning the legal status of the scheme or legal requirements met by the scheme for the jurisdiction in which the scheme is established and/or any constraints and conditions under which the TSL is maintained and offered. It can be provided in multiple languages. This string is either recognized as a registered URI or represents the textual form of the legal notice. The type of this field is TSLpolicy, defined as follows:

```
TSLpolicy  ::= CHOICE {
  pointer  [0] MultiLangPointer,
  text     [1] MultiLangString
  }
```

A.2.13 The historicalInformationPeriod field

This REQUIRED field contains the duration over which historical information in this TSL is provided (see clause 5.3.12). The type of this field is HistoricalInformationPeriod, defined as follows:

```
HistoricalInformationPeriod ::= INTEGER (0..MAX)
```

A.2.14 The pointersToOtherTSLs field

This OPTIONAL field specifies the URI where users can obtain other TSLs. The field can contain a list of tuples holding an URI pointing to the TSL, optional digital identities belonging to the issuer of the pointed TSL and additional information about that TSL. If this field (pointersToOtherTSLs) is present, at least one tuple MUST be present. The additionalInformation field is implementation-specific and it can be empty (zero-length string), free text with characters from ISO 10646 [23], some character-based and machine-readable code (e.g. a URI or a MIME object) or other, with an optional language indication.

The type of this field is PointersToOtherTSLs, defined as follows:

```
PointersToOtherTSLs  ::= SEQUENCE SIZE (1..MAX) OF OtherTSLPointer

OtherTSLPointer ::= SEQUENCE {
  tSLLocation            NonEmptyURI,
  digitalIdentity        ServiceDigitalIdentities OPTIONAL,
  additionalInformation  TSLqualifiers OPTIONAL
  }

ServiceDigitalIdentities ::= SEQUENCE (1..MAX) OF ServiceDigitalIdentity

TSLqualifiers ::= SEQUENCE (1..MAX) OF TSLqualifier

TSLqualifier ::= CHOICE {
  textualQualifier  [0] MultiLangString,
  otherQualifier    [1] OtherQualifier
  }

OtherQualifier ::= SEQUENCE {
  type   OBJECT IDENTIFIER,
  value  ANY DEFINED BY type
  }
```

A.2.14.1 otherQualifier

This clause specifies possible additionalInformation content as otherQualifier field according to clause 5.3.13.1.

```
-- OID for TSLqualifiers
   id-tsl OBJECT IDENTIFIER { itu-t(0) identified-organization(4)
       etsi(0) tsl-specification (2231) }

   id-tsl-qualifiers OBJECT IDENTIFIER ::= { id-tsl tsl-qualifiers (5) }

   id-tslq-tsl-type OBJECT IDENTIFIER ::= { id-tsl-qualifiers (1) }
-- with syntax TSLType defined in A.2.4

   id-tslq-scheme-operator OBJECT IDENTIFIER ::= { id-tsl-qualifiers (2) }
-- with syntax schemeOperatorName defined in name A.2.5

   id-tslq-scheme-name OBJECT IDENTIFIER ::= { id-tsl-qualifiers (3) }
-- with syntax schemeName defined in A.2.7

   id-tslq-scheme-information-uri OBJECT IDENTIFIER ::= { id-tsl-qualifiers (4) }
-- with syntax schemeInformationURI  defined in A.2.8

   id-tslq-scheme-type-community-rules OBJECT IDENTIFIER ::= { id-tsl-qualifiers (5) }
-- with syntax schemeTypeCommunityRules defined in A.2.9

   id-tslq-scheme-territory OBJECT IDENTIFIER ::= { id-tsl-qualifiers (6) }
-- with syntax schemeTerritory defined in A.2.11
```

A.2.15 The listIssueDateTime field

This REQUIRED field gives date and time of the issuance of the TSL, expressed as UTC time. All encoding requirements mandated by the Distinguished Encoding Rules ITU-T Recommendation X.690 [28] apply. In addition, the time indication MUST not include fractional seconds. The type of this field is ListIssueDateTime, defined as follows:

```
ListIssueDateTime ::= GeneralizedTime
```

A.2.16 The nextUpdate field

This REQUIRED field specifies the latest date and time by which the next TSL will be issued expressed as UTC time. All encoding requirements mandated by the Distinguished Encoding Rules ITU-T Recommendation X.690 [28] apply. In addition, the time indication MUST not include fractional seconds. The type of this field is NextUpdate, defined as follows:

```
NextUpdate ::= CHOICE {
  never  NULL,
  update GeneralizedTime
  }
```

A.2.17 The distributionPoints field

This OPTIONAL field specifies the URI where the current TSL is published and where updates to the current TSL can be found. The type of this field is DistributionPoints, defined as follows:

```
DistributionPoints  ::= SEQUENCE SIZE (1..MAX) OF NonEmptyURI
```

A.2.18 The schemeExtensions field

This is an OPTIONAL field useful to carry additional data at the "scheme" hierarchical level. The type of this field is Extensions that is imported from RFC 5280 [17]. The structure of the Extensions type, the meaning of the fields it contains and the processing rules are the same as in RFC 5280 [17]. The additional data are conveyed through one or more "extensions" that MAY be present within the schemeExtensions field. Each "extension" is uniquely identified by the field extnID and may be marked as critical through the critical field. Applications MUST reject the TSL if they encounter a critical "extension" that they do not recognize. However, they MAY ignore a non-critical extension that they do not recognize.

A.2.19 The tSPlist field

This OPTIONAL field includes the list of all TSP information. If present it SHALL contain at least one TSP instance. For each service provider a name field, an alternative trading name, an address, and a pointer to a web page are REQUIRED.

The list of services offered is REQUIRED and at least one service MUST be listed. The type of this field is TSPlist, defined as follows:

```
TSPlist ::=SEQUENCE SIZE (1..MAX) OF TrustServiceProviderInformation

TrustServiceProviderInformation ::= SEQUENCE {
  tSPname            TSPname,
  tSPtradeName       [0] TSPtradeName OPTIONAL,
  tSPaddress         TSPaddress,
  tSPinformationURI  TSPinformationURI,
  tSPextensions      [1] Extensions OPTIONAL,
  listOfServices     [2] ListOfServices
  }
```

A.3 TSP information fields

A.3.1 The tSPname field

This REQUIRED field specifies the name of the Trust Service Provider and supports multiple languages. The type of the field is TSPname, defined as follows:

```
TSPname ::= MultiLangString
```

A.3.2 The tradeName field

This OPTIONAL field contains alternative trading names of the Trust Service Provider and supports multiple languages. The type of this field is TSPtradeName, defined as follows:

```
TSPtradeName ::= MultiLangString
```

A.3.3 The tSPaddress field

This REQUIRED field contains the address of the Trust Service Provider. The type of this field is TSPaddress, defined as follows:

```
TSPaddress ::= PhysicalOrElectronicAddress
```

A.3.4 The tSPinformationURI field

This REQUIRED field contains a pointer to a web page holding service-specific information. The type of this field is TSPinformationURI, defined as follows:

```
TSPinformationURI ::= MultiLangPointer
```

A.3.5 The tSPextensions field

This is an OPTIONAL field useful to carry additional information at the "TSP" hierarchical level. The type of this field is Extensions that is imported from RFC 5280 [17]. The structure of the Extensions type, the meaning of the fields it contains and the processing rules are the same as in RFC 5280 [17]. The additional data are conveyed through one or more "extensions" that MAY be present within the tSPExtensions field. Each "extension" is uniquely identified by the field extnID and may be marked as critical through the critical field. Applications MUST reject the TSL if they encounter a critical "extension" that they do not recognize. However, they MAY ignore a non-critical extension that they do not recognize.

A.3.6 The listOfServices field

This REQUIRED field contains information of a list of Trust Services the TSP offers. At least one service MUST be listed. The type of this field is ListOfServices, defined as follows:

```
ListOfServices ::= SEQUENCE SIZE (1..MAX) OF TSPserviceInformation

TSPserviceInformation ::= SEQUENCE {
  serviceType               ServiceType,
  serviceName               ServiceName,
  serviceDigitalIdentity    ServiceDigitalIdentity,
  currentServiceStatus      ServiceStatus,
  currentStatusStartingTime StatusStartingTime,
  schemeURI                 [0] SchemeURI OPTIONAL,
  tspURI                    [1] TspURI OPTIONAL,
  serviceSupplyPoints       [2] ServiceSupplyPoints OPTIONAL,
  srvcExtensions            [3] Extensions OPTIONAL,
  serviceApprovalHistory    [4] ServiceApprovalHistory OPTIONAL
 }
```

A.4 TSP service information fields

A.4.1 The serviceType field

This REQUIRED field specifies the identifier of the service type. The value SHALL be one of the URIs listed in clause D.2 or another registered URI having the same purpose. The type of this field is ServiceType, defined as follows:

```
ServiceType ::= NonEmptyURI
```

A.4.2 The serviceName field

This REQUIRED field specifies the name under which the service is provided. The type of this field is ServiceName, defined as follows:

```
ServiceName ::= MultiLangString
```

A.4.3 The serviceDigitalIdentity field

This is a REQUIRED field. The service digital identity can be realized in a number of different ways, depending on the service offered. It could be a certificate which can be used to verify electronic signatures of the service provider, a public key or a key identifier or a collection of these types. Each of the included attributes can be used for the identification of the service. How many have to be considered for a complete identification is beyond the scope of the present document, it being dependent on the policy of the TSP as well as that of the user/relying party.

This REQUIRED field MAY be empty; this means that serviceDigitalIdentity MUST be present but no instance of IdentityAttributeTypeAndValue SHALL be. This is implemented by having the content of SET OF empty: according to the Distinguished Encoding Rules ITU-T Recommendation X.690 [28] the tag of SET OF will be present while its content will be zero octets long.

NOTE: The key identifier MUST be used only if there exists an X.509 certificate ITU-T Recommendation X.509 [43] where the subject is the service to be digitally identified. In this case the content of the key identifier MUST be the same as the content of the X.509 SubjectKeyIdentifier extension.

The type of this field is ServiceDigitalIdentity, defined as follows:

```
ServiceDigitalIdentity ::= IdentityAttributeTypeAndValues

IdentityAttributeTypeAndValues ::= SET OF IdentityAttributeTypeAndValue

IdentityAttributeTypeAndValue ::= SEQUENCE {
  type   OBJECT IDENTIFIER,
  value  ANY DEFINED BY type
  }
```

If the service digital identity is a certificate, then the type field MUST assume the following value:

```
id-certificateIdentityType OBJECT IDENTIFIER ::=
        { itu-t(0) identified-organization(4)
                   etsi(0) tsl-specification(2231) identity-types(2) certificate(0) }
```

and the value field MUST be the sequence of octets of a DER-encoded Certificate field imported from RFC 5280 [17].

If the service digital identity is a public key, then the type field MUST assume the following value:

```
id-publicKeyIdentityType OBJECT IDENTIFIER ::=
        { itu-t(0) identified-organization(4)
                   etsi(0) tsl-specification(2231) identity-types(2) public-key(1) }
```

and the value field MUST be the sequence octets of the DER-encoded SubjectPublicKeyInfo field, whose definition MUST be imported from RFC 5280 [17]. The content of SubjectPublicKeyInfo MUST be compliant with RFC 3279 [41] or RFC 4055 [36]; it MAY be compliant with future specifications listing new algorithms and defining the formats for the related parameters.

If the service digital identity is a key identifier, then the type field MUST assume the following value:

```
id-keyIdentifierIdentityType OBJECT IDENTIFIER ::=
        { itu-t(0) identified-organization(4)
                   etsi(0) tsl-specification(2231) identity-types(2) key-identifier(2) }
```

and the value field MUST be the sequence octets of the DER-encoded KeyIdentifier type, whose definition MUST be imported from RFC 5280 [17] and the content of the imported KeyIdentifier MUST be the same as the content of SubjectKeyIdentifier within the Subject Key Identifier extension present in the X.509 certificate issued to the service.

If the service digital identity is a distinguished name, then the type field MUST assume the following value:

```
id-directoryNameIdentityType OBJECT IDENTIFIER ::=
        { itu-t(0) identified-organization(4)
                   etsi(0) tsl-specification(2231) identity-types(2) directory-name(3) }
```

and the value field MUST be the sequence of bytes of the DER-encoded Name type, whose definition MUST be imported from RFC 5280 [17].

A.4.4 The currentServiceStatus field

This REQUIRED field specifies the identifier of the current status of the service. The value SHALL be one of the URIs listed in clause D.2 or another registered URI having the same purpose. The type of this field is ServiceStatus, defined as follows:

```
ServiceStatus  ::= NonEmptyURI
```

A.4.5 The currentStatusStartingTime field

This REQUIRED field specifies the date and time on which the current status became effective. The type of this field is StatusStartingTime, defined as follows:

```
StatusStartingTime ::= GeneralizedTime
```

A.4.6 The schemeURI field

This OPTIONAL field specifies the URI where users can obtain service-specific information provided by the scheme operator. The type of this field is SchemeURI, defined as follows:

```
SchemeURI ::= MultiLangPointer
```

A.4.7 The tspURI field

This OPTIONAL field specifies the URI where users can obtain service-specific information provided by the TSP. The type of this field is TspURI, defined as follows:

```
TspURI ::= MultiLangPointer
```

A.4.8 The serviceSupplyPoints field

This OPTIONAL field carries one or more URIs that indicate the electronic point or points where a service can be accessed. The type of this field is ServiceSupplyPoints, defined as follows:

```
ServiceSupplyPoints ::= SEQUENCE SIZE (1..MAX) OF ServiceSupplyPoint

ServiceSupplyPoint ::= NonEmptyURI
```

A.4.9 The srvcExtensions field

This is an OPTIONAL field useful to carry additional information at the "service" hierarchical level. The type of this field is Extensions that is imported from RFC 5280 [17]. The structure of the Extensions type, the meaning of the fields it contains and the processing rules are the same as in RFC 5280 [17]. The additional data are conveyed through one or more "extensions" that MAY be present within the srvcExtensions field. Each "extension" is uniquely identified by the field extnID and may be marked as critical through the critical field. Applications MUST reject the TSL if they encounter a critical "extension" that they do not recognize. However, they MAY ignore a non-critical extension that they do not recognize.

A.4.10 The serviceApprovalHistory field

This OPTIONAL field provides any historical status information of the service.

This field MAY be absent or present. If present, it MAY be empty; this means that serviceApprovalHistory SHALL be present but no instance of TSPhistoryInformation will be. This is implemented by having the content of SEQUENCE OF empty: according to the Distinguished Encoding Rules ITU-T Recommendation X.690 [28] the tag of SEQUENCE OF will be present while its content will be zero octets long. The history information replicates the current status information. The type of this field is ServiceApprovalHistory, defined as follows:

```
ServiceApprovalHistory ::= SEQUENCE OF TSPhistoryInformation

TSPhistoryInformation ::= SEQUENCE {
  serviceType                ServiceType,
  serviceName                ServiceName,
  serviceDigitalIdentity     ServiceDigitalIdentity,
  previousStatus             ServiceStatus,
  previousStatusStartingTime StatusStartingTime
  srvcExtensions             [0] Extensions OPTIONAL
  }
```

A.5 Service history information fields

A.5.1 The serviceType field

This REQUIRED field specifies the previous service type. Its definition and meaning are as defined in clause A.4.1.

A.5.2 The serviceName field

This REQUIRED field specifies the previous service name. Its definition and meaning are as defined in clause A.4.2.

A.5.3 The serviceDigitalIdentity field

This REQUIRED field specifies the previous service digital identity. Its definition and meaning are as defined in clause A.4.3.

A.5.4 The previousServiceStatus field

This REQUIRED field specifies the identifier of the previous service status. Its definition and meaning are as defined in clause A.4.4.

A.5.5 The previousStatusStartingTime field

This REQUIRED field specifies the date and time on which the previous status became effective. Its definition and meaning are as defined in clause A.4.5.

A.5.6 The srvcExtensions field

This OPTIONAL field specifies the previous service extensions. Its definition and meaning are as defined in clause A.4.9.

A.6 TSL signature fields

A.6.1 The signedTSL field

This REQUIRED field contains the signature value and the signing key information.

This field SHALL contain a signature according to RFC 5652 [37]. The signature MAY include additional security feature provided by TS 101 733 [2]; therefore the content of this field MAY be also compliant with the latter which is in turn compliant with RFC 5652 [37]. The additional informative guidance given in annex F MUST be considered when implementing the signature and selecting the security features.

The value of this field is the octets string of the DER encoding CMS ContentInfo value with the signed-data content type as defined by RFC 5652 [37]. Therefore the CMS contentType field is assigned the OID id-signedData value and the CMS content field contains the octet string of the DER-encoded SignedData type. The CMS eContent field within SignedData SHALL contain the data to be signed, namely the octet string of the DER-encoded ToBeSignedTSL value with the inclusion of the tag and length octets.

The CMS eContentType field MUST be assigned the following OID:

```
id-eContentType-signedTSL OBJECT IDENTIFIER ::=
         { itu-t(0) identified-organization(4)
                    etsi(0) tsl-specification (2231) identifiers (1) tsl-info(0) }
```

According to RFC 5652 [37] the following rules apply:

1) Since the value of eContentType is other than id-data, the value of the Version field within SignedData MUST be "3".

2) For the value of the Version field within SignerInfo the following options are possible: if the CMS SignerIdentifier field is the "CHOICE" issuerAndSerialNumber, then the version MUST be "1". If the SignerIdentifier is subjectKeyIdentifier, then the version MUST be "3".

3) Since the value of eContentType is other than id-data, the signedAttrs field MUST be present and MUST contain at least the following two signed attributes: MessageDigest and ContentType. The value of the former MUST contain the digest calculated over the eContent field. The value of the latter MUST be the same as eContentType, namely id-eContentType-signedTSL.

The following profile specific for signing TSLs applies.

Only one SignerInfo within the SET OF SignerInfos MUST be present, namely only one signature MUST be present.

The certificates field (within SignedData) MUST be either absent or present with only one certificate inside, the one of the signer of TSL. If the signer certificate is present, its type (namely the CHOICE of types among the CertificateChoices) MUST be only the X.509 certificate (namely the certificate CHOICE).

The crls field (within SignedData) MUST be absent.

According to this profile, other signed attributes and also unsigned attributes MAY be present.

A.6.2 The scheme operator identifier

Since this ASN.1 implementation of the signature is based on the CMS specification, it supports the methods natively provided by CMS to identify the scheme operator, namely the signer of TSL; therefore the use of the scheme operator public key as identifier is not supported.

Instead the following combinations are supported by CMS and one of them SHALL be used:

- The issuer/serial number pair only: the issuerAndSerialNumber CHOICE of SignerIdentifier that identifies the scheme operator certificate *not present* within the certificates field within SignedData.
- The issuer/serial number pair with the related X.509 certificate: the issuerAndSerialNumber CHOICE of SignerIdentifier that identifies the scheme operator certificate *present* within the certificates field within SignedData.
- The value of SubjectKeyIdentifier only: the subjectKeyIdentifier CHOICE of SignerIdentifier that identifies the scheme operator certificate *not present* within the certificates field within SignedData; the content of subjectKeyIdentifier MUST be identical to the content of the SubjectKeyIdentifier type of the Subject Key Identifier extension contained within the scheme operator certificate.
- The value of SubjectKeyIdentifier with the related X.509 certificate: the subjectKeyIdentifier CHOICE of SignerIdentifier that identifies the scheme operator certificate *present* within the certificates field within SignedData; the content of subjectKeyIdentifier MUST be identical to the content of the SubjectKeyIdentifier type of the Subject Key Identifier extension contained within the scheme operator certificate.

The choice of one of the listed methods is REQUIRED according to RFC 5652 [37].

Since the inclusion of the signer (i.e. the Scheme Operator) identifier in the signature calculation is REQUIRED as specified in clause 5.7.2, also a signed X.509-certificate identifier MUST be present. This identifier MUST be implemented as a CMS signed attribute in either the following ways.

A.6.2.1 ESS signing certificate attribute

The syntax of the signing certificate attribute is defined in Enhanced Security Services (ESS) RFC 2634 [13] updated by RFC 5035 [38] and further qualified in the present document.

- The sequence of policy information field is not used in the present document.
- The ESS signing-certificate attribute shall be a signed attribute.
- The ESSCertIDv2 attribute SHALL be used to protect the signing certificate.

The certificate identified in ESSCertIDv2 SHALL be used during the signature verification process. If the hash of the certificate does not match the certificate used to verify the signature, the signature SHALL be considered invalid.

This way of implementing the X.509-certificate identifier is identical to the one defined in clause 5.7.3.1 of TS 101 733 [2].

A.6.3 Algorithms and parameters

The algorithms and parameters and their formats supported by the present document for the CMS fields digestAlgorithms (within SignedData and SignerInfo) and signatureAlgorithm (within SignerInfo) are those specified by RFC 3370 [42]. Further algorithms and parameters and their format MAY be specified.

A.7 Extensions defined in the present document

This clause contains all Extensions defined in the present document. These extensions can be used not only within a TSL though but also in any trust service token, where applicable. E.g. the TSLDistributionPoints-extension may be placed into an X.509-certificate to indicate where relevant TSLs may be found.

A.7.1 TSL Extensions OID

This clause specifies the OID for TSL extensions; all TSL Extensions are based on this OID.

```
-- OID for TSL Extensions
   id-tsl OBJECT IDENTIFIER ::= { itu-t(0) identified-organization(4)
       etsi(0) tsl-specification (2231) }
   id-te OBJECT IDENTIFIER ::= { id-tsl extensions(4) }
```

A.7.2 TSLDistributionPoints

This extension is intended to be used outside of TSLs in any trust service token, where applicable. E.g. it may be placed into an X.509-certificate to indicate where relevant TSLs may be found.

```
-- OID for TSLDistributionPoints extension
   id-te-tSLDistributionPoints OBJECT IDENTIFIER ::= { id-te 0 }


-- TSLDistributionPoints extension definition
   tSLDistributionPoints EXTENSION ::= {
       SYNTAX TSLDistributionPoints
       IDENTIFIED BY id-te-tSLDistributionPoints}

   TSLDistributionPoints ::= SEQUENCE SIZE(1..MAX) OF NonEmptyURI
```

A.7.3 ExpiredCertsRevocationInfo

This element has the semantics specified in clause 5.8.2 of the present document. . In consequence this element if present MUST appear within the `tsl:ServiceInformationExtensions` element. Use of this extension outside TSL is allowed but not covered by present document.

```
-- OID ExpiredCertsRevocationInfo extension
   id-te-expiredCertsRevocationInfo OBJECT IDENTIFIER ::= { id-te 1 }

-- expiredCertsRevocationInfo extension definition
   expiredCertsRevocationInfo EXTENSION ::= {
       SYNTAX ExpiredCertsRevocationInfo
       IDENTIFIED BY id-te-expiredCertsRevocationInfo }

   ExpiredCertsRevocationInfo ::= GeneralizedTime
```

A.7.4 AdditionalServiceInformation

This element has the semantics specified in clause 5.8.2 of the present document. In consequence this element if present MUST appear within the `tsl:ServiceInformationExtensions` element. Use of this extension outside TSL is allowed but not covered by present document.

```
-- OID for AdditionalServiceInformation extension
   id-te-additionalServiceInformation OBJECT IDENTIFIER ::= { id-te 2 }

-- additionalServiceInformation extension definition
   additionalServiceInformation EXTENSION ::= {
       SYNTAX AdditionalServiceInformation
       IDENTIFIED BY id-te-additionalServiceInformation }

 AdditionalServiceInformation::= SEQUENCE {
    additionalServiceInformationURI MultiLangPointer,
    informationValue UTF8String(SIZE (1..MAX)),
    otherQualifier OtherQualifier OPTIONAL

}
```

Annex B (normative): Implementation in XML

This annex specifies an XML schema to be used when implementing an XML-version of the present document. The field names used reflect those assigned to fields in clause 5.

B.1 Structure of the Trust-service Status List

This annex specifies an XML schema to be used when implementing an XML-version of the present document. The field names used reflect those assigned to fields in clause 5.

B.1.1 General Rules

This clause contains general rules that apply to the XML version of the TSL.

- Applications MUST use UTF-8 encoding for XML TSLs.
- All time values are in Coordinated Universal Time (UTC) expressed as Zulu. Its value MUST NOT include fractional seconds.

B.1.2 XML-namespace and basic types

The XML namespace URI that must be used by implementations of the present document is:
http://uri.etsi.org/02231/v2#

The following namespace declarations apply for the XML Schema definitions throughout the present document:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema     targetNamespace="http://uri.etsi.org/02231/v2#"
                xmlns:tsl="http://uri.etsi.org/02231/v2#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
                    elementFormDefault="qualified"
                    attributeFormDefault="unqualified">

<xsd:import namespace="http://www.w3.org/XML/1998/namespace"
schemaLocation="http://www.w3.org/2001/xml.xsd"/>
<xsd:import namespace="http://www.w3.org/2000/09/xmldsig#"
schemaLocation="http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/xmldsig-core-schema.xsd"/>
```

Several types are better specified separately. These types are specified in the clauses B.1.2.1 through B.1.2.6.

B.1.2.1 The `InternationalNamesType` and `MultiLangString` Types

The `InternationalNamesType` specifies a format for giving alternative names in different languages and scripts.

It is built on the `MultiLangNormStringType` type. This type contains:

- A non-empty normalized string whose contents follow the rules established for the type `xsd:normalizedString` defined in XML Schema Part 2 [32].
- The `xml:lang` attribute identifying the language used in the string.

The `MultiLangNormStringType` type is used thorough the present document whenever there is the possibility to use normalized textual information in different languages as specified in RFC 5646 [16].

In addition, the `MultiLangStringType` type is defined for those strings that require a qualification of the language they are written but do not require normalization.

All of them are based on two non empty string types: `NonEmptyStringType` for regular strings and `NonEmptyNormStringType` for normalized strings.

Below follow their schema definitions.

```
<xsd:complexType name="InternationalNamesType">
    <xsd:sequence>
        <xsd:element name="Name" type="tsl:MultiLangNormStringType"
        maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MultiLangNormStringType">
    <xsd:complexContent>
        <xsd:extension base="tsl:NonEmptyNormalizedString">
            <xsd:attribute ref="xml:lang" use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="MultiLangStringType">
    <xsd:complexContent>
        <xsd:extension base="tsl:NonEmptyString">
            <xsd:attribute ref="xml:lang" use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:simpleType name="NonEmptyString">
    <xsd:restriction base="xsd:string">
        <xsd:minLength value="1"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="NonEmptyNormalizedString">
    <xsd:restriction base="xsd:normalizedString">
        <xsd:minLength value="1"/>
    </xsd:restriction>
</xsd:simpleType>
```

B.1.2.2 The `AddressType` Type

This type is used for addresses holding postal addresses and electronic addresses.

```
<xsd:complexType name="AddressType">
    <xsd:sequence>
        <xsd:element ref="tsl:PostalAddresses"/>
        <xsd:element ref="tsl:ElectronicAddress"/>
    </xsd:sequence>
</xsd:complexType>
```

B.1.2.3 The `PostalAddresses` Element

The `PostalAddressses` element contains a list of `PostalAddress` element. Each `PostalAddress` element contains a postal address in a specific language and script identified by the `xml:lang` attribute.

```
<xsd:element name="PostalAddresses" type="tsl:PostalAddressListType"/>
<xsd:complexType name="PostalAddressListType">
    <xsd:sequence>
        <xsd:element ref="tsl:PostalAddress" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="PostalAddress" type="tsl:PostalAddressType"/>
<xsd:complexType name="PostalAddressType">
    <xsd:sequence>
        <xsd:element name="StreetAddress" type="tsl:NonEmptyString"/>
        <xsd:element name="Locality" type="tsl:NonEmptyString"/>
        <xsd:element name="StateOrProvince" type="tsl:NonEmptyString" minOccurs="0"/>
        <xsd:element name="PostalCode" type="tsl:NonEmptyString" minOccurs="0"/>
        <xsd:element name="CountryName" type="tsl:NonEmptyString"/>
    </xsd:sequence>
    <xsd:attribute ref="xml:lang" use="required"/>
</xsd:complexType>
```

B.1.2.4 The ElectronicAddressType Type

The ElectronicAddressType Type allows the specification of one electronic address.

```
<xsd:element name="ElectronicAddress" type="tsl:ElectronicAddressType"/>
<xsd:complexType name="ElectronicAddressType">
    <xsd:sequence >
        <xsd:element name="URI" type="tsl:NonEmptyURIType" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
```

The contents of each URI element MUST represent either a RFC 5322 [14] e-mail address, expressed by using the "mailto:" URI scheme as defined by RFC 2368 [11], or a web site address.

B.1.2.5 Types for managing the extensions

The present document allows for extending the content of certain elements in TSLs. This clause defines the elements and types that will be used for such purposes. Below follow their schema definition.

```
<xsd:complexType name="AnyType" mixed="true">
    <xsd:sequence minOccurs="0" maxOccurs="unbounded">
        <xsd:any processContents="lax"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="Extension" type="tsl:ExtensionType"/>
<xsd:complexType name="ExtensionType">
    <xsd:complexContent>
        <xsd:extension base="tsl:AnyType">
        <xsd:attribute name="Critical" type="xsd:boolean" use="required" />
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ExtensionsListType">
    <xsd:sequence>
        <xsd:element ref="tsl:Extension" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
```

AnyType type allows for any kind of content. ExtensionType is derived from AnyType by extension. Its Critical attribute indicates whether this element is critical or not. The ExtensionListType is an unbounded list of Extension elements.

Processing of Critical attribute MUST be as the one defined by RFC 5280 [17] for the critical field of extensions of X.509 v3 certificates. Applications MUST reject the TSL if they encounter a critical extension that they do not recognize. However, they MAY ignore a non-critical extension that they do not recognize.

B.1.2.6 Types for URIs

The present document defines new derived types from `xsd:anyURI`. Their schema definition is shown below.

```
<xsd:simpleType name="NonEmptyURIType">
    <xsd:restriction base="xsd:anyURI">
        <xsd:minLength value="1"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="NonEmptyMultiLangURIType">
    <xsd:complexContent>
        <xsd:extension base="tsl:NonEmptyURIType">
            <xsd:attribute ref="xml:lang" use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="NonEmptyMultiLangURIListType">
    <xsd:sequence>
        <xsd:element name="URI" type="tsl:NonEmptyMultiLangURIType" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="NonEmptyURIListType">
    <xsd:sequence>
        <xsd:element name="URI" type="tsl:NonEmptyURIType" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
```

An element of `NonEmptyURIType` type contains a non empty URI value.

An element of `NonEmptyMultiLangURIType` contains a non empty URI value pointing to a resource written in the language that is signalled by the `xml:lang` attribute.

An element of `NonEmptyMultiLangURIListType` contains a sequence of non empty URI values pointing to a resource written in the language that is signalled by the `xml:lang` attribute.

An element of `NonEmptyURIListType` contains a sequence of non empty URI values.

B.1.3 The `TrustServiceStatusList` element

The `TrustserviceStatusList` element is the root element of an XML TSL. An implementation must generate *laxly schema valid* [XML-schema] `TrustserviceStatusList` elements as specified by the following schema.

```
<xsd:element name="TrustServiceStatusList" type="tsl:TrustStatusListType"/>
<xsd:complexType name="TrustStatusListType">
        <xsd:sequence>
            <xsd:element    ref="tsl:SchemeInformation"/>
            <xsd:element    ref="tsl:TrustServiceProviderList" minOccurs="0"/>
            <xsd:element    ref="ds:Signature"/>
        </xsd:sequence>
        <xsd:attribute name="TSLTag" type="tsl:TSLTagType" use="required"/>
        <xsd:attribute name="Id" type="xsd:ID" use="optional"/>
</xsd:complexType>
```

The optional attribute Id may be used to make a reference to the TrustServiceStatusList element.

B.1.3.1 The `TSLTag` attribute

This REQUIRED attribute shall facilitate the identification of the TSL as such, when electronic searches are conducted across the Internet. It will be a string with a fixed value. Its schema definition follows.

```
<xsd:simpleType name="TSLTagType">
<xsd:restriction base="xsd:anyURI">
    <xsd:enumeration
    value="http://uri.etsi.org/02231/TSLTag"/>
</xsd:restriction>
</xsd:simpleType>
```

B.2 The SchemeInformation element

The SchemeInformation element is a container structure for all the elements giving detailed information about the scheme.

```
<xsd:element name="SchemeInformation" type="tsl:TSLSchemeInformationType"/>
<xsd:complexType name="TSLSchemeInformationType">
    <xsd:sequence>
        <xsd:element name="TSLVersionIdentifier" type="xsd:integer" fixed="3"/>
        <xsd:element name="TSLSequenceNumber" type="xsd:positiveInteger"/>
        <xsd:element ref="tsl:TSLType"/>
        <xsd:element ref="tsl:SchemeOperatorName"/>
        <xsd:element name="SchemeOperatorAddress" type="tsl:AddressType"/>
        <xsd:element ref="tsl:SchemeName"/>
        <xsd:element ref="tsl:SchemeInformationURI"/>
        <xsd:element name="tsl:StatusDeterminationApproach"
          type="tsl: NonEmptyURIType"/>
        <xsd:element ref="tsl:SchemeTypeCommunityRules" minOccurs="0"/>
        <xsd:element ref="tsl:SchemeTerritory" minOccurs="0"/>
        <xsd:element ref="tsl:PolicyOrLegalNotice" minOccurs="0"/>
        <xsd:element name="HistoricalInformationPeriod" type="xsd:nonNegativeInteger"/>
        <xsd:element ref="tsl:PointersToOtherTSL"  minOccurs="0"/>
        <xsd:element name="ListIssueDateTime" type="xsd:dateTime"/>
        <xsd:element ref="tsl:NextUpdate"/>
        <xsd:element ref="tsl:DistributionPoints" minOccurs="0"/>
        <xsd:element name="SchemeExtensions" type="tsl:ExtensionsListType" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

B.2.1 The TSLVersionIdentifier element

This REQUIRED element specifies the version of the TSL format. In this version of the TSL it must have the value "3".

B.2.2 The TSLSequenceNumber element

This REQUIRED element specifies the sequence number of the TSL. At the first release of the TSL, the value of the sequence number shall be "1". The value shall be increased by "1" at each subsequent release of the TSL.

B.2.3 The TSLType element

This REQUIRED element specifies the type of the TSL. Its values are URIs as those listed in clause D.2 or another one registered and described by the scheme operator or another entity. Its schema definition follows.

```
<xsd:element name="TSLType" type="tsl:NonEmptyURIType"/>
```

B.2.4 The SchemeOperatorName element

This REQUIRED element specifies the name(s) under which the scheme operator does business or is given its mandate. Its schema definition follows.

```
<xsd:element name="SchemeOperatorName" type="tsl:InternationalNamesType"/>
```

B.2.5 The SchemeOperatorAddress element

This REQUIRED element contains the address of the scheme operator.

B.2.6 The SchemeName element

This REQUIRED element specifies the name(s) under which the scheme operates. Its schema definition follows.

```
<xsd:element name="SchemeName" type="tsl:InternationalNamesType"/>
```

B.2.7 The SchemeInformationURI element

This REQUIRED element contains the URIs where users can obtain scheme-specific information. Its schema definition follows.

```
<xsd:element name="SchemeInformationURI" type="tsl:NonEmptyMultiLangURIListType"/>
```

B.2.8 The StatusDeterminationApproach element

This REQUIRED element specifies the status determination approach (see clause 5.3.8). Its value may be one of the URIs listed in clause D.2 or any other URI value registered and described by the scheme operator or another entity.

B.2.9 The SchemeTypeCommunityRules element

This OPTIONAL element contains a sequence of registered URIs, used as unique identifier when it is required to indicate one or more sets of rules/policies under which the TSL has been issued. Its schema definition follows.

```
<xsd:element name="SchemeTypeCommunityRules" type="tsl:NonEmptyURIListType" />
```

B.2.10 The SchemeTerritory element

This OPTIONAL element specifies the country in which the scheme is established. See clause 5.3.10 for a discussion of its contents. Its schema definition follows.

```
<xsd:element name="SchemeTerritory" type="tsl:SchemeTerritoryType"/>
<xsd:simpleType name="SchemeTerritoryType">
    <xsd:restriction base="xsd:string">
        <xsd:length value="2"/>
    </xsd:restriction>
</xsd:simpleType>
```

B.2.11 The PolicyOrLegalNotice element

This OPTIONAL element MAY be used to specify the scheme's policy or provide a notice concerning the legal status of the scheme or legal requirements met by the scheme for the jurisdiction in which the scheme is established and/or any constraints and conditions under which the TSL is maintained and offered. It can be provided in multiple languages. This string is either recognized as a registered URI or represents the textual form of the legal notice. Its schema definition follows.

```
<xsd:element name="PolicyOrLegalNotice" type="tsl:PolicyOrLegalnoticeType"/>
<xsd:complexType name="PolicyOrLegalnoticeType">
    <xsd:choice>
        <xsd:element name="TSLPolicy" type="tsl:NonEmptyMultiLangURIType"
            maxOccurs="unbounded"/>
        <xsd:element name="TSLLegalNotice" type="tsl:MultiLangStringType"
            maxOccurs="unbounded"/>
</xsd:choice>
</xsd:complexType>
```

B.2.12 The HistoricalInformationPeriod element

This REQUIRED element contains the duration over which historical information in this TSL is provided (see clause 5.3.12).

B.2.13 The PointersToOtherTSL element

This OPTIONAL element specifies URIs where users can obtain other TSLs. The OtherTSLPointersType specifies a list of OtherTSLPointer elements, each holding a URI pointing to the TSL, optional digital identities belonging to the issuer of the pointed TSL and additional information about that TSL, which is implementation-specific.

```
<xsd:element name="PointersToOtherTSL" type="OtherTSLPointersType"/>
<xsd:complexType name="OtherTSLPointersType">
    <xsd:sequence>
        <xsd:element ref="OtherTSLPointer" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="OtherTSLPointer" type="tsl:OtherTSLPointerType"/>
<xsd:complexType name="OtherTSLPointerType">
    <xsd:sequence>
        <xsd:element ref="tsl:ServiceDigitalIdentities" minOccurs=0/>
        <xsd:element name="TSLLocation" type="tsl:NonEmptyURIType"/>
        <xsd:element ref="tsl:AdditionalInformation"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="ServiceDigitalIdentities" type="tsl:ServiceDigitalIdentityListType"/>

<xsd:complexType name="ServiceDigitalIdentityListType">
    <xsd:sequence>
        <xsd:element ref="ServiceDigitalIdentity" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="AdditionalInformation" type="tsl:AdditionalInformationType"/>
<xsd:complexType name="AdditionalInformationType">
    <xsd:choice maxOccurs="unbounded">
        <xsd:element name="TextualInformation" type="tsl:MultiLangStringType"/>
        <xsd:element name="OtherInformation" type="tsl:AnyType"/>
    </xsd:choice>
</xsd:complexType>
```

The AdditionalInformation element may contain a textual information within the TextualInformation element or any other type of information qualifying the pointed TSL, within the element OtherInformation.

B.2.13.1 Already identified contents of OtherInformation element

This clause identifies a number of already defined elements as potential contents of OtherInformation element. This list is shown below:

1) TSLType element whose syntax and semantics have been specified in clause B.2.3 of the present document.

2) SchemeOperatorName element whose syntax and semantics have been specified in clause B.2.4 of the present document.

3) SchemeName element whose syntax and semantics have been specified in clause B.2.6 of the present document.

4) SchemeInformationURI element whose syntax and semantics have been specified in clause B.2.7 of the present document.

5) SchemeTypeCommunityRules element whose syntax and semantics have been specified in clause B.2.9 of the present document.

6) SchemeTerritory element whose syntax and semantics have been specified in clause B.2.10 of the present document.

7) MimeType element defined as follows:

```
<element name="MimeType" type="xsd:string"/>
```

The list above must not be considered a closed list. The present document does not preclude adding any other type of content to `OtherInformation` element.

B.2.14 The `ListIssueDateTime` element

This REQUIRED element specifies the date and time of the issuance of the TSL.

B.2.15 The `NextUpdate` element

This REQUIRED element specifies the latest date and time by which the TSL will be next issued. Its schema definition follows.

```
<xsd:element name="NextUpdate" type="tsl:NextUpdateType"/>
<xsd:complexType name="NextUpdateType">
    <xsd:sequence>
        <xsd:element name="dateTime" type="xsd:dateTime" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

The `NextUpdate` element MAY be an empty element. This MUST occur when a scheme manager issues its last TSL before ceasing its activities. An empty `NextUpdate` element indicates that this will be the last issuance of a TSL by the scheme manager.

B.2.16 The `SchemeExtensions` element

This OPTIONAL element allows for the inclusion of additional information on a scheme. The specific content of such additional information is left open.

B.2.17 The `DistributionPoints` element

This field is OPTIONAL. If used, it SHALL specify the URI where the current TSL is published and where updates to the current TSL can be found.

```
<xsd:element name="DistributionPoints" type="tsl: ElectronicAddressType"/>
```

B.2.18 The `TrustServiceProviderList` element

This element contains all the information related to all the TSPs recognized by the scheme. It is a list of `TrustServiceProvider` elements, each one containing all the information related to one TSP. If present it SHALL contain at least one `TrustServiceProvider` element. For each TSP, the list of services offered is REQUIRED and at least on service MUST be listed. Their schema definitions follow.

```
<xsd:element name="TrustServiceProviderList" type="tsl:TrustServiceProviderListType"/>
<xsd:complexType name="TrustServiceProviderListType">
    <xsd:sequence>
        <xsd:element ref="tsl:TrustServiceProvider" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="TrustServiceProvider" type="tsl:TSPType"/>
<xsd:complexType name="TSPType">
    <xsd:sequence>
        <xsd:element ref="tsl:TSPInformation"/>
        <xsd:element ref="tsl:TSPServices"/>
    </xsd:sequence>
</xsd:complexType>
```

B.3 The TSPInformation element

The TSPInformation element has the following structure.

```
<xsd:element name="TSPInformation" type="tsl:TSPInformationType"/>
<xsd:complexType name="TSPInformationType">
    <xsd:sequence>
        <xsd:element name="TSPName" type="tsl:InternationalNamesType"/>
        <xsd:element name="TSPTradeName" type="tsl:InternationalNamesType"
                minOccurs="0"/>
        <xsd:element name="TSPAddress" type="tsl:AddressType"/>
        <xsd:element name="TSPInformationURI"
                      type="tsl:NonEmptyMultiLangURIListType"/>
        <xsd:element name="TSPInformationExtensions" type="tsl:ExtensionsListType"
              minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

B.3.1 The TSPName element

This REQUIRED element contains the name of the TSP.

B.3.2 The TSPTradeName element

This OPTIONAL element contains alternative trading names of the TSP.

B.3.3 The TSPAddress element

This REQUIRED element contains the address of the TSP.

B.3.4 The TSPInformationURI element

This REQUIRED element contains a pointer to a web page holding service-specific information.

B.3.5 The TSPInformationExtensions element

This OPTIONAL element allows for the inclusion of additional information on a Trust Services Provider. The specific content of such additional information is left open.

B.3.6 The TSPServices element

This element contains information of a list of Trust Services the TSP offers. It is a sequence of TSPService elements, whose contents are described with detail in clause B.4.

```
<xsd:element name="TSPServices" type="tsl:TSPServicesListType"/>
<xsd:complexType name="TSPServicesListType">
    <xsd:sequence>
        <xsd:element ref="tsl:TSPService"maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="TSPService" type="tsl:TSPServiceType"/>
<xsd:complexType name="TSPServiceType">
    <xsd:sequence>
        <xsd:element ref="tsl:ServiceInformation"/>
        <xsd:element ref="tsl:ServiceHistory" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

B.4 The ServiceInformation element

The ServiceInformation element is a container element holding information about a specific service.

```
<xsd:element name="ServiceInformation" type="tsl:TSPServiceInformationType"/>
<xsd:complexType name="tsl:TSPServiceInformationType">
    <xsd:sequence>
        <xsd:element ref="tsl:ServiceTypeIdentifier"/>
        <xsd:element name="ServiceName" type="tsl:InternationalNamesType"/>
        <xsd:element ref="tsl:ServiceDigitalIdentity"/>
        <xsd:element ref="tsl:ServiceStatus"/>
        <xsd:element name="StatusStartingTime" type="xsd:dateTime"/>
        <xsd:element name="SchemeServiceDefinitionURI"
            type="tsl:NonEmptyMultiLangURIListType" minOccurs="0"/>
        <xsd:element ref="tsl:ServiceSupplyPoints" minOccurs="0"/>
        <xsd:element name="TSPServiceDefinitionURI"
            type="tsl:NonEmptyMultiLangURIListType" minOccurs="0"/>
        <xsd:element name="ServiceInformationExtensions"
            type="tsl:ExtensionsListType" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

B.4.1 The ServiceTypeIdentifier element

This REQUIRED element specifies the identifier of the service type. Its value may be one of the URIs listed in clause D.2 or any other URI value registered and described by the scheme operator or another entity.

```
<xsd:element name="ServiceTypeIdentifier" type="tsl: NonEmptyURIType"/>
```

B.4.2 The ServiceName element

This REQUIRED element specifies the name under which the service is provided.

B.4.3 The ServiceDigitalIdentity element

This is a REQUIRED field. This element MAY be empty or contain a number of several elements. Each element contains alternative information for identifying the same service. When identification is based on a public key they borrow their contents from XML-Signature [34] specification. In these cases implementations MAY use one or several of the following three representations for a key:

1) A ds:Keyvalue element.

2) The X509SKI element.

3) The X509Certificate element.

4) Any other element defined in the present document or by the schema operator

Implementations MAY also use a Distinguished Name (DN).

Applications MUST implement the X509Certificate,the X509SKI and X509SubjectName elements exactly as specified in XML-Signature [34] when they use them. Element:X509SubjectName will contain a Distinguished Name encoded as established by XML-Signature [34] in its clause 4.4.4.

The X509SKI element MAY be used only if there exists a X.509 certificate whose subject is the service to be identified. In this case, the content of this element MUST be the same as the content of the SubjectKeyIdentifier extension of the aforementioned certificate.

The number of elements required for identifying a service depends on the TSP policy as well as of the relying party, and any further consideration on this topic are beyond the scope of the present document.

```
<xsd:element name="ServiceDigitalIdentity" type="tsl:DigitalIdentityListType"/>
<xsd:complexType name="DigitalIdentityListType">
    <xsd:sequence>
        <xsd:element name="DigitalId" type="tsl:DigitalIdentityType" minOccurs="0"
        maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DigitalIdentityType">
    <xsd:choice>
        <xsd:element name="X509Certificate" type="xsd:base64Binary"/>
        <xsd:element name="X509SubjectName" type="xsd:string"/>
        <xsd:element ref="ds:KeyValue" />
        <xsd:element name="X509SKI" type="xsd:base64Binary"/>
        <xsd:element name="Other" type="tsl:AnyType"/>
    </xsd:choice>
</xsd:complexType>
```

In addition, the present document defines the following elements that can be added making use of the `Other` element:

```
<element name="X509CertificateLocation" type="tsl:NonEmptyURIType"/>
<element name="PublicKeyLocation" type="tsl:NonEmptyURIType"/>
```

Additional element can be defined and added by the schema operator using the `Other` element.

B.4.4 The `ServiceStatus` element

This REQUIRED element specifies the identifier of the status of the service. See clause 5.5.4 for an explanation of its contents. Its schema definition follows. Its value may be one of the URIs listed in clause D.2.

```
<xsd:element name="ServiceStatus" type="tsl:NonEmptyURIType"/>
```

B.4.5 The `StatusStartingTime` element

This REQUIRED element specifies the date and time on which the current status became effective.

B.4.6 The `SchemeServiceDefinitionURI` element

This OPTIONAL element specifies the URI where users can obtain service-specific information provided by the scheme operator.

B.4.7 The `ServiceSupplyPoints` element

This element contains a sequence of `ServiceSupplyPoint` elements, each one being a non-empty URI that points to the place where users and relying parties may gain access to the service.

```
<xsd:element name="ServiceSupplyPoints" type="tsl:ServiceSupplyPointsType"/>
<xsd:complexType name="ServiceSupplyPointsType">
    <xsd:sequence maxOccurs="unbounded">
        <xsd:element name="ServiceSupplyPoint" type="tsl:NonEmptyURIType"/>
    </xsd:sequence>
</xsd:complexType>
```

B.4.8 The `TSPServiceDefinitionURI` element

This OPTIONAL field specifies the URI where users can obtain service-specific information provided by the TSP.

B.4.9 The `ServiceInformationExtensions` element

This OPTIONAL element allows for the inclusion of additional information on a service. The specific content of such additional information is left open.

B.4.10 The ServiceHistory element

This OPTIONAL field provides any historical status information.

```
<xsd:element name="ServiceHistory" type="tsl:ServiceHistoryType"/>
```

B.5 The ServiceHistory type

This element is a sequence of ServiceHistoryInstance elements. Each one has a content as specified in clause 5.6 and equivalent to the information contained in clause 5.5 with the addition of the ServiceInformationExtensions element. For XML, the relevant fields have been specified in clauses B.4.1 through B.4.5 (representing clauses 5.6.1 through 5.6.6 as well as clauses 5.5.1 through 5.5.5 inclusive, and clause 5.5.9). The ServiceInformationExtensions element is already specified in clause B.4.9.

This element MAY be present or absent. If present it MAY be empty, for signalling that so far no history has been yet built. Its schema definition follows.

```
<xsd:element name="ServiceHistory" type="tsl:ServiceHistoryType"/>
<xsd:complexType name="ServiceHistoryType">
    <xsd:sequence>
        <xsd:element ref="tsl:ServiceHistoryInstance" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:element name="ServiceHistoryInstance" type="tsl:ServiceHistoryInstanceType"/>
<xsd:complexType name="ServiceHistoryInstanceType">
    <xsd:sequence>
        <xsd:element ref="tsl:ServiceTypeIdentifier"/>
        <xsd:element name="ServiceName" type="tsl:InternationalNamesType"/>
        <xsd:element ref="tsl:ServiceDigitalIdentity"/>
        <xsd:element ref="tsl:ServiceStatus"/>
        <xsd:element name="StatusStartingTime" type="xsd:dateTime"/>
        <xsd:element name="ServiceInformationExtensions" type="tsl:ExtensionsListType"
        minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

B.6 The Signature element

The present document allows the use of XML-Signature [34] based signatures for signing a TSL: this includes use of TS 101 903 [35] signatures (see clause F.3 for further discussion). The TSL-structure contains a ds:Signature element that represents an enveloped signature-type. The present document mandates the following constraints to any XML-Signature [34]-based signature applied to a TSL:

1) It MUST be an enveloped signature.

2) Its ds:SignedInfo element MUST contain a ds:Reference element with the URI attribute set to a value referencing the TrustServiceStatusList element enveloping the signature itself. This ds:Reference element MUST satisfy the following requirements:

   a) It MUST contain only one ds:Transforms element.

   b) This ds:Transforms element MUST contain two ds:Transform elements. The first one will be one whose Algorithm attribute indicates the enveloped transformation with the value: "http://www.w3.org/2000/09/xmldsig#enveloped-signature". The second one will be one whose Algorithm attribute instructs to perform the exclusive canonicalization "http://www.w3.org/2001/10/xml-exc-c14n#".

3) ds:CanonicalizationMethod MUST be "http://www.w3.org/2001/10/xml-exc-c14n#".

4) It MAY have other ds:Reference elements.

Rules 2 and 3 ensure that the enveloping `TrustServiceStatusList` element is actually signed as mandated by the processing model in clause 4.3.3.3 of XML-Signature [34] (with reference to same-document URI references). They also ensure that if relative referencing mechanisms are used in the `ds:Reference` element, the `TrustServiceStatusList` may be safely inserted within other xml documents.

Rule 4 allows, among other things, for inclusion of signed properties in the signature, like the ones standardized in TS 101 903 [35].

B.6.1 The scheme identification

As stated in clause 5.7.2, in a signed TSL the signature MUST also cover the scheme identification. This requirement may be fulfilled by standard mechanisms provided by both XML-Signature [34] and TS 101 903 [35].

When a plain XML-Signature [34] signature is generated, one of the following elements MUST be present within the `ds:KeyInfo`'s child element, `ds:X509Data`: a `ds:X509Certificate` element containing an X.509 certificate ITU-T Recommendation X.509 [43], a `ds:X509SKI` element containing the `SubjectKeyIdentifier` extension, or an XML element containing a public key as established within XML-Signature [34] (for RSA and DSA public keys) or the corresponding specification (as new XML formats for carrying public key information are defined, such as that in RFC 4050 [20] for Elliptic Curve Algorithm public keys).

B.6.1.1 The scheme operator identifier in XAdES signatures

TS 101 903 [35] defines the `xades:SigningCertificate` as a signed property that contains an identifier of the signer's certificate and its digest. This is therefore an effective way of securing the scheme operator identifier (see clause F.3 for further discussion).

Even when the `xades:SigningCertificate` property is present, the current document does not prevent the inclusion of any of the three elements mentioned in the previous clause within the `ds:KeyInfo`'s child element `ds:X509Data`.

Should a `ds:X509Certificate` containing the signer's certificate be present within a XAdES signature as a child of a `ds:X509Data` within `ds:KeyInfo`, its serial number and issuer identifier MUST match the serial number and issuer identifier present in the `xades:SigningCertificate` signed property.

Should the child of ds:X509Data element be a `ds:X509SKI` or an element encapsulating a public key, its contents MUST be consistent with the contents of the `xades:SigningCertificate` signed property, if present.

B.6.2 Algorithm and parameters

The algorithms, their parameters and formats supported by the present document are those supported by XML-Signature [34]. Further algorithms, parameters and their format MAY be specified elsewhere, e.g. as for the Elliptic Curve Signature Algorithm (ECDSA) in RFC 4050 [20].

B.7 Elements and types for TSL extensions

This clause defines general use TSL extensions.

Elements that may be part of scheme level extensions MUST appear within tsl:`SchemeExtensions` element.

Elements that may be part of TSP level extensions MUST appear within tsl:`TSPInformationExtensions` element.

Elements that may be part of Service level extensions MUST appear within tsl:`ServiceInformationExtensions` element.

B.7.1 The `ExpiredCertsRevocationInfo` element

This element has the semantics specified in clause 5.8.1 of the present document. In consequence this element if present MUST appear within the `tsl:ServiceInformationExtensions` element.

Below follows its XML schema definition:

```
<xsd:element name="ExpiredCertsRevocationInfo" type="xsd:dateTime"/>
```

B.7.2 The `AdditionalServiceInformation` element

This element has the semantics specified in clause 5.8.2 of the present document. In consequence this element if present MUST appear within the `tsl:ServiceInformationExtensions` element.

Below follows its XML schema definition:

```
<xsd:element name="AdditionalServiceInformation" type="tsa:AdditionalServiceInformationType"/>
<xsd:complexType name="AdditionalServiceInformationType">
    <xsd:sequence>
        <xsd:element name="URI" type="tsl:NonEmptyMultiLangURIType"/>
        <xsd:element name="InformationValue" type="xsd:string" minOccurs="0"/>
        <xsd:element name="OtherInformation" type="tsl:AnyType" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
```

Annex C (normative): ASN.1 and XML files

C.1 Electronic attachment

The present document has an associated electronic document "ts_102231v030102p0.zip" that contains the ASN.1 module, XML, LDAP and eSig Directive Extensions schemas that are integral parts of the present document and further described below.

CAVEAT: In the event that any part of the module and/or schemas within this electronic attachment are in conflict with the text of either annexes A, B or L then those annexes shall prevail as the authoritative sources.

C.2 ASN.1 module

The ASN.1 module is held in file "ts_102231v030102_asn.asn". For the purpose of integrity checking, the hash values of this file are:

MD-5: cc5e7274cd71443fe5321a73e166418a

SHA-1: 9a62938d6204e880f0950db3f34ae2092748cf6e

SHA-256: 84150f920a3f134bb25956ef17d4edd27ed587d0d002de33895fbbe699453a3e

C.3 XML schema

This XML schema is held in the following files:

1. "ts_102231v030102_xsd.xsd" containing the base schema definitions. For the purpose of integrity checking, the hash values of this file are:

   MD-5: ef3964a7ffe881462ddbdf39259ca73b

   SHA-1: 99bcf0dff4b9872de5f48ccdd8ab2cc11c75d687

   SHA-256: 3867a5db3141462d3ea6b82a87b4124dfcd422885ec987260c2c52a00af14eb5

2. "ts_102231v030102_sie_xsd.xsd" containing the schema definitions for eSig Directive Extensions (Annex L). For the purpose of integrity checking, the hash values of this file are:

   MD-5: daef22d4341f62c3fa967dba2603e14a

   SHA-1: 99c67ae9de9d9211ae10008f7a9714c79a420b9f

   SHA-256: d73722f0781047471eeef7c40edf81ac509d356f34d720ffd26ab08cf294a1a3

3. "ts_102231v030102_additionaltypes_xsd.xsd" containing the schema definitions for additional types. For the purpose of integrity checking, the hash values of this file are:

   MD-5: 600bae6d4ffd85c7a8b6d4ce0cdfadd5

   SHA-1: c3efca97d2b0d3393907b1e16bc86f8d94399dfa

   SHA-256: ab7406480a790076dd91d88b717fad8e596273ce2c83eb22757e1a9dc9e1aaf6

C.4 LDAP schema

This XML schema is held in file "ts_102231v030102_sch.schema". For the purpose of integrity checking, the hash values of this file are:

MD-5: 7c1db12c4a67ccfb0ba7cc7b2aa7c694

SHA-1: 9735ba488bf7bedc9e59cb5f6233367d734009f5

SHA-256: 079291f1fbf3442cc519af0bfc3708686a7dbc76233c429871e87c90d1aee5a1

Annex D (normative): Registered Uniform Resource Identifiers

This annex specifies those Uniform Resource Identifiers (URIs) which have been registered in connection with the present document. Those with the radix (base) "http://uri.etsi.org/02231/……" are registered and declared by their presence in the present document, for specific usage within the present document: those with the radix "http://uri.etsi.org/TrstSvc/……" are registered by ETSI as a Common Domain (see http://portal.etsi.org/pnns/xml.asp#Common_Domain) on behalf of the TC ESI because they have a wider applicability and usage and are listed here for the convenience of users of the present document.

Where URIs registered on behalf of the TC ESI are used within the specifications of TSL fields (see clause 5) it is generally the case that users can register other URIs for their own purposes and extend the range of that field, although it is strongly RECOMMENDED that the scheme operator makes a clear declaration of the meaning of that URI. Refer to clause 5.2 and onwards.

In the following tables the following layout is used for each URI declaration:

The URI is given as an unbroken string	Related TSL field (if any)
The meaning of the URI is given, indented to emphasize its relationship to the preceding URI.	

Where more than one URI relates to a specific TSL field the second column will extend across all URI declarations (row-pairs) which apply.

D.1 URIs registered within the present document

The following URIs are hereby declared and registered under the present document's assigned radix:

http://uri.etsi.org/02231/v3.1.2	N/a
This issue of TS 102 231 and its related parts.	

http://uri.etsi.org/02231/TSLTag	TSL tag
A data structure which conforms to the TSL specification published in TS 102 231, in any of its historical issues or this one.	

http://uri.etsi.org/02231/v2#	N/a
The XML namespace identifier relating to the TSL version specified in this issue of TS 102 231.	

http://uri.etsi.org/02231/TDPContainer	N/a
A qualifier for web pages that contain one or more TDPs which can be used as a value of the attribute "profile" for the "head" element of the web page.	

D.2 ETSI Common Domain URIs

The following URIs have been declared and registered by ETSI under the Technical Committee Electronic Signatures Infrastructure's (TC ESI) assigned radix:

http://uri.etsi.org/TrstSvc/TSLType/generic	TSL type
Indicates a "generic" TSL that exclusively contains trust services which are approved or recognized by the scheme operator owning the TSL through a process of direct oversight (whether voluntary or regulatory).	
http://uri.etsi.org/TrstSvc/TSLType/schemes	
Indicates a "schemes" TSL that exclusively contains TSL Issuers, independently responsible for the approval or recognition by a community of trust services through a process of direct oversight (whether voluntary or regulatory).	

http://uri.etsi.org/TrstSvc/TSLType/StatusDetn/active	Status determination approach (see note)
Services listed have their status determined after assessment by or on behalf of the scheme operator against the scheme's criteria (active approval/recognition).	
http://uri.etsi.org/TrstSvc/TSLType/StatusDetn/passive	
Services listed have been nominated by their provider or are known to be operating in the marketplace, but have not undergone assessment by or on behalf of the scheme operator for initial approval (passive approval/recognition).	
http://uri.etsi.org/TrstSvc/TSLType/StatusDetn/delinquent	
Services listed have been deemed to be non-compliant with scheme criteria.	
http://uri.etsi.org/TrstSvc/TSLType/StatusDetn/list	
No predetermined criteria. The TSL is just a collection of pointers to other TSLs. The issuer will not necessarily take any responsibility or even liability for the content of TSLs pointed to.	
NOTE: In the case of meanings "active" and "passive", a scheme could include in the TSL both services and schemes whose current status is approved/ recognized (either actively or passively, but each indicating a positive assertion) and those which have failed to meet the criteria. In the case of meaning "delinquent", the TSL would list only those services which had explicitly failed to fulfil the criteria of the scheme (i.e. had exhibited delinquency). It is therefore unlikely that such a status determination approach would include other schemes, although this could be determined by the scheme operator's rules.	

http://uri.etsi.org/TrstSvc/Svctype/CA/PKC	Service type identifier
A Certification authority issuing public key certificates.	
http://uri.etsi.org/TrstSvc/Svctype/CA/QC	
A Certification authority issuing Qualified Certificates.	
http://uri.etsi.org/TrstSvc/Svctype/TSA	
A Time stamping authority.	
http://uri.etsi.org/TrstSvc/Svctype/Certstatus/OCSP	
A Certificate status provider operating an OCSP-server.	
http://uri.etsi.org/TrstSvc/Svctype/Certstatus/CRL	
A Certificate status provider operating a CRL.	
http://uri.etsi.org/TrstSvc/Svctype/RA	
A Registration authority.	
http://uri.etsi.org/TrstSvc/Svctype/IdV	
An Identity verification service.	
http://uri.etsi.org/TrstSvc/Svctype/CGen	
A Certificate generation service which responds to requests for certificate generation from an authenticated source of identity information.	
http://uri.etsi.org/TrstSvc/Svctype/ACA	
An Attribute certification authority.	
http://uri.etsi.org/TrstSvc/Svctype/Archiv	
An Archival service.	
http://uri.etsi.org/TrstSvc/Svctype/REM	
A Registered Electronic Mail service	
http://uri.etsi.org/TrstSvc/Svctype/KEscrow	
A Key escrow service.	
http://uri.etsi.org/TrstSvc/Svctype/PPwd	
Issuer of PIN- or password-based identity credentials.	
http://uri.etsi.org/02231/Svctype/SignaturePolicyAuthority	
Service responsible for issuing, publishing or maintenance of signature policies	
http://uri.etsi.org/TrstSvc/Svctype/supervision	
An assessment scheme which is a system of supervision as defined in, and which complies with all applicable requirements of Directive 1999/93/EC [1].	
http://uri.etsi.org/TrstSvc/Svctype/voluntary	
An assessment scheme which is a voluntary approval [accreditation] scheme as defined in, and which complies with all applicable requirements of Directive 1999/93/EC [1].	
http://uri.etsi.org/TrstSvd/Svctype/TSLIssuer	
An issuer of TSLs.	
http://uri.etsi.org/TrstSvc/Svctype/unspecified	
A trust service of an unspecified type.	

http://uri.etsi.org/TrstSvc/Svcstatus/inaccord	Service current status
The subject service is in accordance with the scheme's specific status determination criteria (*only for use in positive approval schemes*).	
http://uri.etsi.org/TrstSvc/Svcstatus/expired	
The subject service is no longer overseen by the scheme, e.g. due to non-renewal or withdrawal by the TSP, or cessation of the service **or** the scheme's operations.	
http://uri.etsi.org/TrstSvc/Svcstatus/suspended	
The subject service's status is temporarily uncertain whilst checks are made by the scheme operator (typically e.g. while a revocation request is being investigated or if action is required to resolve a deficiency in the service fulfilling the scheme's criteria.	
http://uri.etsi.org/TrstSvc/Svcstatus/revoked	
The subject service's approved status has been revoked because it is no longer in accordance with the scheme's specific status determination criteria (*only for use in positive approval schemes*).	
http://uri.etsi.org/TrstSvc/Svcstatus/notinaccord	
The subject service is ***not*** in accordance with the scheme's specific status determination criteria (*only for use in negative approval schemes*).	

http://uri.etsi.org/TrstSvc/schemerules/Dir-1999-93-EC/supervision An assessment scheme which is a system of supervision as defined in, and which complies with all applicable requirements of Directive 1999/93/EC [1].	Scheme type/community/rules (at the primary level)
http://uri.etsi.org/TrstSvc/schemerules/Dir-1999-93-EC/volapproval An assessment scheme which is a voluntary approval [accreditation] scheme as defined in, and which complies with all applicable requirements of Directive 1999/93/EC [1].	

D.3 Registering additional URIs

Any organization operating a scheme might choose to create its own URIs for its own specific purposes or request ETSI to assign a registered URI root under the ETSI Identified Organization Domain, and then define its own URIs under this root. It might be appropriate to register certain of those URIs where they complement URIs required by or which might be used in the context of the publication of a TSL. The following examples suggest how additional URIs could be created, including showing a second level of rules, after using the applicable Optional URI as shown above:

Potential URI **Meaning**	**Related TSL field (if any)**
http://uri.etsi.org/"registered_org"/"schemename" This could mean an assessment scheme called "schemename" being operated by "registered_org", where "registered_org" is replaced by the name of the scheme operator and "schemename" is replaced by the actual scheme name	
http://"scheme_op_URI_root"/.../schemerules/ "schemename" This URI would be registered under a different root, e.g. the scheme operator's, distinguished by "scheme_op_URI_root", or it could be another organization which maintains a registry of URIs. This URI could mean an assessment scheme called "schemename" being operated by "scheme_op" where "scheme_op" is replaced by the name of the scheme operator and "schemename" is replaced by the actual scheme name.	Scheme type/community/rules (at the secondary level)

Annex E (normative): Implementation notes for multilingual support

E.1 Multilingual character string

The string contained within a multilingual character string SHALL fulfil the requirements of annex N of ISO 10646 [23] subject to the following restrictions:

1) the content SHALL be a string of characters from the Universal Character Set (UCS) as defined by ISO 10646 [23];

2) the content MUST be UTF-8 encoded;

3) the content MUST NOT include any signature to identity the UCS (see annex H of ISO 10646 [23]);

4) control functions (ISO/IEC 6429 [39]), escape sequences (ISO/IEC 2022[40]) and control sequences or strings MUST NOT be used; therefore control characters such as TAB, CR, LF MUST NOT be present;

5) private-use characters (see clause 10 of ISO 10646 [23]) from the private use zone (code points E000 to F8FF) in the Basic Multilingual Plane (BMP) and from the private-use Planes 0F and 10 in Group 00, SHALL NOT be used;

6) Tag Characters (see annex T of ISO 10646 [23]) MUST NOT to be used: therefore the characters from the TAGS (3001) collection MUST not be used (see annex A of ISO 10646 [23] for the list of defined collections);

7) the content SHALL be plain text without any mark-up elements or tags from languages as SGML, HTML, XML, XHTML, RTF, TeX and others;

8) it is RECOMMENDED that the content follows the semantic rules defined by UNICODE version 4.00 for the corresponding characters;

9) combining characters SHOULD NOT be used if the content can be expressed without them; if there is the need to use combining characters but it is possible not to use the ones listed in clause B.1 of ISO 10646 [23], then that latter set MUST NOT be used (this helps to keep as low as possible the required implementation level (as defined by clause 14 of ISO 10646 [23]) for parsing applications.

E.2 Multilingual pointer

If the content pointed by the multilingual pointer is plain text, it SHALL meet the following requirements that express the conformity to ISO 10646 [23] according to the annex N of ISO 10646 [23] and add further restrictions:

1) the pointed content SHALL be a string of characters from the Universal Character Set (UCS) as defined by ISO 10646 [23];

2) the pointed-to content MUST be UTF-8 encoded;

3) the pointed-to content MAY include the signature for UTF-8 (see annex H of ISO 10646 [23]) to identify the UCS;

4) control functions (ISO/IEC 6429 [39]), escape sequences (ISO/IEC 2022 [40]) and control sequences or strings MAY be used;

5) private-use characters (see clause 10 of ISO 10646 [23]) from the private use zone (code points E000 to F8FF) in the Basic Multilingual Plane (BMP) and from the private-use Planes 0F and 10 in Group 00, SHALL NOT be used;

6) Tag Characters (see annex T of ISO 10646 [23]) MUST NOT to be used: therefore the characters from the TAGS (3001) collection MUST not be used (see annex A of ISO 10646 [23] for the list of defined collections);

7) if the pointed-to content is expressed by means of mark-up languages as SGML, HTML, XML, XHTML then:

   a) the requirements described in W3C Technical Report #20 [33] are RECOMMENDED;

   b) a language indication MAY be present according to the mechanisms listed in W3C Technical Report #20 [33].

8) it is RECOMMENDED that the pointed-to content follows the semantic rules defined by UNICODE version 4.00 for the corresponding characters;

9) combining characters SHOULD NOT be used if the pointed-to content can be expressed without them; if there is the need to use combining characters but it is possible not to use the ones listed in clause B.1 of ISO 10646 [23], then that latter set MUST NOT be used (this helps to keep as low as possible the required implementation level (as defined by clause 14 of ISO 10646 [23]) for parsing applications).

E.3 Overall requirements

For the XML implementation of a TSL, it is RECOMMENDED that the requirements of W3C Technical Report #20 [33] be met.

For interoperability purposes, all applications parsing TSLs MUST be able to store and manage all characters defined by ISO 10646 [23]. This way the digital signature applied to the TSL can be always verified, whatever UCS characters are used within the TSL. However the parsing application may not be able to correctly present all characters.

NOTE: Developers of TSL parsing applications are advised that if their application does not support some of these characters, the application SHOULD give notice to the user about possible incorrect representation of the content of multilingual fields; the precise behaviour of the application while presenting unsupported characters is left to developers.

Annex F (informative): TSL Signing considerations

Although this annex is informative implementers are strongly recommended to satisfy the guidance which it provides, if not immediately, then as soon as suitable applications are available.

F.1 Signing application maturity

The present document requires that, when signing a TSL, the signer's certificate is bound into the signature. The most reasonable means to accomplish this is by using the SigningCertificate signed attribute (or property) available in TS 101 733 [2], TS 101 903 [35] and TS 102 778-3 [46] signatures.

F.2 CMS/ESS and CAdES

The present document supports two options to accomplish binding the certificate into the signature:

1) Basic CMS signatures with the addition of an ESS feature.

   For CMS-Signatures RFC 5652 [37] (see clause A.6), using the SigningCertificate signed attribute defined in RFC 2634 [13] fulfils the requirement of signing the signing identifier together with the TSL. This attribute is one of the two possible options for the implementation of this requirement for a TS 101 733 [2] signature; a CMS signature that contains this attribute with the profile specified in clause A.6.2.1 is also a -signature compliant with TS 101 733 [2] (a CAdES-BES).

2) TS 101 733 [2] signatures that are CMS signatures using advanced security features.

Applications supporting TSLs are RECOMMENDED to implement option 2 in contexts where is known that all parties use CAdES compliant applications.

Instead, in contexts where none or few parsing applications compliant with TS 101 733 [2] are used, it is recommended to generate only basic signatures compliant with CMS and ESS (i.e. option 1). Since these basic signatures are also compliant with TS 101 733 [2], applications supporting TS 101 733 [2] would be able to completely parse and verify these "basic signatures".

In the case of contexts where applications compliant to both basic CMS/ESS signatures and TS 101 733 [2] are used, if a TSL is signed by using the advanced features provided by TS 101 733 [2], the implementations that support only CMS/ESS but not the advanced features of TS 101 733 [2] will be still able to verify the TS 101 733 [2] signature calculated over the TSL and the TS 101 733 [2] signed attributes, but probably they would not be able to understand any of the attributes present other than those supported by CMS/ESS. Therefore the CMS/ESS implementation will not be able to exploit/check the advanced security services provided by TS 101 733 [2], but the possibility to use the basic service (i.e. verify the signature over the TSL) will be always retained.

F.3 XML

Using XML, applications not supporting TS 101 903 [35] are advised to put the signing certificate into the KeyInfo element and add a reference to this into the signature. This is the standard XML-Signature [34] way to have an element included within the signature. Such applications are encouraged to ensure they will not refuse a TSL whose TS 101 903 [35] signature contains elements unknown to the application.

If an implementation supports TS 101 903 [35] signatures, it is recommended that the xades:SigningCertificate element is included in xades:SignedSignatureProperties. Adding the reference to ds:KeyInfo is not necessary and in fact is discouraged, although, as acknowledged in annex B, ds:KeyInfo itself may be present. Such implementations should be flexible enough to accept TSLs signed without TS 101 903 [35].

If an implementation supports TS 101 903 [35] signatures, it is recommended that the SigningCertificate element is included in SignedSignatureProperties. Adding the reference to KeyInfo is not necessary and in fact is discouraged. Such implementations should be flexible enough to accept TSLs signed without TS 101 903 [35].

F.4 PDF/A

Using PDF/A applications should implement TS 102 778-3 [46]. Being this standard based on a TS 101 733 [2] signature, the same implementation notes related to TS 101 733 apply.

Annex G (informative): Management and Policy considerations

The TSL is a mechanism which is supporting of electronic transactions but not essential for them. There remains a variety of different models on which schemes may operate and a variance in how information from TSLs may be interpreted. Because of this lesser degree of dependence upon the TSL, the need to keep up to date information within a TSL is less urgent than that for, e.g. a CRL.

Scheme operators should publish their specific criteria for the provision of revisions to TSL information. These revisions will fall into the following categories.

G.1 Change of scheme administrative information

This category includes any changes to information concerning the scheme and which is embedded within the TSL. Such changes could include, *inter alia*, change of scheme addresses, revisions to acceptance criteria, scheme policy. When these change the TSL should be re-issued.

If there are material changes to information directly referenced through the TSL but the reference itself does not change then there will be no need to amend the TSL.

Any changes in this category should not affect the status information concerning any trust services mentioned within the TSL.

If the changes were the result of a change of ownership of the entity operating the scheme then the scheme could continue to operate without change or the scheme could cease operations and re-establish itself as a new scheme. It would be for the operators to determine how they wanted to handle this and how they would deal with the handling of services recognized under the scheme.

G.2 Trust-service identification

Whenever a scheme operator adds trust service to a TSL, it is important to users of the TSL to be able to unambiguously identify that service's status definition. While name and address may be highly relevant and therefore very important, the digital identity-field is the only option that can provide secure identification of the trust service and tokens which it supplies. The service digital identity-field does not, however, prescribe a specific format for this identifier, since the TSL is intended to be applicable to services based on technologies other than PKI.

For PKI-applications, applications also have choices as to how to present the digital identifier. For creating or parsing TSLs, applications should support three formats for the service digital identity:

1) one of the two methods defined in clause 4.2.1.2 of RFC 5280 [17], on how to calculate subject key identifiers for CA certificates;

2) X.509-certificates;

3) Public key.

G.3 Change of trust-service status

These changes are those directly affecting the inclusion, exclusion or reported status of any trust service within the TSL (and possibly also information concerning their provider) and whether the information is current or historical (e.g. the introduction of a new TSP and service; the revocation of a service).

When any such change occurs the TSL should be re-issued with the previous current status becoming the most recent historical status and current status being amended to reflect the situation.

Where a service changes its "Service digital identity" (see clause 6.4.3), e.g. as a result of a take-over or a re-branding or a renewal of associated digital data for security reasons, the situation should be handled effectively as if the service using the old identity had ceased to operate and the service using the new identity had come into being.

The service which is effectively stopping should have its "Service current status" (see clause 6.4.4) revised to meaning 2 (ceased operations) and the previous status information placed into the "History information" (see clause 6.5) of the TSL. This should then be retained for the published retention period (since there may be requirements to check on services rendered during it period of activity - no ceased service's "Historical information" should be discarded.

The service under the new digital identity should be given its own new entry, which at this initial stage would have no "History information" which required recording.

G.4 Amendment response times

Changes to any TSL information should be provided in a timely fashion, which as a minimum should be the following (the response times taking account of the format of the information's presentation):

a) Within four working hours of a decision to implement a change in status.

b) Where each TSL revision is disseminated electronically to those parties who are obliged by the scheme operator to maintain copy of the TSL for their own clients, a four working hour response should be met. Such parties would typically be TSPs whose services are listed in the TSL, and should themselves undertake to post the revised TSL within the same response criteria.

G.5 On-going verification of authenticity

The frequency at which information within a TSL will change is likely to be low. This could give a determined hacker sufficient time to replicate and replace all instances of a TSL, *IF* they were able to replace all examples of the TSL itself and a surrogate PKC for the TSL operator. This should be protected against by the scheme operator itself making frequent verification of its own TSL and all authorized and recognized replications of it. In addition, the regular re-issuing of the TSL, even when there is no change to any statuses within it, will also ensure that, at the least, the signature value changes periodically. This clause has already discussed some security measures which would reduce significantly the likelihood of this being achievable.

G.6 Upon a scheme's cessation of operations

Owing to the dependence which users may place upon the TSL, schemes which operate a TSL should have in place appropriate mechanisms for any cessation of their operations, be it temporary or permanent. The normative parts of the present document provide for a "Next update" date and time. This field makes explicit provision for a scheme to indicate that it is no longer functioning, by setting this field to null.

Notwithstanding that technical provision which allows a final TSL to be published "in perpetuity", scheme operators need also to consider additional actions to ensure a controlled cessation of their operations. As a minimum, the scheme should revoke the keys used for signing and verification of its TSL and make a public announcement of its cessation of operations, indicating (if known) whether this is temporary or permanent.

If time permits and circumstances warrant, a new TSL should be issued (ref. Next update) which relegates all status records to the history components as of a specific date after which the scheme no longer accepted responsibility for status determination and produces an archive for long-term reference. In addition to the specific provisions of the "Next update" field discussed above, it is required by the normative part of the present document that in such a circumstance the field "Service current status" is set to indicate "Expired".

Whilst the issues of the long-term validity of this archived TSL may be something for consideration it is beyond the scope of the present document to deal with them in depth. Suffice to say that, where there is a decision or obligation to hold available the final TSL status for an extended period, appropriate measures (already widely known and discussed in this field) should be taken to protect signatures against the decay of the strength of crypto algorithms.

G.7 User reference to TSL

When and how often a user/relying party should reference to a TSL for status information is not an issue within the scope of the present document. Such a decision lies with the user and should be a determination made according to a variety of factors reflecting their own circumstances, *inter alia*, the degree of reliance they place in a TSL status indication, how often they deal with the other party, the nature of the business relationship and the value of the business or the transaction in question. These are factors only they can determine after conducting their own risk analysis. They may have such infrequent recourse to a TSL that they will always check for any TSL records of status.

Scheme operator's could assist in this by offering additional services to notify when a new TSL is issued, or to guarantee frequent re-issue of a TSL at a frequency which may mean numerous re-issue without change of any services' status. However, the mechanisms proposed for having multiple copies of TSLs existing contemporaneously are designed to cater for the low rate of information change already discussed, and these may not be suitable for frequent TSL re-issue.

G.8 Reliance upon hard-copy TSL information

Whilst it is a requirement that scheme operators make available information which is "human-readable in printable, hard-copy form" there is no requirement, nor expectation, that hard copy should be provided in a manner which can be authenticated by any printable means. Users should expect that authenticated information presented on-screen by an application accessing a TSL will faithfully reproduce that information when it is printed and should take the trouble to cross-check the information with that on-screen where they have any doubts.

Scheme operators might choose to make paper copy available by surface post if that seems desirable.

G.9 TSL size

The present document provides a number of fields in which the scheme operator may choose to provide actual natural language text in preference to a URI or other reference to a source of information. Clearly the inclusion of large quantities of text will have a direct influence of down-load and parsing times, this especially so if e.g. it relates to the descriptions of services, and the scheme has a large number of trust services listed. It is therefore recommended that implementers take advantage of the opportunity to use URIs and limit embedded text as much as is reasonably, accounting for the overall size of the TSL and the available bandwidth and storage capacities of the typical user of their TSL. Referencing other documents also allows advantage to be taken of more sophisticated presentation options which formats such as PDF and other formats enable.

Annex H (informative): Locating and Authenticating a TSL

H.1 Introduction

This annex offers guidance on how to locate and authenticate TSLs. It does not try to cover all possible scenarios, but focuses on those that are likely to occur. It is based on the following assumptions:

- A relying party intends to authenticate a trust service token (TrST, e.g. a certificate) that has been received from some counter-party (see note).

NOTE: Whilst the relying party may have the desire to authenticate the TrST, the TSL cannot generally be relied upon to provide more than a secondary source of trust. In some circumstances it may be possible to derive from the TrST, information which provides a digital identity for its issuer, and that issuer may then be located within a TSL, there are many assumptions about trust which have to be satisfied before a true authentication can be claimed by this process. One should therefore expect that, in general, further steps need be taken to authenticate the TrST.

- The relying party has at least reasons to assume there exists a scheme which the TrST-issuing trust service is part of.
- The relying party has at least reasons to assume the scheme is using a TSL for publishing the status of the services overseen by that scheme.

No further assumptions are made. It may be straightforward to retrieve the TSL or the relying party has to do a thorough search on the internet. Trusting the TSL-issuer is a question of policy and not dealt with at all.

Although this annex is written very much in terms of the relying party searching for and within a TSL which lists general trust services, the principles described may apply equally to the location and authentication of TSLs which list other assessment schemes (i.e. "Schemes" TSLs).

H.2 Locating a TSL

Locating a TSL can either be easy, if the trust service token provides a direct link or any other hint on where the TSL can be retrieved from. If no such information is available. The relying party may use certain strategies to find a suitable TSL. Both models are discussed in the clauses that follow.

H.2.1 TSL location models

We can consider three models by which TSL location information can be provided. They are: Bound, Linked, and De-coupled. Each is explained and their comparative merits considered below.

H.2.1.1 Bound information

In this model, information about a TSL (or possibly more than one) is intimately bound into the TrST. In other words, the TSP advertises the fact that its service fulfils the criteria of the indicated scheme. The user initiating the communication (i.e. the sender) need not be aware of the inclusion of this information.

Such a solution is easy in terms of the need to locate a TSL - the work is done - but it is "dirty" in that it renders the token a victim of the continued fulfilment of the scheme's criteria, and indeed the stability of the scheme itself. In the event that the status of the trust service changes, or the scheme's PKC itself is revoked, or the scheme substantially changes its criteria, or even ceases to exist in its recognized state, the TrST would most probably need to be revoked. This has the implication that a TSP issuing large volumes of tokens would have to revoke and re-issue them in the case of any of these failures originating largely outside its control (of course it may well be that this change in its status is the result of some action (or inaction) on the part of the TSP itself).

In the case of "black list" principle TSLs, it is manifestly unlikely that a TSP will bind in information of a negative nature, and so here the Bound model most probably does not apply. By the same token, even schemes applying positive criteria may find TSPs unwilling to bind in a pointer to information which may put them in a bad light if, for example, they have suffered a degradation in their approval status.

The bound model therefore suffers from its sensitivity to changes from a number of other sources and from circumstances where the TSP may feel jeopardized by inclusion of a reference to its present status. Nevertheless, if used this model obviates the need to search for a TSL (although there may be other TSLs not referenced which might have useful information about the trust service).

The TSL Distribution Point (see clause 6.3) is one of the prime mechanisms to locate a TSL relevant for validating a TrST. This mechanism may be used in all three models.

H.2.1.2 Linked information

In this model, information about any relevant TSL(s) is included within the transaction but not in a way which binds it intimately to the service token. The TSL location could be included by an application, possibly configured by either the user or their service provider; the user may not need to know about it, but transparency may not always be as clear as with the Bound model. The Linked model has the obvious advantage that status information is provided separately from the TrST and hence could change without having any impact on the TrST (although according to the nature of the scheme, this may not always be so).

Most of the arguments about the willingness of TSPs to include this information apply as they do to the Bound model. However, it is clearly less sensitive to status changes and also makes it unnecessary to search for TSL information, with the same caveat that there may be other TSLs not referenced which might have useful information about the trust service.

H.2.1.3 De-coupled information

In the De-coupled model there is no TSL location information provided with the transaction - it is up to the relying party to find it herself. This has the distinct advantage of there being no dependency on the TSP to provide the information, no need for the sender to have any knowledge of this information either.

This model carries a potential penalty: the relying party's system has to search for the TSL, and the search may have no initial clues as to where to look.

H.2.2 Searching for a TSL

It becomes necessary to search for a TSL particularly in the case of the De-coupled model, but it may also be necessary where the information provided through the Bound and Linked cases is inadequate for some reason. Note that a search may also be appropriate simply when an interested party seeks information about a particular TSP and/or its services but does not know where to find an associated TSL.

Searching can be broken down into four potential stages which can be regarded as offering decremental ease of searching. These are described below, starting with the simplest.

H.2.2.1 Same-scheme searching

In this case the relying party is able to use the TSL belonging to any scheme(s) within which fall any trust services with whom she herself has a relationship (and presumably, therefore, in which he has some assurance) - we will use the term "relying-party's scheme/TSL" as a convenience, although strictly speaking there is no direct relationship between the relying party as a subscriber to a service and any scheme under which that service operates. Such an approach would work where the counter-party's trust service is overseen by the same, or one of the, relying party's schemes. Each of the TSLs associated with those schemes could be searched for the presence of status information relating to the counter-party's trust service.

H.2.2.2 Known scheme searching

In this case there are three possible options, each dependent upon the relying party being a subscriber to at least one trust service which is within a TSL-issuing scheme, i.e. that there is a "relying-party scheme" as explained above. These options may exist in any combination.

In the first case, if the relying-party's scheme operates under, or within a federation or community of schemes all supervised by, a Root Key Authority (RKA) then it may be possible to derive from that RKA the location of other schemes which provide TSLs and which could be assumed to have the same degree of assurance as the relying-party's scheme.

In the second case, the relying-party's TSL could contain within it a pointer or pointers to other TSLs (see clause 6.2.12) which the relying-party's scheme operator feels worthy of some degree of recognition, or the scheme operator may publish a "Schemes" TSL to which the relying party could refer. (see TSL type). How one scheme operator determines that another TSL is sufficiently reliable to merit inclusion in their own is not defined by the present document. The scheme operator would be expected to make publicly accessible their policy for doing so, whether by using "Pointers" to other TSLs' or by publishing a "Schemes" TSL.

In the third case, the relying party may have built up their own list of TSLs or have access to an alternative "Schemes" TSL which they regard as reliable and could search any of those.

Thus by any combination of the above options, the relying party could have identified TSLs within which they could search for the presence of status information relating to the counter-party's TSP.

If none of the options in this and the preceding part are successful, then a "blind" search may be conducted, as described below.

H.2.2.3 "Blind" (unknown) scheme searching

If a relying party has absolutely no information about a scheme issuing TSLs relevant for authenticating a TrST, maybe even no information that such a scheme or a TSL exist, the fallback-strategy described in this clause may be successful.

The concept follows the model human users would apply in similar cases: they would use any internet-search engine. TSLs compliant with the present document will use the TSL tag value specified for that field. Thus, finding that tag value in the appropriate field of a data structure should identify it as a TSL. Further qualification and confidence can be drawn by parsing and matching other fields, such as the issuer distinguished name. If the issuers of the TSL follow the recommendations given in the present document, we expect the results of any web search to provide a direct link to a TSL in most cases. This expectation may be thwarted though by sort-of denial of service attacks, e.g. by publishing fake pages that would also show up as hits, but indeed lead to junk information only. It is considered unlikely that such attacks will be interesting enough to execute.

To be able to find a TSL using a search engine, the following assumptions and requirements are relevant:

- A TSL is unlikely to be found directly, so long as search engines do not index unspecific XML or DER-encoded data - at the time of publication of the present document only HTML, PDF and similar formats are indexed. To enable search-engines to find a TSL, an HTML-page is needed that contains a) a searchable string and b) a link to the TSL. By specifying a simple structure for such a page, and simple criteria to make that page "findable", applications will have a straightforward way to locate the TSL.

When a TSL is located by any of these means, any further parsing depends on which type of TSL it is (TSL type).

H.2.2.3.1 Structure of the HTML-Page.

A scheme issuing a TSL is RECOMMENDED to publish a web-page defined by using either:

a) HTML 4.01 [31] or XHTML 1.0 [29] with **strict** DTD; or

b) XHTML 1.1 [30].

Later versions of XHTML MAY be used as and when they become available and widely accepted. The web page should be compliant with the following structure.

HTML version information.

It is RECOMMENDED to use the following declarations:

for HTML 4.01:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
```

for XHTML 1.0:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

for XHTML 1.1:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

for future versions of XHTML the declaration should be taken from their specifications.

A document head consisting of:

- The HEAD element using the profile-URI http://uri.etsi.org/02231/TDPContainer which clearly identifies that HTML-document as being a TSL-container.
- A TITLE element with the content "*Trust-service Status List Distribution Points Container*".
- A META element with the name "*contains*" and the content "*XML*" resp. "*DER*" or "*XML,DER*" if the page contains the XML resp. the DER version of the TSL, or both.
- Other META element, such as the element with the name *keywords*, are also possible.

```
<head profile="http://uri.etsi.org/02231/TDPContainer">
<title>Trust-service Status List Distribution Points Container</title>
<meta name="contains" content="XML,DER">
<meta name="keywords" content="TSL,Trust Status List,TDP">
</head>
```

The body-section contains a paragraph with the string suitable for searching this page, followed by several paragraphs, each of which contains exactly one anchor (A) element. The href attribute contains a URI pointing to a TSL. The content of the element starts with the string TSLLink and specify the type of TSL pointed to by adding XML or DER to the string. This is followed by a colon and the name of the scheme to which the TSL relates. This name should be exactly the same as the field Scheme name. If this field contains names in multiple languages, one, some or all of those names can be selected.

```
<body>
<p>This page contains links to objects of type http://uri.etsi.org/02231/TSLTag; the CMS
EncapsulatedContentInfo is identified by the 0.4.0.2231.1.0 / itu-t(0) identified-organization(4)
etsi(0) tsl-specification (2231) identifiers (1) tsl-info(0); the XML object is identified by
(http://uri.etsi.org/02231/v2#, TrustServiceStatusList)</p>
<p>
<a href="URI">TSLLink+[XML|DER]:SchemeName</a>
</p>
</body>
</html>
```

H.2.2.3.2 Example

The following example provides links to two formats of a TSL from the scheme "SomeScheme":

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head profile="http://uri.etsi.org/02231/TDPContainer">
<meta name="contains" content="XML,DER">
<meta name="keywords" content="TSL,Trust Status List,TDP">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Trust-service Status List Distribution Points </title>
```

```
</head>

<body>
<p>This page contains links to objects of type http://uri.etsi.org/02231/TSLTag; the CMS
EncapsulatedContentInfo is identified by the 0.4.0.2231.1.0 / itu-t(0) identified-organization(4)
etsi(0) tsl-specification (2231) identifiers (1) tsl-info(0); the XML object is identified by
(http://uri.etsi.org/02231/v2#, TrustServiceStatusList)</p>
<p>
<a href="http://somescheme.org/tsl/xml/current">TSLLink+XML:SomeScheme</a>
</p>
<p>
<a href="http://somescheme.org/tsl/xml/current">TSLLink+DER:SomeScheme</a>
</p>
</body>
</html>
```

H.3 Authenticating a TSL

It is assumed that each scheme provides its users with the means to authenticate the TSLs it publishes, which may be performed by a TSL:

1) Ensure that the validity period of the TSL has not expired (see clause 5.3.15).

Starting with the scheme operator digital identity reference found within the TSL, retrieve the public key to be used to verify the signaturenumber of different mechanisms. This, therefore, is implementation specific, and it is recommended that scheme operators specify in their policy how to authenticate their TSLs, or provide users with the means to authenticate them. For example, a scheme could:

2) provide a trusted channel (e.g. TLS) to download the TSL from a secured site;

3) publish in a reliable source (e.g. an official bulletin) the digest of the scheme's public key or public key certificate corresponding to the private key used to sign the TSL.

For TSLs located after a "blind search" the means applicable to the authentication of such TSLs may not be immediately apparent to the relying party, and may require human intervention to make it possible.

The continued validity of the TSL should also be verified, by ensuring that the validity period of the TSL has not expired (see clause 5.3.15).

If either of these checks fails, the TSL authentication should be considered to have failed.

NOTE: The decision to trust an authenticated TSL is covered in clause H.4.

H.4 Trusting a TSL

A TSL is a signed electronic document. To verify the signature, relying parties need to be able to access the applicable public key. Since the scheme issuing the TSL is effectively positioned "above" the TSPs approved by that scheme, the authenticity of the public key cannot be verified solely on the basis of its certificationby any TSP inside or outside the scheme. Providing the scheme's public key is therefore a problem very similar to providing the public key of a CA service and any details are out of scope for the present document. Nevertheless, self-signed keys established by well-known entities may prove to be a suitable solution. It is imperative that the key used for signing the TSL has a public-key certificate published and made available to relying parties and users (e.g. publication in official journal).

Widespread replication of a TSL may also be constructive in reducing traffic volumes accessing a single source, where the TSL is large.

After successful authentication of the TSL, the relying party needs to decide if it can trust the TSL. The process to be followed by any user that wants to use a TSL is very similar to the steps that need to be taken when deciding about trust in a certification authority. If public key certificates are used in this process, the relying parties' software should be able to distinguish between certificates trusted for issuing certificates and certificates trusted for issuing TSLs.

Having identified, located and authenticated a TSL, the user could then carry out any further steps to establish trust in the scheme/TSL as required by their own policy. Consequently the user decides whether or not to trust the scheme and the TSLs it operates, and the extent of that trust. Only if these further checks are positive is the information within the TSL relied upon.

The user can then take steps to ensure that on future searches this TSL is automatically accepted as being reliable. A typical procedure might therefore look like the following:

1) user imports the TSL's public key certificate or public key into the software;

2) user sets the status of the imported certificate or public key to something like "*trusted for issuing TSLs"*;

3) user subsequently uses the certificate or public key to verify TSLs maintained by the specified scheme.

It is assumed that the user is able to establish for themselves sufficient trust in the certificate or public key in question by verifying themselves a publicized hash of the certificate or the public key itself, available from some reputable source, e.g. published in an official journal.

The procedure described above can be performed by each user, but will in many cases be carried out on the level of an organization according to their own policy. In this case, the software environment of each user's machine would typically be pre-configured by the system administration or by the security officer. In time it is likely and certainly possible that such certificates or public keys could also be pre-installed in browsers, so enabling personal users to gain advantage from this approach.

In the case of compromise of the scheme's private key, the operator's policy should require informing the user in the same manner as in the case of a key compromise of a TSP's self-certified key. Such key compromise will get broad attention, since there will only be a limited number of schemes operational, they will be widely known, and furthermore their certificates (and therefore notification of their certificates' revocation) will be widely available, ensuring that such events will not remain unnoticed.

A scheme operator may also provide mechanisms compatible with the standard way of handling revocation information: add a CRL distribution point extension into the self-signed certificate and provide a CRL at that point. A compliant client implementation could then also automatically check that CRL to detect any revocation.

H.5 Replicating TSLs

TSLs will be relatively few in number, with only moderate numbers of service statuses described within them and furthermore, since it is unlikely that services will come and go with great rapidity (in terms of internet-speed), they will have a low frequency of information change. For this reason, low-complexity approaches to the publication of TSLs and to control over their authenticity are adopted. A scheme either can build upon the safety in numbers concept (i.e. multiple copies of each TSL) rather than developing more stringent management processes (e.g. specific access controls rather than general publication) or can alternatively adopt the standard central repository approach that is well known and understood from normal certification authority services.

The *safety in numbers*-concept builds on the premise that it is sufficiently difficult to insert multiple forged copies of a TSL into multiple repositories of a number of different organizations. Applications which want to validate a certain TSL therefore can retrieve copies from such repositories and compare them. Whether they only accept a TSL when all copies are equal or takes a majority vote is a policy question and out of scope of the present document.

H.6 Security issues

The security of this approach relies upon there being a reasonable number of TSPs and services, on the web sites of which the TSL and the related scheme's PKC are published, to ensure that complete replacement of these sources is a complex and difficult task. However, some specific considerations need to be made.

Where the number of services covered by any one scheme is small the low number of replications increases the vulnerability of the system. This should be overcome by encouraging the publication of the TSL and related PKC on other sites, such as those of government and industry bodies, and co-operating schemes.

Additionally, the public key corresponding to the scheme operator's signing key should be bound into a certificate by each participating TSP, and these certificates published as widely as is the list and the scheme operator's self-signed certificate. Thus, the level of complexity required of any agent intending to corrupt the TSL is increased quite significantly.

Although the idea of a harmonized TSL is to bring all scheme representations up to a consistent level of robustness, early implementations which exercise the "opt-out" implementation of a TSL may find themselves unable to publish their TSL a sufficient number of times. Taking for example a scheme operating only on a "black list" principle, it could be naïve to expect to find willing those TSPs whose services have been indicated as being in default according to the scheme's criteria - there is absolutely no incentive for them to display their own failure! A solution to this could be for such schemes to actually include within their list all TSPs falling within the scope of the scheme and making a distinct separation between those schemes who continue to operate in conformance with the "failure" criteria as well as those who fall into the "black list" zone. This could readily be accomplished by using the appropriate "status" indicators in the standard.

Additionally, some schemes may find comfort in existing within a hierarchical trust model, the wider implications of which could compensate for a small number of published copies of their TSL.

This trust decision process may be a manual one where a person assesses TSP-related information, or an automated one. It is beyond the scope of the present document to consider the complexities of how subjective manual decisions based upon TSL-derived information can be reached, whether published as a web page or printed on paper. This clause therefore focuses on the automated case only, where a signed TSL is handled by some piece of software which needs to make an automated decision.

H.7 Implications for authentication of Trust Service Tokens

Although a relying party searching a TSL for a status indication relating to the issue of some TrST it possesses, may have the desire to authenticate the TrST, the TSL generally provides only a secondary source of trust. In some circumstances it may be possible to derive from the TrST information which provides a digital identity for its issuer (e.g. a Time Stamping Token that includes the TSA's PKI certificate), and that issuer may then be located within a TSL, there are many assumptions about trust which have to be satisfied before a true authentication can be claimed by this process. One should therefore expect that, in general, further steps need be taken to authenticate the TrST.

For sufficient confidence to exist such that a TrST can be considered to be a source of primary trust (i.e. to provide sufficient confidence to the relying party that the TrST is valid and issued according to certain criteria such that the relying party can depend upon the token and the transaction for which it stands) a number of factors have to be considered, amongst which might be:

- When the TSL is of type "Generic", the strict relationship between the scheme issuing the TSL and the included service have to be understood, in terms of the processes and criteria which are vouched-for.
- When the TSL is of type "Schemes", the relationship between the scheme issuing the TSL and those other schemes to which it refers have to be understood, in terms of the processes and criteria which are vouched-for by those schemes, and in turn by those schemes and the services they list.
- Legal implications, such as the standing of the schemes concerned, and potentially whether authentication by reference to the TSL listing would be sufficient for legal evidentiary purposes (e.g. as opposed to parsing a certificate chain to a root certificate, as may be required in some jurisdictions).

With a sufficiently rigorous definition and understanding of a scheme's operation, its management processes and the criteria which it applied to determine the status of services which it listed (or other schemes which it listed, as appropriate), perhaps coupled with appropriate understanding of the liability implications, a scheme could, within a well-define community, be a source of primary trust, and therefore a source of authentication for trust services.

Annex I (informative): General TSL usage

I.1 Introduction

This annex serves to describe some general scenarios in which TSLs can be used, including how they can be located. It is not the intention to exhaustively detail all possible cases of use, nor does it assume any specific types of trust service, although it does discuss some key distinctions which are recognized by the type of TSL being used. The annex assumes familiarity with annex H, which describes how TSLs can be located and authenticated.

The present document describes two types of TSL, "Generic" and "of Schemes". This annex first considers TSLs of the "Generic" type, and then considers the alternative "of Schemes' type.

I.2 Generic TSL usage

The TSL was originally envisaged as a means to provide status information on electronic trust services falling within the scope of a scheme's oversight, whether by regulatory power or by voluntary acceptance. Such services evolved principally from those required to support Public-Key Infrastructures (PKI), although other electronic services not directly related to PKI but still providing trust through their functions were also anticipated, and the TSL structure as defined allows for these and is extensible to account for new electronic trust services as they arise.

Some examples of how a TSL can be used are given below. They do not go into great depth, but they do show the range of possible application of a TSL and the flexible nature of the present document.

I.2.1 Trusted Lists

Member States of the European Union are expected to use a common template for their national "Trusted List of supervised/accredited Certification Service Providers" in which information is provided by each Member State about the supervision/accreditation status of the certification services from Certification Service Providers who are supervised/accredited by them for compliance with the relevant provisions of Directive 1999/93/EC [1]. The common template is compatible with an implementation based on the specifications from the present document and will make use in particular of the URIs and extensions defined in annex L.

A compiled list of pointers towards EUMS TSL implementation of the EUMS Trusted Lists is expected to be organized at EC level according to a "Scheme" TSL type implementation.

I.2.2 Trust service status as legal evidence

In this case we imagine that "Consumer-alpha" denies that they ever entered a contract with "OrganizationX". "OrganizationX" holds "Alphas" e-signature on a contract, and believes it to be supported by a QC and therefore having the legal status and value which that provides. However, "OrganizationXs" company policy is not to verify the certificates on contracts below one thousand Euros. Now "OrganizationX" needs to prove its case - its legal representatives refer to the contract, find the date it was executed, then LOCATE a TSL which has oversight of the issuer (the issues around locating a TSL have now been amply discussed), and look for a record of the status of the certificate issuer on the date on which the contract was effected. There are a number of possible outcomes:

- No record in any TSL - no supporting evidence available; TSL with no history, or no history for the date in question - as previous outcome.
- History present for the required date - status good (i.e. was operating as a valid issuer of QCs at the time of issuing the certificate on which the contract signature is based - supporting evidence available:
  - status bad, may not be a QC; and
  - no obviously positive evidence to support "OrganizationXs" case.

I.2.3 Checking for anomalous status before accepting a credential

A voluntary approval scheme, "Trustscheme", is registered in one country but is an industry scheme set up for the good of many players within a larger community extending across national (and therefore legislative) boundaries. Approval by "Trustscheme" does not confer or deny any legal rights. It shows that the service is (or is not) being operated according to defined practices and criteria which are freely publicized, and that the services claiming compliance with those criteria are regularly audited. Finding such status information within the scheme's TSL will provide a secondary level of trust to a relying party. A parser could flag a bad status for checking prior to a transaction being enabled (similar to the way in which a browser may warn about a certificate it does not recognize when accessing web resources - a little window pops up and says "certificate not recognized - what do you want to do?" (not being bothered, most users will click "Accept" - but it is their or their employer's choice!)). Such flags could be based upon final value or other criteria an automated process could apply - e.g. only if from a particular country, a particular organization, etc.

I.2.4 Cross-certification status confirmation

Should a national government wish to establish a National PKI Bridge CA (NBCA), which enables a community of Cas (in the all-inclusive term of them being either separate service components or all-in registrars, issuers, status publishers, etc.) to inter-operate against equivalent policy requirements. NBCA publishes a TSL listing all those services which have been certified according to the NBCA Policy Authority. Whenever any member of the NBCA community receives some TrST it first looks in the known TSL_{NBCA} which tells it how to react. Assuming the issuer of the TrST is shown having a good status at the time of issuing the TrST and at the current time, then the TrST is given due recognition, i.e. treated according to the agreed cross-certification rules. If the issuer/service provider cannot be found, some other process has to be invoked (alert for human action, apply some other automated process, which may involve searching elsewhere), but cross-certification cannot be assumed. The textually-published TSL serves to assist subscribers and other users as to which organizations are cross-certified.

I.3 TSLs used to list other schemes

In the first version of the present document the field Pointers to other TSLs was provided. This allowed a scheme operator to provide pointers to other TSLs about which it knew, and according to whatever selection process it chose to apply (i.e. the specification imposed no specific selection criteria, even implicitly).

A specific development in the potential application of a TSL has been to make reference to other Scheme Operators and their TSLs, should those Scheme Operators issue them. From release 2.1.1 of the present document, there has been the capability to include another trust assessment scheme as a recognized "electronic trust service". The use of the TSL structure in such a case does not vary although the scheme-operator is at liberty to establish and publish their own rules for how their TSL is managed (i.e. the rule-set which applies to it).

This is based upon the principal of including another scheme operator's services as a type of trust service. This is logically consistent with the approach taken by the TSL specification: define the service, define the rules for inclusion of any specific service, apply those rules and list qualifying services accordingly. Those rules may be as rigid or as flexible as the scheme operator chooses, and need not be the same as those used by any other assessment scheme which is included.

By this means one scheme operator can be included within another TSL. It is worth noting that the *referenced* scheme need not necessarily provide its own TSL - that would be a decision factor left to the owner of the scheme which is referenced.

I.3.1 Hierarchical relationships

In this clause, the term "hierarchy" is not intended to imply that any control exists between a scheme and other assessment schemes which it may include with its TSL. It may be that controls ***do*** exist between them, but here there is no presumption or reliance of that being the case.

Where a TSL "of Schemes" refers to other assessment schemes (the referenced schemes) the operators of those referenced schemes should be regarded as TSPs. The actual schemes which they operate should be regarded as trust services. The same rules which apply to the treatment of conventional trust services and their providers apply here. This approach enables a common TSL format and accommodates an organization operating more than one scheme and publishing a TSL for each.

The following table indicates how key fields within a TSL "of Schemes" should be derive their content from fields within a referenced TSL (which could be of any type recognized by the present document or by the scheme operator which publishes the TSL "of Schemes").

"TSL of Schemes" field	Source field in the referenced TSL
TSP name	Scheme operator name
TSP address	Scheme operator address
Service name	Scheme name
Scheme service definition URI	Scheme information URI
Service digital identity	Scheme identification

Further to the above, the Service Supply Points of a "Schemes" TSL field may be used to provide the URI at which any TSLs (i.e. the TrST) issued by the listed schemes can be found (noting that an assessment scheme may issue a TSL by choice, not by any normative requirement of the present document). As indicated in clause H.7, the content of the field "Service digital identity" of a "Schemes" TSL may also be used to authenticate the TSL pointed to by these URIs. Therefore a "Schemes" TSL may be used to locate and/or authenticate TSLs issued by other schemes, if all schemes so-referenced can be relied upon to apply the same rules and field usage (e.g. by adhering to a commonly-agreed TSL profile).

One can consider a number of potential reasons for wishing to establish a TSL "of Schemes" (ToSch)- the following clauses offer a brief number of cases where a TSL can be used in this way. As they progress they illustrate use cases where the degree of certitude as to the meanings and processes in each case is greater.

I.3.2 A collection of TSLs

The previous annex acknowledged the need sometimes to search for TSLs, which could be a laborious and time-consuming process if it has to be performed frequently (in practice this should not be the case, but circumstances may vary). A beneficent entity might set up a web-crawling application to continuously crawl the internet and locate TSLs. Each time it did so it could perform checks on the TSL (identified because it had a verifiable "TSL tag") to see whether it had previously been located, and if not then the new TSL could be highlighted in order that the beneficent entity could research details of the scheme concerned, which could then be added to the TSL "of Schemes" the entity maintained. Depending on the checks it performed, and possibly filtering and rejection rules it applied, the resultant "Schemes" TSL could range from having a completely unqualified selection of other schemes, to having those schemes categorized or even selected for inclusion against defined criteria.

Such a TSL might be used by third parties who would more quickly locate other TSLs and could then apply their own specific queries to determine the TSL type and whether the service of interest was recorded. Note that in this web-crawling scenario, an un-filtered TSL "of Schemes" may include other TSLs "of Schemes", which users would need to recognize in order to correctly handle them.

I.3.3 Schemes applying common rules

Within a well-defined community, e.g. the EU or EFTA, there are a number of sovereign states working within a common legislative framework. Different states may (and generally do) implement framework legislation in different ways, but within the scope of the framework. A "RegionalBridge" might address this need.

In the European Union it could be used as follows. Each country may have a supervisory system: one might observe that they are likely to vary and some schemes publish a TSL, not all will do. There may be no obvious (i.e. consistent, normalized) way to locate these schemes, or any TSL they may operate - different ministries are involved, some schemes are outsourced to an industry body and no standardized naming conventions are recognized.

A central body might sponsor a simple scheme to merely list all supervisory schemes of the participating states. This could also be extended to include also voluntary schemes - it would be for the central scheme operator to define within their TSL how they did this. The provision now within the TSL specification for scheme operators to use registered URIs would facilitate the distinction between supervisory systems and voluntary schemes (see Service type identifier, in the context of "Scheme" TSLs). In the absence of a central body to support such a TSL, any other national body may provide such a function which might become widely recognized as a reliable reference source.

A similar use case exists for defined industry sectors, e.g. aerospace / defence / automotive / etc.

I.3.4 Schemes trusted by a vendor community

In a commercial use case, one might suppose that a large software company, "Megatuff", wants to add to its browsers a capability to add secondary trust to any certificates used in web sites and related services but has a problem in knowing where such trust may be found. It implements a scheme which publishes a TSL listing only other schemes which provide a degree of secondary trust which. "Megatuff" defines some basic requirements that these schemes must fulfil and then adds to its TSL all those which meet those requirements. Where regional considerations dictate, a hierarchy might be created: TSLglobal, which points to TSLregionA, TSLregionB, etc. Thus a set two-level hierarchy of TSLs "of Schemes" is created, perhaps locally managed against common policy.

I.3.5 Industrial trading consortium

In the final use case, we consider a Trans-Oceanic Consortium (TOC) which wants to establish some common rules for the identity proofing and credential-issuing of participants within a collaborative industry network. National criteria apply and must be fulfilled by industry located in that region. Assuming that participants within the consortium are required to use credentials issued by a service provider whose service has been assessed for compliance with the common rules, the TOC has two possible approaches to help consortium members check the status of their own and their counter-parties' services:

a) establish a "Generic" TSL, which individually lists each suitable service provider. In this case oversight may be difficult, since the TOC would need to effectively operate an assessment process of its own (even if outsourced);

b) establish a "Schemes" TSL, which referred to schemes which might be nationally established or which were industry / sector-based (see previous regional case).

The above use cases cover a broad spectrum of potential application of the TSL in both its types as defined within the present document. Adoption of the present document by assessment schemes will resolve the specifics and provide practical lessons.

Annex J (informative): TSL manual/auto field usage

The following table lists all fields defined for the TSL and indicates whether the field contents should be made available to users when presenting the TSL in a human-readable form (column 2) or whether the field is considered to be essential for effective automatic parsing (column 3), noting that all fields will be accessible through an automated process.

Although this annex is informative implementers are strongly recommended to satisfy the guidance which it provides, in order to provide users with information about TSLs in a consistent manner.

Field name	Human-readable?	Machine-processable?
Identification Tag		
TSL tag		✓
Scheme information		
TSL version identifier		✓
TSL sequence number		✓
TSL type	✓	✓
Scheme operator name	✓	
Scheme operator address	✓	
Scheme name	✓	
Scheme information URI	✓	✓
Status determination approach	✓	✓
Scheme type/community/rules	✓	✓
Scheme territory	✓	✓
TSL policy/legal notice	✓	✓
Historical information period	✓	✓
Pointers to other TSLs	✓	✓
List issue date and time	✓	✓
Next update		✓
Scheme extensions	where recognized and meaningful	where recognized
TSP information		
TSP name	✓	
TSP trade name	✓	
TSP address	✓	
TSP information URI	✓	✓
TSP information extensions	where recognized and meaningful	where recognized
Service information		
Service type identifier	✓	✓
Service name	✓	
Service digital identity	✓	✓
Service current status	✓	✓
Current status starting date and time	✓	✓
Scheme service definition URI	✓	✓
Service supply points	✓	✓
TSP service definition URI	✓	✓
Service information extensions	where recognized and meaningful	where recognized
Historical service information		
Service type identifier	✓	✓
Service name	✓	
Service digital identity	✓	✓
Service previous status	✓	✓
Previous status starting date and time	✓	✓
Service information extensions	where recognized and meaningful	where recognized
TSL signature information		
Scheme identification		✓
Textual certificate details, time and date of signing	✓	✓
Cryptographic data		✓

Annex K:
Void

Annex L (normative): URIs and extensions used for the EU Member States' national Trusted List of supervised/accredited Certification Service Providers

L.1 Introduction

Member States of the European Union are expected to use a common template for their national "Trusted List of supervised/accredited Certification Service Providers" in which information is provided by each Member State about the supervision/accreditation status of the certification services from Certification Service Providers who are supervised/accredited by them for compliance with the relevant provisions of Directive 1999/93/EC [1]. The common template is compatible with an implementation based on the specifications from the present document and will make use in particular of the URIs and extensions defined in the present annex L.

L.2 eSig Directive URIs

The following URIs, are registered under the radix "http://uri.etsi.org/TrstSvc/……" registered in annex D.

http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/TSLType/generic	TSL type
A TSL implementation of a supervision/accreditation status list of certification services from certification service providers which are supervised/accredited by the referenced Member State owning the TSL implementation for compliance with the relevant provisions laid down in the Directive 1999/93/EC [1] of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures, through a process of direct oversight (whether voluntary or regulatory).	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /TSLType/schemes	
A TSL implementation of a compiled list of pointers towards Member States supervision/accreditation status lists of certification services from certification service providers which are supervised/accredited by the referenced Member State owning the pointed TSL implementation for compliance with the relevant provisions laid down in the eSignature Directive 1999/93/EC [1] of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures, through a process of direct oversight (whether voluntary or regulatory).	

http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/StatusDetn/appropriate	Status determination approach (see below the section Supervision/accredition status flow)
Services listed have their status determined by or on behalf of the Scheme Operator under an appropriate system for a referenced Member State that allows for 'supervision' (and when applicable for 'voluntary accreditation') of certification service providers who are established on its territory (or established in a third country in the case of 'voluntary' accreditation') and issue qualified certificates to the public according to Art. 3.3 (respectively Art. 3.2 or Art. 7.1(a)) of the Directive 1999/93/EC [1] of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures, and, when applicable, that allows for the 'supervision' / 'voluntary accreditation' of certification service providers not issuing qualified certificates, according to a nationally defined and established "recognized approval scheme(s)" implemented on a national basis for the supervision of compliance of services from certification service providers not issuing Qualified Certificates with the provisions laid down in Directive 1999/93/EC [1] and potentially extended by national provisions with regards to the provision of such certification services.	

http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/OCSP-QC	Service information extensions / additionalServiceInformation Extension
a certificate status provider operating an OCSP-server as part of a service from a certification service provider issuing Qualified Certificates. Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/Certstatus/OCSP	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/CRL-QC	
a certificate status provider operating a CRL as part of a service from a certification service provider issuing Qualified Certificates. Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/Certstatus/CRL	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/RootCA-QC	
a Root Certification Authority from which a certification path can be established down to a Certification Authority issuing Qualified Certificates. Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/CA/QC	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/TSS-QC	
a time stamping service as part of a service from a certification service provider issuing Qualified Certificates that issue TST that can be used in the qualified signature verification process to ascertain and extend the signature validity when the QC is revoked or expired.	
Each of the above four URIs MUST be used at service level in an "additionalServiceInformation" extension (see clause 5.8.2) in the field defined in clause 5.5.9. The usage of the "RootCA/QC" URIs MAY be combined with the below defined URIs in accordance with the specifications provided in clause L.2, i.e. with those below URIs used in a "Qualifications" extension in the field defined in clause 5.5.9.	

http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/QCWithSSCD	Service information extensions/Qualifications Extension/Qualifiers
QCWithSSCD it is ensured by the certification service provider and controlled (supervision model) or audited (accreditation model) by the referenced Member State (respectively its Supervisory Body or Accreditation Body) that any Qualified Certificate issued under the service (RootCA/QC or CA/QC) identified in "Service digital identity" and further identified by the filters information used to further identify under the "Sdi" identified certification service that precise set of Qualified Certificates for which this additional information is required with regards to the presence or absence of Secure Signature Creation Device (SSCD) support ARE supported by an SSCD (i.e. that that the private key associated with the public key in the certificate is stored in a Secure Signature Creation Device conformant with annex III of Directive 1999/93/EC [1]); Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/CA/QC	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/QCNoSSCD	
QCNoSSCD it is ensured by the certification service provider and controlled (supervision model) or audited (accreditation model) by the referenced Member State (respectively its Supervisory Body or Accreditation Body) that any Qualified Certificate issued under the service (RootCA/QC or CA/QC) identified in "Service digital identity" and further identified by the filters information used to further identify under the "Sdi" identified certification service that precise set of Qualified Certificates for which this additional information is required with regards to the presence or absence of Secure Signature Creation Device (SSCD) support ARE NOT supported by an SSCD (i.e. that that the private key associated with the public key in the certificate is not stored in a Secure Signature Creation Device conformant with annex III of the Directive 1999/93/EC [1]). Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/CA/QC	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/QCSSCDStatusAsInCert	
QCSSCDStatusAsInCert it is ensured by the certification service provider and controlled (supervision model) or audited (accreditation model) by the referenced Member State (respectively its Supervisory Body or Accreditation Body) that any Qualified Certificate issued under the service (RootCA/QC or CA/QC) identified in "Service digital identity" and further identified by the filters information used to further identify under the "Sdi" identified certification service that precise set of Qualified Certificates for which this additional information is required with regards to the presence or absence of Secure Signature Creation Device (SSCD) support SHALL contain the machine-processable information indicating whether or not the Qualified Certificate is supported by an SSCD. Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/CA/QC.	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/SvcInfoExt/QCForLegalPerson	
QCForLegalPerson it is ensured by the certification service provider and controlled (supervision model) or audited (accreditation model) by the referenced Member State (respectively its Supervisory Body or Accreditation Body) that any Qualified Certificate issued under the service (RootCA/QC or CA/QC) identified in "Service digital identity" and further identified by the filters information used to further identify under the "Sdi" identified certification service that precise set of Qualified Certificates for which this additional information is required with regards to the issuance to Legal Person ARE issued to Legal Persons. Only to be used as an extension, if the servicetype is http://uri.etsi.org/TrstSvc/Svctype/CA/QC	

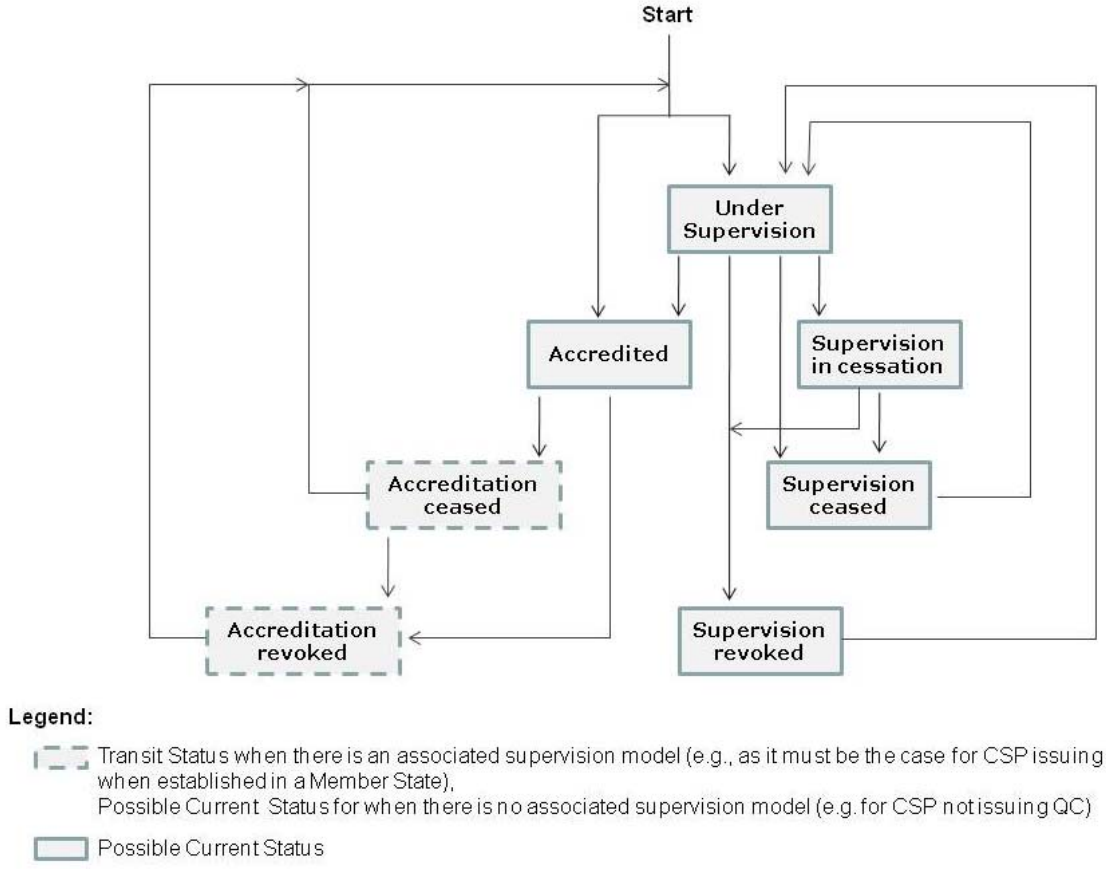
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/schemerules/common A URI common to all Member States' Trusted Lists pointing towards a descriptive text that SHALL be applicable to all Trusted Lists: • By which participation is denoted of the Member State's scheme (identified via the "TSL type" (see clause 5.3.3) and "Scheme name" (clause 5.3.6)) in a scheme of schemes (i.e. a TSL listing pointers to all Member States publishing and maintaining a Trusted List in the form of a TSL); • Where users can obtain policy/rules against which services included in the list SHALL be assessed and from which the type of the TSL (see clause 5.3.3) can be determined; • Where users can obtain description about how to use and interpret the content of the TSL implementation of the Trusted List. These usage rules SHALL be common to all Member States' Trusted Lists whatever the type of listed service and whatever the supervision/accreditation system(s) is (are).	Scheme type/community/rules
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/schemerules/CC where CC is replaced with the code used in the "Scheme territory" field (see clause 5.3.10) A URI specific to each Member State's Trusted List pointing towards a descriptive text that SHALL be applicable to this Member State's Trusted List: • Where users can obtain the referenced Member State's specific policy/rules against which services included in the list SHALL be assessed in compliance with the Member State's appropriate supervision system and voluntary accreditation schemes. • Where users can obtain a referenced Member State's specific description about how to use and interpret the content of the TSL implementation of the Trusted List with regard to the certification services not related to the issuing of Qualified Certificates. This may be used to indicate a potential granularity in the national supervision/accreditation systems related to certification service providers not issuing Qualified Certificates and how the "Scheme service definition URI" (see clause 5.5.6) and the "Service information extension" field (see clause 5.5.9) are used for this purpose.	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList/schemerules/CompiledList A URI pointing towards a descriptive text where users can obtain information about the scheme of schemes type (i.e. a TSL listing pointers to all Member States' Trusted Lists published and maintained in the form of a TSL) and the relevant driving rules and policy.	

http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/undersupervision	Service current status
Under Supervision The service identified in "Service digital identity" (see clause 5.5.3) provided by the Certification Service Provider (CSP) identified in "TSP name" (see clause 5.4.1) is currently under supervision, for compliance with the provisions laid down in Directive 1999/93/EC [1], by the Member State identified in the "Scheme territory" (see clause 5.3.10) in which the CSP is established.	
http: //uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/supervisionincessation	
Supervision of Service in Cessation The service identified in "Service digital identity" (see clause 5.5.3) provided by the Certification Service Provider (CSP) identified in "TSP name" (see clause 5.4.1) is currently in a cessation phase but still supervised until supervision is ceased or revoked. In the event a different legal person than the one identified in "TSP name" has taken over the responsibility of ensuring this cessation phase, the identification of this new or fallback legal person (fallback CSP) shall be provided in clause 5.5.6 of the service entry.	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/supervisionceased	
Supervision Ceased The validity of the supervision assessment has lapsed without the service identified in "Service digital identity" (see clause 5.5.3) being re-assessed. The service is currently not under supervision any more from the date of the current status as the service is understood to have ceased operations.	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/supervisionrevoked	
Supervision Revoked Having been previously supervised, the Certification Service Provider (CSP)'s service and potentially the CSP itself has failed to continue to comply with the provisions laid down in Directive 1999/93/EC, as determined by the Member State identified in the "Scheme territory" (see clause 5.3.10) in which the CSP is established. Accordingly the service has been required to cease its operations and must be considered as ceased for the above reason. NOTE: The status value "Supervision Revoked" can be a definitive status, even if the CSP then completely ceases its activity; there is no need to migrate to either "Supervision of Service in Cessation" or to "Supervision Ceased" status in this case. Actually, the only way to change the "Supervision Revoked" status is to recover from non-compliance to compliance with the provisions laid down in Directive 1999/93/EC according the appropriate supervision system in force in the Member State owing the Trusted List, and regaining "Under Supervision" status. "Supervision of Service in Cessation" status, or "Supervision Ceased" status only happens when a CSP directly ceases its related services under supervision, not when supervision has been revoked.	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/accredited	
Accredited An accreditation assessment has been performed by the Accreditation Body on behalf of the Member State identified in the "Scheme territory" (see clause 5.3.10) and the service identified in "Service digital identity" (see clause 5.5.3) provided by the Certification Service Provider (CSP) identified in "TSP name" (see clause 5.4.1) is found to be in compliance with the provisions laid down in Directive 1999/93/EC [1]. This accredited CSP may be established in another Member State than the one identified in the "Scheme territory" (see clause 5.3.10) of the TSL implementation of the Trusted List or in a third country (see article 7.1(a) of Directive 1999/93/EC [1]).	
http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/accreditationceased	
Accreditation Ceased The validity of the accreditation assessment has lapsed without the service identified in "Service digital identity" (see clause 5.5.3) being re-assessed.	

http://uri.etsi.org/TrstSvc/eSigDir-1999-93-EC-TrustedList /Svcstatus/accreditationrevoked	
Accreditation Revoked Having been previously found to be in conformance with the scheme criteria, the service identified in "Service digital identity" (see clause 5.5.3) provided by the Certification Service Provider (CSP) identified in "TSP name" (see clause 5.4.1) and potentially the CSP itself have failed to continue to comply with the provisions laid down in Directive 1999/93/EC [1].	

Supervision/accreditation status flow

When used in the context of a Certification Service Provider (CSP) issuing Qualified Certificates (QCs) that is established in the "Scheme territory" (clause 5.3.10), the two statuses "Accreditation Revoked" and "Accreditation Ceased" MUST be considered as "transit statuses" and MUST NOT be used as value for "Service current status" as, in case they are used, they MUST be immediately followed in the "Service approval history information" or in the "Service current status" by an "Under Supervision" status, potentially followed by any other supervision status defined here above and as illustrated in the figure below.



When used in the context of a CSP not issuing QCs when there is only an associated "voluntary accreditation" scheme with no associated supervision scheme or in the context of a CSP issuing QCs where the CSP is not established in the "Scheme territory" (e.g. a third country), those "Accreditation Revoked" and "Accreditation Ceased" statuses MAY be used as value for "Service current status".

Exactly the same status values must be used for CSPs issuing QCs and for CSPs not issuing QCs (e.g. Time Stamping Service Providers issuing TSTs, CSP issuing non-qualified certificates, etc.). The "Service Type identifier" (see clause 5.5.1) shall be used to distinguish between applicable supervision/accreditation systems.

Additional status-related "qualification" information defined on the level of national supervision/accreditation systems for CSPs not issuing QCs MAY be provided at the service level when applicable and required (e.g. to distinguish between several quality/security levels). Scheme Operators SHALL use the "additionalServiceInformation" extension (clause 5.8.2) of the *"Service information extension"* field (see clause 5.5.9) according to the purpose of providing such additional "qualification" information. Additionally, the Scheme operator can optionally use clause 5.5.6 ("Scheme service definition URI).

L.3 eSig Directive Extensions

The XML implementation of the extensions in the following clauses of this clause is included in the file referenced in clause C.3 (point 2). No implementation in ASN.1 is provided with the present version of the document.

L.3.1 Qualifications Extension

Description: This extension is OPTIONAL but SHALL be present when specific criteria is required to fully qualify a CA service and/or part or all the certificates it issues, e.g. when:

- the information provided in the "Service digital identity" is not sufficient to unambiguously identify the characteristics of the certificates issued by this service (e.g. being a QC);
- the information present in the related qualified certificates does not allow machine-processable identification of the facts about whether or not the QC is supported by an SSCD.

When present, this extension MUST only be used in the service level field defined in clause 5.5.9.

If this extension is marked "critical" a compliant implementation SHALL discard the certificate under validation if it cannot parse and fully understand its semantic.

The qualification criteria is specified by a set of Qualification Elements, each one expressed as a list of assertions to be verified and a list of qualifiers that apply to the examined certificate when all the assertions are verified. The certificate is qualified with all the qualifiers obtained with the application of all the qualification elements.

Format: A non-empty sequence of one or more Qualification Elements defined below in clause L.3.1.1. For the formal definition see `Qualifications` element in the schema referenced by clause C.3 (point 2).

L.3.1.1 QualificationElement

Description: This field bundles a list of assertions that specifies the attributes a certificate must have (e.g. certain key-usage-bits set) and a list of qualifiers that specify some certificate properties (e.g. it is a qualified certificate).

Format: A tuple consisting of a list of assertions (CriteriaList, see clause L.3.1.2) and a list of qualifiers (Qualifiers, see clause L.3.1.3). For the formal definition see `QualificationElementType` element in the schema referenced by clause C.3 (point 2).

Meaning: Each QualificationElement gives additional information needed to identify whether a given certificate issued by this service is qualified and supported by an SSCD or not, and/or information regarding the fact if such QCs can be issued to Legal Persons.

L.3.1.2 CriteriaList

Description: This field is REQUIRED and SHALL provide a list of assertions related to certificate contents (e.g. key usage) and status (e.g. additional assessment) used to filter certificates. An assertion can be itself a CriteriaList allowing a recursive definition. An optional Description field is present to allow the schema operator to specify the rationale of the defined criteria. If present it is RECOMMENDED that the description is expressed in English.

Format: A non-empty sequence of assertions whose syntax is specified in clauses L.3.1.2.1 to L.3.1.2.3 followed by a matching criteria indicator that can assume the following values: For the formal definition see `CriterieListType` element in the schema referenced by clause C.3 (point 2).

- "all" if all of the assertion MUST be me;
- "one" if at least one of the assertion MUST be met; or
- "none" if all the assertions MUST NOT be met;

for the given set of qualifiers, related to the CriteriaList, to apply.

L.3.1.2.1 KeyUsage

Description: This field is OPTIONAL and provides a list of key usage bit-values to match with the correspondent bits present in the keyUsage certificate Extension. The assertion is verified if the KeyUsage Extension is present in the certificate and all key usage bits provided are matched with the corresponding bit in the certificate KeyUsage Extension.

Format: A non-empty sequence of tuples composed by a Key Usage Bit identifier and the asserted value. The key usage bits identifiers MUST be those defined in X.509 [43] for the KeyUsage Extension. For the formal definition see `KeyUsageType` element in the schema referenced by clause C.3 (point 2).

L.3.1.2.2 PolicySet

Description: This field is OPTIONAL and provides list of Certificate Policy identifiers to match with the content of the CertificatePolicy certificate Extension. The assertion is verified if the CertificatePolicy Extension is present in the certificate and all the Certificate Policy identifiers provided are present in the certificate CertificatePolicy Extension.

Format: A sequence of one of more Object Identifiers indicating a Certificate Policy. For the formal definition see `PoliciesListType` element in the schema referenced by clause C.3 point 2).

L.3.1.2.3 OtherCriteria

This field is OPTIONAL and allows the inclusion of new criteria that can be required by EU Member States for additional assertions on certificate content/status. Here follows some OtherCriteria definition, new criteria can be added in future. If not included in the following list it is the responsibility of the schema operator that defines a new criteria to publish the related definition in an effective way.

1) ExtendedKeyUsage

Description: This field is OPTIONAL and provides a non empty list of key purposes values to match with the correspondent KeyPurposes present in the ExtendedKeyUsage certificate Extension. The assertion is verified if the ExtendedKeyUsage Extension is present in the certificate and all key purposes provided are present in the certificate ExtendedKeyUsage Extension.

Format: A non-empty sequence of KeyPurposes, whose semantic is defined in X.509 [43] for the ExtendedKeyUsage Extension. For the formal definition see `ExtendedKeyUsage` element in the schema referenced by clause C.3 (point 3).

2) CertSubjectDNAttribute

Description: This field is OPTIONAL and provides a non empty set of OIDs. Each OID maps to a possible attribute in the Subject DN of the certificate. The criteria is matched if all OID refers to an attribute present in the DN.

Format: A non-empty sequence of OIDs representing Directory attributes, whose meaning respect the description above. For the formal definition see `CertSubjectDNAttribute` element in the schema referenced by clause C.3 (point 3).

L.3.1.3 Qualifier

Description: This field is REQUIRED and specifies the properties a certificate with the specified criteria possesses.

Format: Sequence of URIs whose value SHALL be one of the following: QCSSCD, QCNoSSCD, QCSSCDStatusAsInCert, QCForLegalPerson, whose syntax is specified in clause L.2.

Meaning: Each URIs specifies one property a certificate has, which fulfils the set of criteria specified.

L.3.2 TakenOverBy Extension

Description: This extension is OPTIONAL but SHALL be present when a service that was formerly under the legal responsibility of a CSP is taken over by another TSP and is meant to state formally the legal responsibility of a service and to enable the verification software to display to the user some legal detail. The related service entry in the TSL MUST NOT be copied or moved inside the new TSP list of services and it is under the responsibility of the schema operator to maintain updated the correct service trust state. If the new TSP issues a new digital identity related to this service (e.g. a new self signed certificate for a CA) then a new service entry MUST be created. If the previous service is still in operation, even for a limited scope (e.g. CRL issuing as for the example above) its status MUST be maintained by the schema operator, according to the established rules, until the service terminates its operations.

This extension contains an URI, pointing towards a descriptive text to inform the user about who is the entity currently responsible for the service and any detail the schema operator considers useful. In addition it contains a set of attributes, uniquely identifying the taking over TSP allowing the application to locate this TSP in the TSL and to display its details.

The content of this extension is not meant to enforce any specific action on the signature validation.

When present, this extension MUST only be used in the service level field defined in clause 5.5.9.

If this extension is marked "critical" a compliant implementation SHALL discard the certificate under validation if it cannot parse and fully understand its semantic.

Format: This extension contains an URI, and a sequence of the following attributes:

- The TSP name, as defined in clause 5.4.1.
- The Scheme operator name and territory, as specified in clauses 5.3.4 and 5.3.10.
- An optional additional information field for further qualification of the taking over TSP, defined in following versions of the present document or by schema operators as schema specific.

This extension is implemented with the `TakenOverBy` element defined in the schema referenced by clause C.3 (point 3).

Annex M (informative): Changes since the last major version

This annex lists all changes since the last major version of the present document that are or may be relevant to a user of the standard. Fixing typos or changes like adding acknowledges are not listed.

M.1 Changes from v2.1.1 to v3.1.1

Clause	Change(s)
2	multiple outdated RFC references replaced by current versions.
3.1	definition of *Trusted List* added.
4	List type - now defined using the "Pointer to other TSL" fields only.
5.1.3	For EU-wide schemes, two languages (EUMS of issue and UK English) are required for multilingual elements.
5.1.4	Removed format constraints in Date-Time information; 5.1.4 was incompatible to annex B and XML-date/time-encoding practices.
5.3.1	Version number incremented; now "3".
5.3.5.1	Postal code only to be used if applicable; there are countries without postal codes.
5.3.8	The status determination approach field can now also contain a distinctive value, if all the TSLs contained as TSPs follow the given status determination approach. In all other cases, the new type "List", replacing the null-type, must be used.
5.3.12	For List-type TSLs, no history is preservable, as they do not contain any services,
5.3.13	Added the possibility to provide digital identities for each TSL pointed to, that can be used to verify the signature of these TSLs. MUST be present in TSLs of type List.
5.3.16	New: DistributionPoint Field
5.3.18	List-type TSLs MUST NOT contain any service providers.
5.5.1	Added new service types for Registered Electronic Mail; Signature Policy Authority; TSL Issuer. Changed the wording for supervisory and voluntary schemes to make them more generic. Adding the possibility to use additional URIs for entirely new service types.
5.5.4	Possibility to use additional URIs for Service current status.
5.8	Generic section for defining extensions (that may be used in one section or multiple sections) added.
5.8.1	New: expiredCertsRevocationInfo Extension.
5.8.2	New: additionalServiceInformation.
6.1.1	Cleaned up some LDAP issues. Content types now defined as application/vnd.etsi.tsl.der /application/vnd.etsi.tsl.xml as registered with IANA.
Several Annexes	Changes reflecting changes in the general section as listed above.
A.7	Restructured; added new extension.
B.7	Added new extension definition.
Annexes D and I	Removed some of the Dir-1999-93-EC references where we now know they are not going to be used (clauses D.2 and annex I).
Annex L	New Annex for the Member States' "Trusted List of supervised/accredited Certification Service Providers".
Annex M	Added this clause, listing all relevant changes made.

M.2 Changes from v3.1.1 to v3.1.2

Clause	Change(s)
5.1.6	Section added to specify Country Codes values not conformant to ISO 3166-1 [21].
5.1.7	Section added for better clarity with a new supported human readable format (PDF/A).
5.3.2	Sequence number can be incremented by values greater than 1.
5.3.13	Clarified that now also a pointer to the current Signing certificate or related public key can be added.
5.3.13.1	Clarified use of Additional Information and added MimeType to allow TSL selection based on format.
5.3.15	Allowed publication of not scheduled TSL and clarified a security issue.
5.7.1	Allowed other TSL authentication means other than signature.

Clause	Change(s)
5.7.1 to 5.7.4	Clarified that specified fields must be in the signature implementation and are not TSL fields.
5.8.2	Clarified that more than one additionalServiceInformation Extension can be present to qualify a service.
6.1 and subclauses	Allowed HTTP publication as alternative to LDAP for TSL publication and extended prescriptions to PDF format.
6.2	Recommendations for TSL Signer certificate content.
Annex A and B	Aligned with changes required by new prescriptions in other parts of the document.
A.6.2.1	Updated with new CAdES prescriptions.
B.7.2	OtherInformation field in AdditionalServiceInformation was not optional: fixed.
Annex C	Updated with new files associated with the present document and related hashes.
F.1	Previous considerations were not updated.
F.2	Updated with new CAdES prescriptions.
F.4	Added section for PDF/A format TSL.
H.4	Added clarifications on TSL signing certificate verification.

History

Document history		
V1.1.1	October 2003	Publication (Historical)
V2.1.1	March 2006	Publication (Historical)
V3.1.1	October 2009	Publication (Historical)
V3.1.2	December 2009	Publication