XML Schema Documentation

Table of Contents

top

Schema Document Properties

Target Namespace	**http://ep.nbu.gov.sk/kca/tsl/x509types#**
Element and Attribute Namespaces	• Global element and attribute declarations belong to this schema's target namespace. • By default, local element declarations belong to this schema's target namespace. • By default, local attribute declarations have no namespace.
Documentation	This document is not a part of ETSI TS 119 612. It contains extensions of ETSI TS 119 612, which support requirements of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market.

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.w3.org/2001/XMLSchema
xml	http://www.w3.org/XML/1998/namespace
tlx509	**http://ep.nbu.gov.sk/kca/tsl/x509types#**
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance

Schema Component Representation

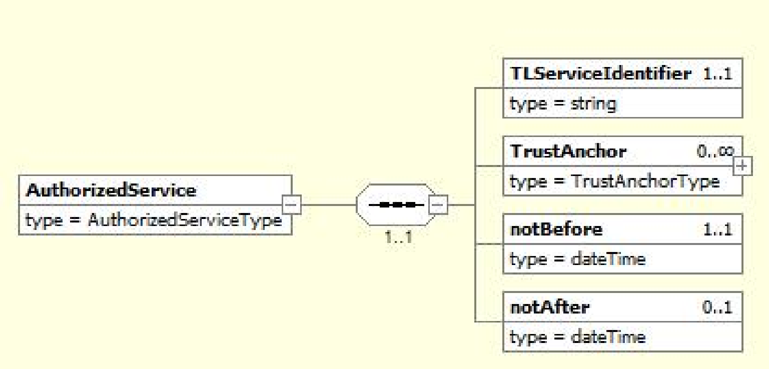
```
<schema targetNamespace="http://ep.nbu.gov.sk/kca/tsl/x509types#"
 elementFormDefault="qualified" attributeFormDefault="unqualified">
   ...
</schema>
```

top

Global Declarations

Element: AuthorizedService

Name	AuthorizedService
Type	tlx509:AuthorizedServiceType

Nillable	no
Abstract	no
Documentation	The "AuthorizedService" element specifies the other service authorized to providing services on behalf of this service. The other service is identified by "TLIxx-y" or "TLIxx-" included in "TLServiceIdentifier", where "xx" is a Country Code from 'Scheme territory' (TS 119 612 v2.1.1,5.3.10) and "y" is a positive number (service name index) in TL "Name" of "ServiceName" as "(y) z", where the "z" is a service name or when "(y) " is not included in "Name" of "ServiceName" then the other service is identified by "TLIxx-" and by one or more "TrustAnchor" elements containing, e.g. certificates in "TLServiceX509Certificate" element. The elements "notBefore" and "notAfter" determine the time interval, in which the other service provider is authorized to provide the service on behalf of this service provider.
Diagram	 AuthorizedService type = AuthorizedServiceType 1..1 TLServiceIdentifier 1..1 type = string TrustAnchor 0..∞ type = TrustAnchorType notBefore 1..1 type = dateTime notAfter 0..1 type = dateTime

XML Instance Representation

```
<tlx509:AuthorizedService>
   <tlx509:TLServiceIdentifier> xsd:string </tlx509:TLServiceIdentifier> [1]
   <tlx509:TrustAnchor> ... </tlx509:TrustAnchor> [0..*]
   <tlx509:notBefore> xsd:dateTime </tlx509:notBefore> [1]
   <tlx509:notAfter> xsd:dateTime </tlx509:notAfter> [0..1]
</tlx509:AuthorizedService>
```

Schema Component Representation

```
<element name="AuthorizedService" type="tlx509:AuthorizedServiceType"/>
```

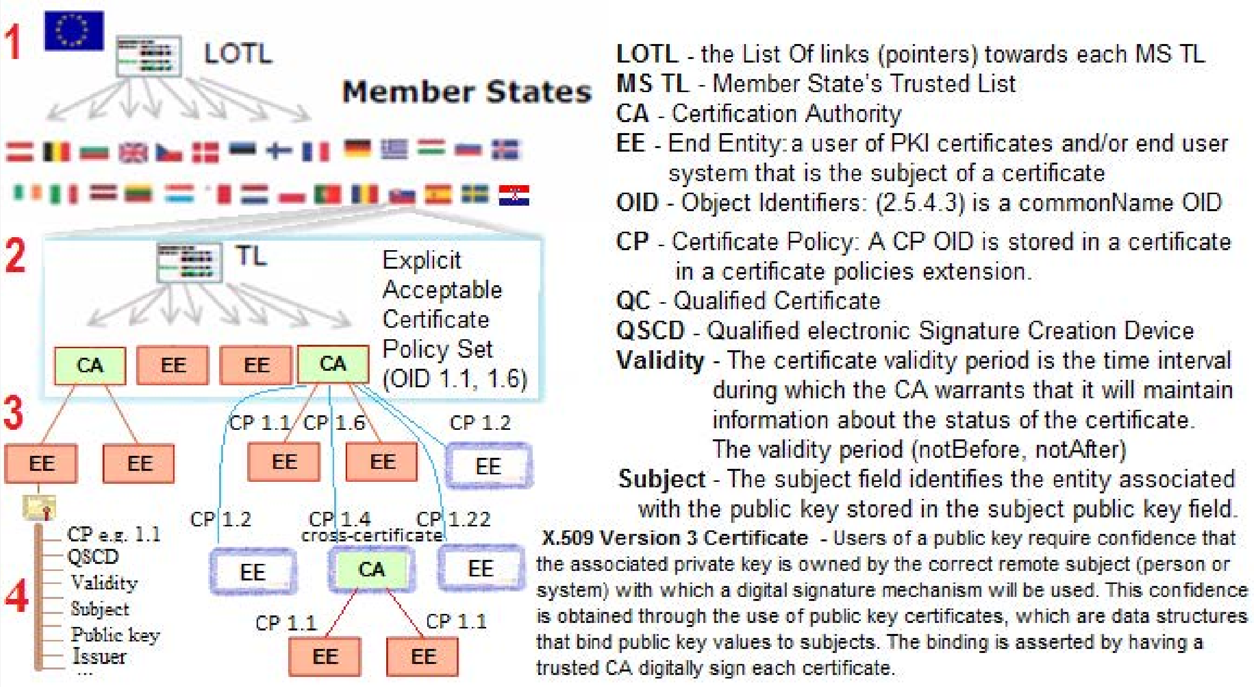
top

Element: ExplicitAcceptablePolicySet

Name	ExplicitAcceptablePolicySet
Type	tlx509:ExplicitAcceptablePolicySetType
Nillable	no
Abstract	no
Documentation	The ExplicitAcceptablePolicySet is not specified in TS 119 612 v2.1.1 and therefore it is specified in this document and shall be included in sub-element "Other" of the element "Service digital identity" specified in TS 119 612 v2.1.1 Clause 5.5.3. The content of the ExplicitAcceptablePolicySet element is used in: a) the initial-policy-set path processing variable and b) the initial-explicit-policy indicator which is set to the value "true", for more details see Clause 10.3 (Certification path processing procedure - Path processing variables) of ITU-T Rec. X.509 or ISO/IEC 9594-8 The ExplicitAcceptablePolicySet element contains the set of explicit acceptable policy identifiers included in the sub-element OID. The OID element of the ExplicitAcceptablePolicySet element contains one Object Identifier in a dot notation e.g. 1.3.158.36061701.0.0.0.1.2.2 value. NOTE: It is a usual practice to identify whether the X.509v3 certificate is issued by the qualified trust service according to OID in the certificate policies extension when such trust service is also the issuer of non-qualified certificates (is also a non-qualified trust service). The ExplicitAcceptablePolicySet element determines the selection of certificates issued by this service according to OID in the certificate policy extension; this service fulfils the type of the service (this type is published in the trusted list) when this service is the issuer of qualified and also non-qualified types of certificates. EXAMPLE: The acceptable certificate policies defined for the qualified trust services provider are the following: - OIDs for the identification of the national rules, which must be fulfilled by qualified trust service providers. Such rules are e.g. delegated by Regulation (EU) No 910/2014 to the national law as is the suspension laid down in Article 28(5) or creation of the trust infrastructure laid down in Article 17(5).

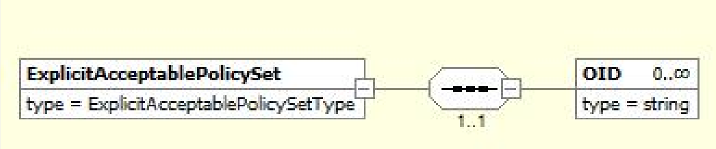
a) OID 1.3.158.36061701.0.0.0.1.2.2: The certificate policy for certificates issued by the qualified trust service provider, where:
1. the usage of suspension is forbidden,
2. when the trust infrastructure implements the OCSP then the OCSP response must contain CertHash OID 1.3.36.8.3.13 (a positive statement is an extension defined for the type SingleResponse in singleExtensions) and
3. the Member State’s supervisory body database must be the additional store (the store is actualized at least once a month) which contains all qualified certificates (at least the certificates once they reach their expiration date) issued by supervised qualified trust service providers together with their validity status. This store guarantees the qualified certificate usage and the qualified certificate verification after the certificate expiration and also after the date when the qualified status of the provider issuing the certificate or of the affected service is withdrawn. It means, the supervisory body database is used in the situation when the trust service provider is not able to remedy any failure to fulfil requirements under Regulation (EU) No 910/2014 and where that provider does not act accordingly.
b) OID of equivalent certificate policies recognised by this trust service provider.

- OIDs in ETSI standards ETSI TS 101 456 and ETSI EN 319 411-2 V2.1.1 (2016-02):
a) OID 0.4.0.1456.1.1 (QCP+) QCP public + SSCD: the certificate policy for qualified certificates issued to the public, requiring the use of secure signature-creation devices,
b) OID 0.4.0.1456.1.2 (QCP) QCP public: the certificate policy for qualified certificates issued to the public,
c) OID 0.4.0.194112.1.0 (QCP-n): the certificate policy for EU qualified certificates issued to natural persons,
d) OID 0.4.0.194112.1.1 (QCP-l): the certificate policy for EU qualified certificates issued to legal persons,
e) OID 0.4.0.194112.1.2 (QCP-n-qscd): the certificate policy for EU qualified certificates issued to natural persons with private key related to the certified public key in a QSCD,
f) OID 0.4.0.194112.1.3 (QCP-l-qscd): the certificate policy for EU qualified certificates issued to legal persons with private key related to the certified public key in a QSCD,
g) OID 0.4.0.194112.1.4 (QCP-w): the certificate policy for EU qualified web site authentication certificates



LOTL - the List Of links (pointers) towards each MS TL
MS TL - Member State's Trusted List
CA - Certification Authority
EE - End Entity: a user of PKI certificates and/or end user system that is the subject of a certificate
OID - Object Identifiers: (2.5.4.3) is a commonName OID
CP - Certificate Policy: A CP OID is stored in a certificate in a certificate policies extension.
QC - Qualified Certificate
QSCD - Qualified electronic Signature Creation Device
Validity - The certificate validity period is the time interval during which the CA warrants that it will maintain information about the status of the certificate. The validity period (notBefore, notAfter)
Subject - The subject field identifies the entity associated with the public key stored in the subject public key field.
X.509 Version 3 Certificate - Users of a public key require confidence that the associated private key is owned by the correct remote subject (person or system) with which a digital signature mechanism will be used. This confidence is obtained through the use of public key certificates, which are data structures that bind public key values to subjects. The binding is asserted by having a trusted CA digitally sign each certificate.

Diagram


ExplicitAcceptablePolicySet
type = ExplicitAcceptablePolicySetType
1..1
OID 0..∞
type = string

XML Instance Representation

```
<tlx509:ExplicitAcceptablePolicySet>
   <tlx509:OID> xsd:string </tlx509:OID> [0..*]
</tlx509:ExplicitAcceptablePolicySet>
```

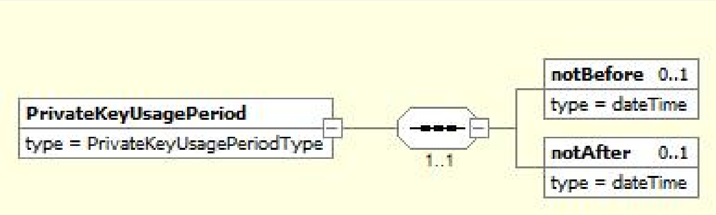
Schema Component Representation

```
<element name="ExplicitAcceptablePolicySet" type="tlx509:ExplicitAcceptablePolicySetType"/>
```

top

Element: PrivateKeyUsagePeriod

Name	PrivateKeyUsagePeriod

Type	tlx509:PrivateKeyUsagePeriodType
Nillable	no
Abstract	no
Documentation	Private key usage period extension Extension PrivateKeyUsagePeriod is not specified in TS 119 612 v2.1.1 and therefore it is specified in this document and shall be included in sub-element "Other" of the element "Service digital identity" specified in TS 119 612 v2.1.1 Clause 5.5.3. The element PrivateKeyUsagePeriod is a service extension with the same semantic as the extension defined in Clause 8.2.2.5 "Private key usage period extension" ITU-T Rec. X.509 or ISO/IEC 9594-8: "This field indicates the period of use of the private key corresponding to the certified public key. It is applicable only for digital signature keys". Validation applications must be aware that more than one cross-certificate with different certificate validity period can be issued for the one key pair and the qualified status interval of the service is usually longer than the certificate usage period or the private key usage period. The private key usage period can be shortened at any time after issuing the certificate and for that reason it is not practical to be a part of the certificate extension. More adequate location is the trusted list service extension which can be regularly updated. The change of the "Private key usage period" extension does not affect the status of the trusted list service and therefore it is not necessary to move actual fields of the service to the history. The validation application shall use only the intersection interval between "private key usage period" and the period in which the status of the service is qualified to determine the time of the validity interval, in which the token must be issued as a valid token according to the trusted list.
Diagram	 PrivateKeyUsagePeriod type = PrivateKeyUsagePeriodType 1..1 notBefore 0..1 type = dateTime notAfter 0..1 type = dateTime

XML Instance Representation

```
<tlx509:PrivateKeyUsagePeriod>
  <tlx509:notBefore> xsd:dateTime </tlx509:notBefore> [0..1]
  <tlx509:notAfter> xsd:dateTime </tlx509:notAfter> [0..1]
</tlx509:PrivateKeyUsagePeriod>
```

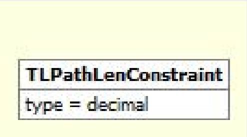
Schema Component Representation

```
<element name="PrivateKeyUsagePeriod" type="tlx509:PrivateKeyUsagePeriodType"/>
```

top

Element: TLPathLenConstraint

Name	TLPathLenConstraint
Type	decimal
Nillable	no
Abstract	no
Documentation	The TLPathLenConstraint is not specified in TS 119 612 v2.1.1 and therefore it is specified in this document and shall be included in sub-element "Other" of the element "Service digital identity" specified in TS 119 612 v2.1.1 Clause 5.5.3. The content of the TLPathLenConstraint element is used as the pathLenConstraint field defined in Clause 8.4.2.1 "Basic constraints extension" and used in Clause 10.5.1 "Basic certificate checks", for more details see Clause 10 "Certification path processing" of ITU-T Rec. X.509 or ISO/IEC 9594-8 and Section 2.5 "Certification Path Controls" of Trust Anchor Format RFC 5914

Diagram	

XML Instance Representation

```
<tlx509:TLPathLenConstraint> decimal </tlx509:TLPathLenConstraint>
```

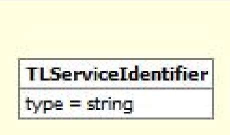
Schema Component Representation

```
<element name="TLPathLenConstraint" type="decimal"/>
```

top

Element: TLServiceIdentifier

Name	TLServiceIdentifier
Type	xsd:string
Nillable	no
Abstract	no
Documentation	The TLServiceIdentifier is not specified in TS 119 612 v2.1.1 and therefore it is specified in this document and shall be used as "TLIxx-y" or "TLIxx-", where "xx" is a Country Code from 'Scheme territory' (TS 119 612 v2.1.1,5.3.10) and "y" is a positive number (service name index) in TL "Name" of "ServiceName" as "(y) z", where the "z" is a service name or when "(y) " is not included in "Name" of "ServiceName" then identifier can be included in sub-element "Other" of the element "Service digital identity" specified in TS 119 612 v2.1.1 Clause 5.5.3. Format: The 'TLServiceIdentifier' field shall contain the string "TLIxx-y" where "xx" is the Country Code included in 'Scheme territory' (see clause 5.3.10 TS 119 612 v2.1.1) and "y" shall contain a positive number as a unique service digital identifier (index) generated according to rules defined in the following part for the 'Id' attribute of tsl:TrustServiceStatusList element. The 'TLServiceIdentifier' field should not be included in 'Service history instance' of 'Service digital identity' (see 5.6.3 TS 119 612 v2.1.1) because the value of 'TLServiceIdentifier' field shall be unique for this service in the whole TL. NOTE 1: When this service is a qualified trust service, the value "TLIxx-y" of TL "Name" of "ServiceName" as "(y) z" or of 'TLServiceIdentifier' field can be associated with graphical elements defined for the EU trust mark in Annexes I and II of the Commission Implementing Regulation (EU) 2015/806 of 22 May 2015 laying down specifications relating to the form of TLISK-4 the EU trust mark for qualified trust services. NOTE 2: When this service is an issuer of the certificate, then issued certificate can contain the value "TLIxx-y" of TL "Name" of "ServiceName" as "(y) z" or of this service 'TLServiceIdentifier' field, as a reference to this service, in the certificate extension Authority Information Access. In this case the accessMethod field contains id-ad-caIssuers (see RFC5280 Section 4.2.2.1). The trusted list service identifier of the issuer service is included in the accessLocation field of GeneralName type as directoryName as a component of X520SerialNumber type. Example: X520SerialNumber = "TLISK-4" Definition of the format of the content included in the 'Id' attribute of tsl:TrustServiceStatusList element. When this 'Id' contains "TLIxx-y" string, then this 'Id' shall contain in "y" the last assigned number included in TL "Name" of "ServiceName" as "(y) z" or the service 'TLServiceIdentifier' field in sub-element "Other" of the element 'Service digital identity' (see clause 5.5.3) of services included in TL, plus one. When a new service is going to be included, the value of this 'Id' is used as a digital identifier for a new including service in TL "Name" of "ServiceName" as "(y) z" or 'TLServiceIdentifier' field in sub-element "Other" of the element 'Service digital identity' (see clause 5.5.3) by automated TL editing tool of the TLSO and then this 'Id' of tsl:TrustServiceStatusList element is incremented. Format: A character string which shall contain "TLIxx-y", where "xx" shall contain Country Codes values (see clause 5.1.5) and "y" shall contain only decimal digits representing a positive number. Value: The string "TLIxx-y" shall contain in "xx" the Country Codes included in 'Scheme territory' (see clause 5.3.10) and "y" shall contain a value of monotonically increasing sequence number representing the last assigned number included in the

	service in TL "Name" of "ServiceName" as "(y) z" or 'TLServiceIdentifier' field in sub-element "Other" of the element 'Service digital identity' (see clause 5.5.3) of services included in TL, plus one. When the TL does not contain services, the value of "y" from "TLIxx-y" shall contain "1".
Diagram	

XML Instance Representation

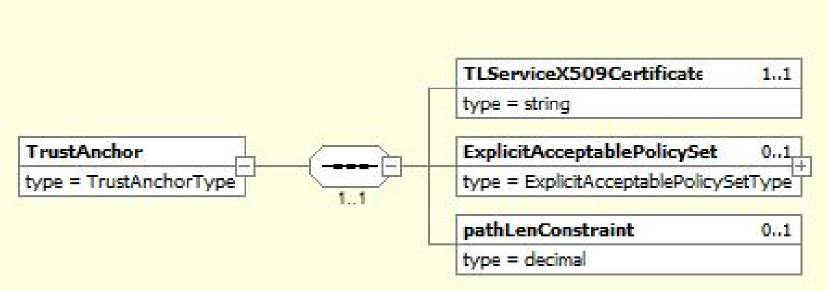
```
<tlx509:TLServiceIdentifier> xsd:string </tlx509:TLServiceIdentifier>
```

Schema Component Representation

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

top

Element: TrustAnchor

Name	TrustAnchor
Type	tlx509:TrustAnchorType
Nillable	no
Abstract	no
Diagram	 TrustAnchor type = TrustAnchorType 1..1 TLServiceX509Certificate 1..1 type = string ExplicitAcceptablePolicySet 0..1 type = ExplicitAcceptablePolicySetType pathLenConstraint 0..1 type = decimal

XML Instance Representation

```
<tlx509:TrustAnchor>
   <tlx509:TLServiceX509Certificate> xsd:base64Binary </tlx509:TLServiceX509Certificate> [1]
   <tlx509:ExplicitAcceptablePolicySet> ... </tlx509:ExplicitAcceptablePolicySet> [0..1]
   <tlx509:pathLenConstraint> xsd:decimal </tlx509:pathLenConstraint> [0..1]
</tlx509:TrustAnchor>
```

Schema Component Representation

```
<element name="TrustAnchor" type="tlx509:TrustAnchorType"/>
```

top

Element: URLContentTypeAndAuthorizedServiceList

Name	URLContentTypeAndAuthorizedServiceList
Type	tlx509:URLContentTypeAndAuthorizedServiceListType
Nillable	no
Abstract	no
Documentation	The extension URLContentTypeAndAuthorizedServiceList is not specified in TS 119 612 v2.1.1 and therefore it is specified in this document and shall be included in "Service information extensions" specified in TS 119 612 v2.1.1 Clause 5.5.9. It specifies, in the form suitable for automated (machine) processing, additional information on one or more URLs where relying parties can obtain the service tokens issued by this service or the service tokens issued by another service authorized by this service e.g. indirect CRL or OCSP response. See Clause 7.10 of ITU-T Rec. X.509 or ISO/IEC 9594-8: "The revocation and a notification of the revocation may be done directly by the same authority that issued the certificate, or indirectly by another authority duly authorized by the authority that issued the certificate."

The content of the URL element of URLContentTypeAndAuthorizedService element can be included in the element specified in TS 119 612 v2.1.1 Clause 5.5.7 "Service supply points" element.
The ContentType element of the URLContentTypeAndAuthorizedService element contains Content-Type of data/protocol provided at URL address in the form defined in Section 3.1.1.5 IETF RFC 7231 for identification of data in the form which is suitable for automated (machine) processing. If the ContentType element is empty, the content on the URL address will provide only human readable information inconvenient for the machine processing.
When the AuthorizedService element is included in the URLContentTypeAndAuthorizedService element then TLServiceIdentifier element contains the service identifier of the other service (or only the TL issuer Country Code part and the other service certificate is stored in the TLServiceX509Certificate element) as a link (shortcut) ensuring that the other service was explicitly authorized by this service for providing the information on tokens issued by this service on the URL address specified in this URL field. The AuthorizedService element shall be included only in case when the URL address is pointing to the other service, e.g. when this service certificate expires, but the status of the issued token must be also provided after the service certificate expiration (see Article 24(4) of Regulation (EU) No 910/2014 "This information shall be made available at least on a per certificate basis at any time and beyond the validity period of the certificate in an automated manner that is reliable, free of charge and efficient.").

The service identifier of the other service included in TLServiceIdentifier element consists of two forms: The first form of the TL service identifier is "TLIxx-y" and the second form is "TLIxx-", where "xx" is the TL issuer’s Country Code (see 5.1.5 TS 119 612 v2.1.1). The form "TLIxx-y" is used only when the value of TL "Name" of "ServiceName" as "(y) z" or 'TLServiceIdentifier' field of the service digital identifier is assigned by the TLSO in the TL sub-element "Other" of the element "Service digital identity" specified in TS 119 612 v2.1.1 Clause 5.5.3 and contains data in the "TLIxx-y" form.
When "TLIxx-" form is used in TLServiceIdentifier element then the element TrustAnchor must contain

TLServiceX509Certificate element and can also contain optional ExplicitAcceptablePolicySet element or pathLenConstraint element as a subset of Trust Anchor Format RFC 5914 fields to achieve simplification and interoperability when the trusted list is used.

The element notBefore of the AuthorizedService element contains the date and the time from which the period of the other authorized service starts.

The optional element notAfter of the AuthorizedService element is present only when the end of the period, in which the other service is authorized by this service, is known.

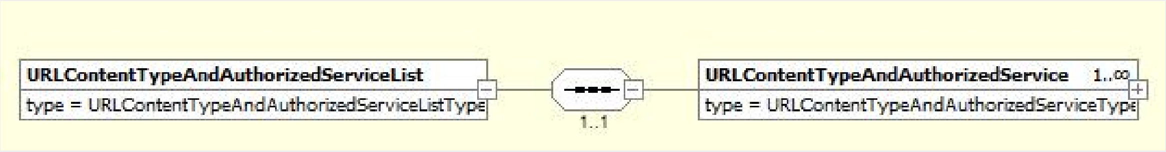
The change of the URLContentTypeAndAuthorizedServiceList extension does not affect the status of the trusted list service and therefore it is not necessary to move the actual fields of the service to the history.

Summary:
The URL address points to the tokens issued by this service or to the tokens issued by the other service, which is authorized to take over some parts of this service, e.g. the other service is issuing verification tokens (indirect CRL or OCSP responses) for certificates issued by this service. When the other service of this TL or the service of another Member State's TL is authorized by this service to provide services related to this service, it shall be indicated by the presence of the AuthorizedService element. The AuthorizedService element contains in TLServiceIdentifier element:
1. the TL service identifiers 'TLIxx-y' as the value included in TL "Name" of "ServiceName" as "(y) z" or the 'TLServiceIdentifier' element in "Other" element of the 'Service digital identities' (TS 119 612 v2.1.1,5.5.3) or
2. when the 'TLServiceIdentifier' element is not present in "Other" element of the 'Service digital identities' (see 5.5.3 TS 119 612 v2.1.1) then TLServiceIdentifier element contains 'TLIxx-' where "xx" is the TL issuer Country Code and in the TLServiceX509Certificate element is X.509 certificate of the authorized service.

NOTE: When this service is the 'certificates creation' service (as defined in Regulation (EU) No 910 2014), this service must store certificates and its certificate status in the database, managed by this service (see Article 24(3) of Regulation (EU) No 910 2014). The certificate status in the form of the verification tokens (like indirect CRL or OCSP responses) is provided from the database of this service by this service or by other service which was authorized to do it by this service.

EXAMPLES of the Content-Type of data/protocol provided by URL:
(Content-Type: text/html;charset=utf-8),
(Content-Type: application/ocsp-request) URL of OCSP protocol for OCSP request for information ".ORQ",
(Content-Type: application/ocsp-response) One pre-produced cryptographically signed response ".ORS" e.g. for TLS server certificate,
(Content-Type: application/pkix-crl) One CRL ".crl",
(Content-Type: application/pkix-cert) One certificate ".cer",
(Content-Type: application/pkcs7-mime) Certificates stored in CMS ".p7c" (for a degenerate SignedData see IETF RFC 5751),
(Content-Type: application/timestamp-query) URL for the time-stamp request in the ASN.1 DER-encoded Time-Stamp message ".TSQ" (see IETF RFC 3161) .

Diagram


URLContentTypeAndAuthorizedServiceList
type = URLContentTypeAndAuthorizedServiceListType
1..1
URLContentTypeAndAuthorizedService 1..∞
type = URLContentTypeAndAuthorizedServiceType

XML Instance Representation

```
<tlx509:URLContentTypeAndAuthorizedServiceList>
  <tlx509:URLContentTypeAndAuthorizedService> tlx509:URLContentTypeAndAuthorizedServiceType
   </tlx509:URLContentTypeAndAuthorizedService> [1..*]
</tlx509:URLContentTypeAndAuthorizedServiceList>
```

Schema Component Representation

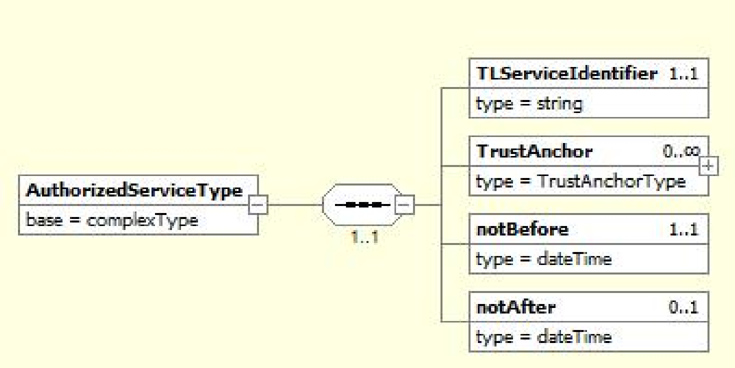
```
<element name="URLContentTypeAndAuthorizedServiceList"
 type="tlx509:URLContentTypeAndAuthorizedServiceListType"/>
```

top

Global Definitions

Complex Type: AuthorizedServiceType

Super-types:	None
Sub-types:	None

Name	AuthorizedServiceType
Abstract	no
Diagram	

XML Instance Representation

```
<...>
  <tlx509:TLServiceIdentifier> xsd:string </tlx509:TLServiceIdentifier> [1]
  <tlx509:TrustAnchor> ... </tlx509:TrustAnchor> [0..*]
  <tlx509:notBefore> xsd:dateTime </tlx509:notBefore> [1]
  <tlx509:notAfter> xsd:dateTime </tlx509:notAfter> [0..1]
</...>
```

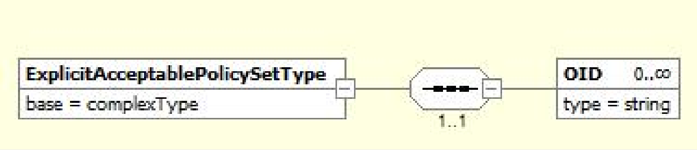
Schema Component Representation

```
<complexType name="AuthorizedServiceType">
  <sequence>
    <element name="TLServiceIdentifier" type="xsd:string"/>
    <element ref="tlx509:TrustAnchor" minOccurs="0" maxOccurs="unbounded"/>
    <element name="notBefore" type="xsd:dateTime"/>
    <element name="notAfter" type="xsd:dateTime" minOccurs="0"/>
  </sequence>
</complexType>
```

top

Complex Type: ExplicitAcceptablePolicySetType

Super-types:	None
Sub-types:	None

Name	ExplicitAcceptablePolicySetType
Abstract	no
Diagram	 ExplicitAcceptablePolicySetType base = complexType 1..1 OID 0..∞ type = string

XML Instance Representation

```
<...>
   <tlx509:OID> xsd:string </tlx509:OID> [0..*]
</...>
```

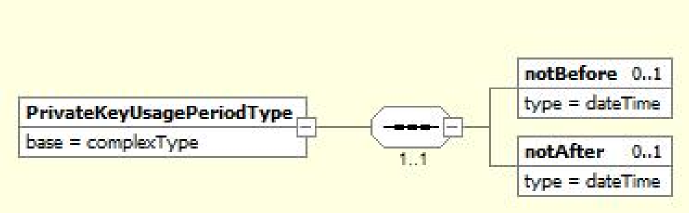
Schema Component Representation

```
<complexType name="ExplicitAcceptablePolicySetType">
   <sequence>
      <element name="OID" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
   </sequence>
</complexType>
```

top

Complex Type: PrivateKeyUsagePeriodType

Super-types:	None
Sub-types:	None

Name	PrivateKeyUsagePeriodType
Abstract	no
Diagram	 PrivateKeyUsagePeriodType base = complexType 1..1 notBefore 0..1 type = dateTime notAfter 0..1 type = dateTime

XML Instance Representation

```
<...>
   <tlx509:notBefore> xsd:dateTime </tlx509:notBefore> [0..1]
   <tlx509:notAfter> xsd:dateTime </tlx509:notAfter> [0..1]
</...>
```

Schema Component Representation

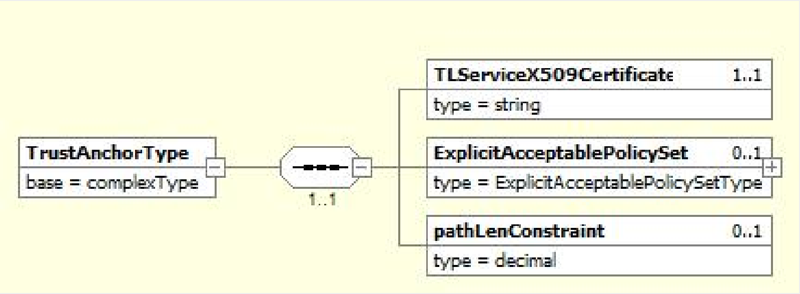
```
<complexType name="PrivateKeyUsagePeriodType">
   <sequence>
      <element name="notBefore" type="xsd:dateTime" minOccurs="0"/>
      <element name="notAfter" type="xsd:dateTime" minOccurs="0"/>
   </sequence>
</complexType>
```

top

Complex Type: TrustAnchorType

Super-types:	None
Sub-types:	None

Name	TrustAnchorType
Abstract	no
Diagram	


TrustAnchorType
base = complexType
1..1
TLServiceX509Certificate 1..1
type = string
ExplicitAcceptablePolicySet 0..1
type = ExplicitAcceptablePolicySetType
pathLenConstraint 0..1
type = decimal

XML Instance Representation

```
<...>
   <tlx509:TLServiceX509Certificate> xsd:base64Binary </tlx509:TLServiceX509Certificate> [1]
   <tlx509:ExplicitAcceptablePolicySet> ... </tlx509:ExplicitAcceptablePolicySet> [0..1]
   <tlx509:pathLenConstraint> xsd:decimal </tlx509:pathLenConstraint> [0..1]
</...>
```

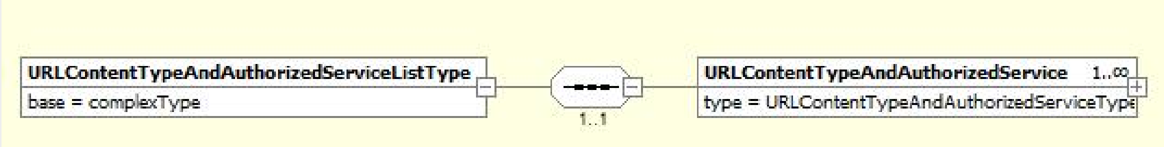
Schema Component Representation

```
<complexType name="TrustAnchorType">
   <sequence>
      <element name="TLServiceX509Certificate" type="xsd:base64Binary"/>
      <element ref="tlx509:ExplicitAcceptablePolicySet" minOccurs="0"/>
      <element name="pathLenConstraint" type="xsd:decimal" minOccurs="0"/>
   </sequence>
</complexType>
```

top

Complex Type: URLContentTypeAndAuthorizedServiceListType

Super-types:	None
Sub-types:	None

Name	URLContentTypeAndAuthorizedServiceListType
Abstract	no
Diagram	

URLContentTypeAndAuthorizedServiceListType
base = complexType
1..1
URLContentTypeAndAuthorizedService 1..∞
type = URLContentTypeAndAuthorizedServiceType

XML Instance Representation

```
<...>
   <tlx509:URLContentTypeAndAuthorizedService> tlx509:URLContentTypeAndAuthorizedServiceType
    </tlx509:URLContentTypeAndAuthorizedService> [1..*]
</...>
```

Schema Component Representation

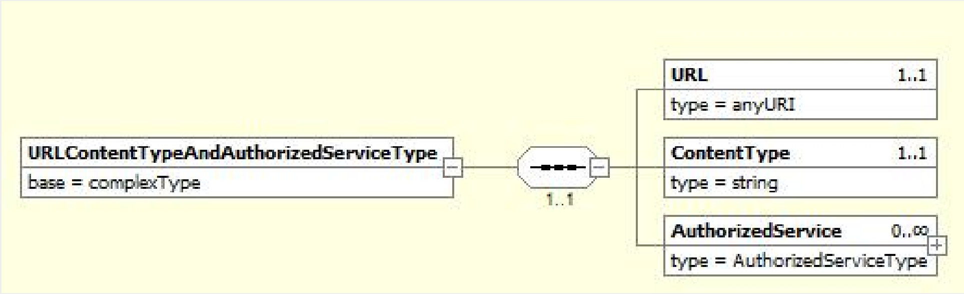
```
<complexType name="URLContentTypeAndAuthorizedServiceListType">
   <sequence>
      <element name="URLContentTypeAndAuthorizedService"
       type="tlx509:URLContentTypeAndAuthorizedServiceType" maxOccurs="unbounded"/>
   </sequence>
</complexType>
```

top

Complex Type: URLContentTypeAndAuthorizedServiceType

Super-types:	None
Sub-types:	None

Name	URLContentTypeAndAuthorizedServiceType
Abstract	no
Diagram	



XML Instance Representation

```
<...>
   <tlx509:URL> xsd:anyURI </tlx509:URL> [1]
   <tlx509:ContentType> xsd:string </tlx509:ContentType> [1]
   <tlx509:AuthorizedService> ... </tlx509:AuthorizedService> [0..*]
</...>
```

Schema Component Representation

```
<complexType name="URLContentTypeAndAuthorizedServiceType">
   <sequence>
      <element name="URL" type="xsd:anyURI"/>
      <element name="ContentType" type="xsd:string"/>
      <element ref="tlx509:AuthorizedService" minOccurs="0" maxOccurs="unbounded"/>
   </sequence>
</complexType>
```

top