

Mark and Signed Mark Objects Mapping

Abstract

Domain Name Registries (DNRs) may operate in special modes for certain periods of time, enabling trademark holders to protect their rights during the introduction of a Top-Level Domain (TLD).

One of those special modes of operation is the Sunrise Period. The Sunrise Period allows trademark holders an advance opportunity to register domain names corresponding to their trademarks before names are generally available to the public.

This document describes the format of a mark and a digitally signed mark used by trademark holders for registering domain names during the Sunrise Period of generic Top-Level Domains (gTLDs). Three types of Mark objects are defined in this specification: registered trademarks, court-validated marks, and marks protected by statute or treaty.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc7848>.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Terminology	4
2. Object Description	4
2.1. Holder and Contact Objects	5
2.2. Mark	6
2.3. Signed Mark	10
2.4. Encoded Signed Mark	13
3. Formal Syntax	14
3.1. Signed Mark Schema	14
3.2. Mark Schema	15
4. IANA Considerations	21
5. Security Considerations	22
6. References	22
6.1. Normative References	22
6.2. Informative References	24
Acknowledgements	24
Author's Address	24

1. Introduction

Domain Name Registries (DNRs) may operate in special modes for certain periods of time enabling trademark holders to protect their rights during the introduction of a Top-Level Domain (TLD).

One of those special modes of operation is the Sunrise Period. The Sunrise Period allows trademark holders an advance opportunity to register domain names corresponding to their trademarks before names are generally available to the public.

This specification was defined as part of the development of the ICANN Trademark Clearinghouse (TMCH). The ICANN TMCH is a global repository for trademark data used by DNRs, registrars, and trademark holders during the registration process of domain names.

This document describes a mapping of the common elements found in trademark data. A digitally signed mark format is defined in order to support digital signatures on the mark. Finally, a mapping for encoding the signed mark document is defined.

Three types of Mark objects are defined in this specification: registered trademarks, court-validated marks, and marks protected by statute or treaty.

This specification is intended to be used in the gTLD space, but nothing precludes the use of this format by other entities.

The detailed policy regarding the public key infrastructure (PKI), authorized validators, and other requirements must be defined based on the local policy of the entities using this specification. In the case of gTLDs, the detailed policy regarding the use of this specification is defined in the Rights Protection Mechanism Requirements document (see [ICANN-TMCH]), and the PKI is defined in [TMCH]. Implementations will need to implement such a PKI (or an equivalent) in order for the signatures defined in this document to have any useful semantics.

The objects specified in this document can be referenced by application protocols like the Extensible Provisioning Protocol (EPP), defined in [RFC5730].

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

XML (EXtensible Markup Language) is case sensitive. Unless stated otherwise, XML specifications and examples provided in this document MUST be interpreted in the character case presented in order to develop a conforming implementation.

"signedMark-1.0" is used as an abbreviation for "urn:ietf:params:xml:ns:signedMark-1.0". The XML namespace prefix "smd" is used, but implementations MUST NOT depend on it and instead employ a proper namespace-aware XML parser and serializer to interpret and output the XML documents.

"mark-1.0" is used as an abbreviation for "urn:ietf:params:xml:ns:mark-1.0". The XML namespace prefix "mark" is used, but implementations MUST NOT depend on it and instead employ a proper namespace-aware XML parser and serializer to interpret and output the XML documents.

2. Object Description

This section defines the Mark and Signed Mark objects. Empty complex element types and abstract elements are defined to support additional definitions using XML schema substitution groups. Support for replacement through the XML schema substitution groups is included in the description of the objects.

This section defines some elements as OPTIONAL. If an element is not defined as OPTIONAL, then it MUST be included in the object.

The following elements are defined as telephone numbers: <mark:voice>, <mark:fax>, and <smd:voice>. The representation of telephone numbers in this specification is derived from structures defined in [ITU.E164.2005]. Telephone numbers described in this mapping are character strings that MUST begin with a plus sign ("+", ASCII value 0x002B), followed by a country code defined in [ITU.E164.2005], followed by a dot (".", ASCII value 0x002E), followed by a sequence of digits representing the telephone number. An optional "x" attribute is provided to note telephone extension information.

The following elements are defined as email addresses: <mark:email> and <smd:email>. Email address syntax is defined in [RFC5322].

2.1. Holder and Contact Objects

Marks are linked to Holder objects and optionally linked to Contact objects. This section defines the `<mark:holder>` and `<mark:contact>` objects.

o The child elements of `<mark:holder>` include:

- * A `<mark:name>` element that contains the name of the individual holder of the mark. At least one of `<mark:name>` and `<mark:org>` MUST be specified, and `<mark:name>` is OPTIONAL if `<mark:org>` is specified.
- * A `<mark:org>` element that contains the name of the organization holder of the mark. At least one of `<mark:name>` and `<mark:org>` MUST be specified, and `<mark:org>` is OPTIONAL if `<mark:name>` is specified.
- * A `<mark:addr>` element that contains the address information of the holder of a mark. A `<mark:addr>` contains the following child elements:
 - + One, two, or three OPTIONAL `<mark:street>` elements that contain the holder's street address.
 - + A `<mark:city>` element that contains the holder's city.
 - + An OPTIONAL `<mark:sp>` element that contains the holder's state or province.
 - + An OPTIONAL `<mark:pc>` element that contains the holder's postal code.
 - + A `<mark:cc>` element that contains the holder's country code. This is a two-character code from [ISO3166-2].
- * An OPTIONAL `<mark:voice>` element that contains the holder's voice telephone number.
- * An OPTIONAL `<mark:fax>` element that contains the holder's facsimile telephone number.
- * An OPTIONAL `<mark:email>` element that contains the email address of the holder.

- o The child elements of `<mark:contact>` include:
 - * A `<mark:name>` element that contains the name of the responsible person.
 - * An OPTIONAL `<mark:org>` element that contains the name of the organization of the contact.
 - * A `<mark:addr>` element that contains the address information of the contact. A `<mark:addr>` contains the following child elements:
 - + One, two, or three OPTIONAL `<mark:street>` elements that contain the contact's street address.
 - + A `<mark:city>` element that contains the contact's city.
 - + An OPTIONAL `<mark:sp>` element that contains the contact's state or province.
 - + An OPTIONAL `<mark:pc>` element that contains the contact's postal code.
 - + A `<mark:cc>` element that contains the contact's country code. This is a two-character code from [ISO3166-2].
 - * A `<mark:voice>` element that contains the contact's voice telephone number.
 - * An OPTIONAL `<mark:fax>` element that contains the contact's facsimile telephone number.
 - * A `<mark:email>` element that contains the contact's email address.

2.2. Mark

A `<mark:mark>` element that describes an applicant's prior right to a given domain name.

A `<mark:mark>` element substitutes for the `<mark:abstractMark>` abstract element to define a concrete definition of a mark. The `<mark:abstractMark>` element can be replaced by other mark definitions using the XML schema substitution groups feature.

The child elements of the `<mark:mark>` element include:

One or more `<mark:trademark>`, `<mark:treatyOrStatute>`, and `<mark:court>` elements that contain the detailed information of marks.

- o A `<mark:trademark>` element that contains the following child elements:
 - * A `<mark:id>` that uniquely identifies a mark in relation to a repository of marks potentially maintained by more than one issuer. A `<mark:id>` value is a concatenation of the local identifier, followed by a hyphen ("-", ASCII value 0x002D), followed by the issuer identifier.
 - * A `<mark:markName>` element that contains the mark text string.
 - * One or more `<mark:holder>` elements that contain the information of the holder of the mark. An "entitlement" attribute is used to identify the entitlement of the holder; possible values are "owner", "assignee", and "licensee".
 - * Zero or more OPTIONAL `<mark:contact>` elements that contain the information of the representative of the mark registration. A "type" attribute is used to identify the type of contact; possible values are "owner", "agent", or "thirdparty".
 - * A `<mark:jurisdiction>` element that contains the two-character code of the jurisdiction where the trademark was registered. This is a two-character code from [WIPO.ST3].
 - * Zero or more OPTIONAL `<mark:class>` elements that contain the World Intellectual Property Organization (WIPO) Nice Classification class numbers of the mark as defined in the WIPO Nice Classification [WIPO-NICE-CLASSES].
 - * Zero or more OPTIONAL `<mark:label>` elements that contain the A-label form (as defined in [RFC5890]) of the label that corresponds to the `<mark:markName>`.
 - * A `<mark:goodsAndServices>` element that contains the full description of the goods and services from the document certifying the registration of the mark.
 - * An OPTIONAL `<mark:apId>` element that contains the trademark application ID registered in the trademark office.
 - * An OPTIONAL `<mark:apDate>` element that contains the date the trademark was applied for.

- * A <mark:regNum> element that contains the trademark registration number registered in the trademark office.
- * A <mark:regDate> element that contains the date the trademark was registered.
- * An OPTIONAL <mark:exDate> element that contains the expiration date of the trademark.
- o A <mark:treatyOrStatute> element that contains the following child elements:
 - * A <mark:id> element; see definition in the <mark:trademark> section above.
 - * A <mark:markName> element; see definition in the <mark:trademark> section above.
 - * One or more <mark:holder> elements; see definition in the <mark:trademark> section above.
 - * Zero or more OPTIONAL <mark:contact> elements; see definition in the <mark:trademark> section above.
 - * One or more <mark:protection> elements that contain the countries and region of the country where the mark is protected. The <mark:protection> element contains the following child elements:
 - + A <mark:cc> element that contains the two-character code of the country in which the mark is protected. This is a two-character code from [ISO3166-2].
 - + An OPTIONAL <mark:region> element that contains the name of a city, state, province, or other geographic region of <mark:country> in which the mark is protected.
 - + Zero or more OPTIONAL <mark:ruling> elements that contain the two-character code of the national territory in which the statute or treaty is applicable. This is a two-character code from [ISO3166-2].
 - + Zero or more OPTIONAL <mark:label> elements; see definition in the <mark:trademark> section above.
 - * A <mark:goodsAndServices> element; see definition in the <mark:trademark> section above.

- * A <mark:refNum> element that contains the serial number of the mark.
- * A <mark:proDate> element that contains the date of protection of the mark.
- * A <mark:title> element that contains the title of the treaty or statute.
- * A <mark:execDate> element that contains the execution date of the treaty or statute.
- o A <mark:court> element that contains the following child elements:
 - * A <mark:id> element; see definition in the <mark:trademark> section above.
 - * A <mark:markName> element; see definition in the <mark:trademark> section above.
 - * One or more <mark:holder> elements; see definition in the <mark:trademark> section above.
 - * Zero or more OPTIONAL <mark:contact> elements; see definition in the <mark:trademark> section above.
 - * Zero or more OPTIONAL <mark:label> elements; see definition in the <mark:trademark> section above.
 - * A <mark:goodsAndServices> element; see definition in the <mark:trademark> section above.
 - * A <mark:refNum> element that contains the reference number of the court's opinion.
 - * A <mark:proDate> element that contains the date of protection of the mark.
 - * A <mark:cc> element that contains the two-character code of the country where the court is located. This is a two-character code from [ISO3166-2].
 - * Zero or more OPTIONAL <mark:region> elements that contain the name of a city, state, province, or other geographic region of <mark:cc> in which the mark is protected. In case <mark:region> is specified, a default-deny approach MUST be assumed regarding the regions of a country.

- * A <mark:courtName> element that contains the name of the court.

2.3. Signed Mark

The <smd:signedMark> is a digitally signed XML document using an XML Signature [XMLDSIG]. The <smd:signedMark> XML document (SMD) includes a required "id" attribute of type XML Schema Definition (XSD) ID for use with an IDREF URI from the Signature element. The SMD might be transmitted as part of a protocol already based on XML; therefore, exclusive XML canonicalization as defined in [XMLC14N] MUST be used.

A <smd:signedMark> element substitutes for the <smd:abstractSignedMark> abstract element to define a concrete definition of a signed mark. The <smd:abstractSignedMark> element can be replaced by other signed mark definitions using the XML schema substitution groups feature.

The child elements of the <smd:signedMark> element include:

- o The <smd:id> element that uniquely identifies an SMD in relation to a repository of SMDs potentially maintained by more than one issuer. The <smd:id> value is a concatenation of the local identifier, followed by a hyphen ("-", ASCII value 0x002D), followed by the issuer identifier.
- o A <smd:issuerInfo> element that contains the information of the issuer of the mark registration. An "issuerID" attribute is used to specify the issuer identifier. The child elements include:
 - * A <smd:org> element that contains the organization name of the issuer.
 - * A <smd:email> element that contains the issuer customer support email address.
 - * An OPTIONAL <smd:url> element that contains the HTTP or HTTPS URL of the issuer's site.
 - * An OPTIONAL <smd:voice> element that contains the issuer's voice telephone number.
- o A <smd:notBefore> element that contains the creation date and time of the SMD.
- o A <smd:notAfter> element that contains the expiration date and time of the SMD.

- o A <mark:mark> element that contains the mark information as defined in Section 2.2.

The following is an example of an SMD:

```
<?xml version="1.0" encoding="UTF-8"?>
<smd:signedMark xmlns:smd="urn:ietf:params:xml:ns:signedMark-1.0"
id="smd1">
  <smd:id>0000001751376056503931-65535</smd:id>
  <smd:issuerInfo issuerID="65535">
    <smd:org>ICANN TMCH TESTING TMV</smd:org>
    <smd:email>notavailable@example.com</smd:email>
    <smd:url>https://www.example.com</smd:url>
    <smd:voice>+32.000000</smd:voice>
  </smd:issuerInfo>
  <smd:notBefore>2013-08-09T13:55:03.931Z</smd:notBefore>
  <smd:notAfter>2017-07-23T22:00:00.000Z</smd:notAfter>
  <mark:mark xmlns:mark="urn:ietf:params:xml:ns:mark-1.0">
    <mark:trademark>
      <mark:id>00052013734689731373468973-65535</mark:id>
      <mark:markName>Test & Validate</mark:markName>
      <mark:holder entitlement="owner">
        <mark:org>Ag corporation</mark:org>
        <mark:addr>
          <mark:street>1305 Bright Avenue</mark:street>
          <mark:city>Arcadia</mark:city>
          <mark:sp>CA</mark:sp>
          <mark:pc>90028</mark:pc>
          <mark:cc>US</mark:cc>
        </mark:addr>
      </mark:holder>
      <mark:contact type="agent">
        <mark:name>Tony Holland</mark:name>
        <mark:org>Ag corporation</mark:org>
        <mark:addr>
          <mark:street>1305 Bright Avenue</mark:street>
          <mark:city>Arcadia</mark:city>
          <mark:sp>CA</mark:sp>
          <mark:pc>90028</mark:pc>
          <mark:cc>US</mark:cc>
        </mark:addr>
        <mark:voice>+1.2025562302</mark:voice>
        <mark:fax>+1.2025562301</mark:fax>
        <mark:email>info@agcorporation.com</mark:email>
      </mark:contact>
      <mark:jurisdiction>US</mark:jurisdiction>
      <mark:class>15</mark:class>
      <mark:label>testandvalidate</mark:label>
    </mark:mark>
  </smd:signedMark>
</smd>
```

```

    <mark:label>test---validate</mark:label>
    <mark:label>testand-validate</mark:label>
    <mark:label>test-et-validate</mark:label>
    <mark:label>test-validate</mark:label>
    <mark:label>test--validate</mark:label>
    <mark:label>test-etvalidate</mark:label>
    <mark:label>testetvalidate</mark:label>
    <mark:label>testvalidate</mark:label>
    <mark:label>testet-validate</mark:label>
    <mark:goodsAndServices>guitar</mark:goodsAndServices>
    <mark:regNum>1234</mark:regNum>
    <mark:regDate>2012-12-31T23:00:00.000Z</mark:regDate>
  </mark:trademark>
</mark:mark>
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
  <SignedInfo>
    <CanonicalizationMethod
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    <SignatureMethod
Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
    <Reference URI="#smd1">
      <Transforms>
        <Transform
Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
      </Transforms>
      <DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
      <DigestValue>wgyW3nZPoEfptlhRILKnOQnbdU6ArM7ShrAfHgDFg=</DigestValue>
    </Reference>
  </SignedInfo>
  <SignatureValue>
jMu4PfyQGiJBF0GWSEPFcJjmywCEqR2h4LD+ge6XQ+JnmKFFCuCZS/3SLKAX0Llw
QDFO2e0Y69k2G7/LGE37X3vOflobFMloGwja8+GMVraoto5xAd4/AF7eHukgAymD
o9toxo2h0yV4A4PmXzsU6S86XtCcUE+S/WM72nyn47zoUCzzPKHZBryeWehVFQ+
jYRMIAmzM57HHQA+6eaXefRvtPETgu04aVIVSugc40UAZZwbYcZrC6w0aQqqqAZi
30aPOBYbAvHMSmWSS+hFkbshomJfHxb97TD2grlYnrQIzqXk7WbHWy2SYdA+sI/Z
ipJsXNa6ostUw1CzA7jfwA==
  </SignatureValue>
  <KeyInfo>
    <X509Data>
      <X509Certificate>
MIIESTCCAzGgAwIBAgIBAJANBgkqhkiG9w0BAQsFADBiMQswCQYDVQQGEwJVUzEL
MAKGA1UECBMCQ0ExFDASBgNVBACTC0xvcyBBbmdlbGVzMRMwEQYDVQQKEwpJQ0FO
TiBUTUNIMRswGQYDVQQDEExJJQ0FOTiBUTUNIIFRFU1QgQ0EwHhcNMjMwMjA4MDAw
MDAwWhcNMjMwMjA4MDAwWjBzMQswCQYDVQQGEwJVUzELMAKGA1UECBMCQ0Ex
FDASBgNVBACTC0xvcyBBbmdlbGVzMRcwFQYDVQQKEw5WYWxpZGF0b3IgVE1DSDEh
MB8GA1UEAxMYVmFsaWRhdG9yIFRNQ0ggVEVTVCBDRVJUMIIBIjANBgkqhkiG9w0B
AQEFAAOCAQ8AMIIBCgKCAQEAO/cwvXhbVYl0RDWWvOyeZpETVZVVcMCovUVNg/sw
      </X509Certificate>
    </X509Data>
  </KeyInfo>
</Signature>

```

```

WinuMgEWgVQFrz0xA04pEhXCFVv4evbUpekJ5buqU1gmQyOsCKQ1hOHTdPjvkC5u
pDqa51Flk0TMaMkIQjs7aUKCmA4RG4tTTGK/EjR1ix8/D0gHYVRldy1YPrMP+ou7
5bOVnIos+HifrAtrIv4qEqwLL4FTZAUPaCa2BmgXfy2CSRQbxD5OrlgcSa3vurh5
sPMCnxqaXmIXmQipS+DuEBqMM8tldaN7RYojUEKRGVsNk5i9y2/7sjnlzzyUPf7v
L4GgDYqhJYWV61DnXgx/Jd6CWxvsndF6scscQzUTel+hywIDAQABo4H/MIH8MAwG
AlUdEwEB/wQCMAAwHQYDVR0OBBYEFpZEcIQcD/Bj2IFz/LEruo2ADJviMIGMBgNV
HSMEgYQwgYGAFO0/7kEh3FuEKS+Q/kYHaD/W6wihoWakZDBiMQswCQYDVQQGEwJV
UzELMAkGA1UECBMCQ0ExFDASBgNVBAcTC0xvcyBBbmdlbGVzMRRMwEQYDVQQKEwPJ
Q0FOTiBUTUNIMRswGQYDVQQDExJJQ0FOTiBUTUNIIFRFU1QgQ0GCAQEwDgYDVVR0P
AQH/BAQDAgeAMC4GAlUdHwQnMCUwI6AhoB+GHWh0dHA6Ly9jcmwuaWNhbm4ub3Jn
L3RtY2guY3JsMA0GCSqGSIb3DQEBCwUAA4IBAQB2qSy7ui+43cebKUKwWPrzz9y/
IkrMeJGKjo40n+9uekaw3DJ5EqiOf/qZ4pjbD++oR6BJCb6NQuQKwnoAz5lE4Ssu
y5+i93oT3HfyVc4gNMIOHm1PS19l7DBKrbwbzAea/0jKWVzrvvmV7TBfjxD3AQo1R
bU5dBr6IjbdLFlno5x0G0mrG7x5OUPuurihyiURpFDpwh8KAHlwMcCpXGXFRTGKk
wydgyVYAty7otkl/z3bZkCVT34gPvF70sR6+QxUy8u0LzF5A/beYaZpxSYG3lamL
AdXittTWFipaIGea9lEGFM0L9+Bg7XzNn4nVLXokyEB3bgS4scG6QznX23FGk
</X509Certificate>
</X509Data>
</KeyInfo>
</Signature>
</smd:signedMark>

```

NOTE: The example shown above includes white space for indentation purposes. It is RECOMMENDED that SMDs do not include white space between the XML elements, in order to mitigate risks of invalidating the digital signature when transferring of SMDs between applications takes place.

2.4. Encoded Signed Mark

The <smd:encodedSignedMark> element contains an encoded form of an SMD (described in Section 2.3), with the encoding defined by the "encoding" attribute with the default "encoding" value of "base64" [RFC4648].

The following is an example of a <smd:encodedSignedMark> element that uses the default "base64" for encoding a <smd:signedMark> element.

```

<smd:encodedSignedMark
  xmlns:smd="urn:ietf:params:xml:ns:signedMark-1.0">
PD94bWwgdmVyc2lvcj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4KPHNtZDpzaWduZWRYXJ
rIHhtbG5zOnNtZD0idXJuOmllldGY6cGFyYW1zOnhtbDpuczpzaWduZWRYXJrLTEuMCIGA
... (base64 data elided for brevity) ...
PC9zbWQ6c2lnbmVktWFyaz4=
</smd:encodedSignedMark>

```

3. Formal Syntax

Two schemas are presented here. The first schema is the schema for the Signed Mark object. The second schema is the schema for the Mark object.

The formal syntax presented here is a complete schema representation of the object mapping suitable for automated validation of EPP XML instances. The BEGIN and END tags are not part of the schema; they are used to note the beginning and ending of the schema for URI registration purposes.

3.1. Signed Mark Schema

BEGIN

```
<?xml version="1.0" encoding="UTF-8"?>
<schema
  targetNamespace="urn:ietf:params:xml:ns:signedMark-1.0"
  xmlns:smd="urn:ietf:params:xml:ns:signedMark-1.0"
  xmlns:mark="urn:ietf:params:xml:ns:mark-1.0"
  xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <annotation>
    <documentation>
      Schema for representing a Signed Mark object.
    </documentation>
  </annotation>

  <import namespace="urn:ietf:params:xml:ns:mark-1.0" />
  <import namespace="http://www.w3.org/2000/09/xmldsig#" />

  <!--
Abstract Signed Mark object for replacement via substitution.
-->
  <element name="abstractSignedMark" type="smd:abstractSignedMarkType"
    abstract="true"/>

  <!--
Empty type for use in extending for a Signed Mark object.
-->
  <complexType name="abstractSignedMarkType"/>

  <element name="signedMark" type="smd:signedMarkType"
    substitutionGroup="smd:abstractSignedMark"/>

  <element name="encodedSignedMark" type="smd:encodedSignedMarkType"/>
```

```

<complexType name="signedMarkType">
  <complexContent>
    <extension base="smd:abstractSignedMarkType">
      <sequence>
        <element name="id" type="mark:idType"/>
        <element name="issuerInfo" type="smd:issuerInfoType"/>
        <element name="notBefore" type="dateTime"/>
        <element name="notAfter" type="dateTime"/>
        <element ref="mark:abstractMark"/>
        <element ref="dsig:Signature"/>
      </sequence>
      <attribute name="id" type="ID" use="required"/>
    </extension>
  </complexContent>
</complexType>

<complexType name="issuerInfoType">
  <sequence>
    <element name="org" type="token"/>
    <element name="email" type="mark:minTokenType"/>
    <element name="url" type="token" minOccurs="0"/>
    <element name="voice" type="mark:el64Type" minOccurs="0"/>
  </sequence>
  <attribute name="issuerID" type="token" use="required"/>
</complexType>

<complexType name="encodedSignedMarkType">
  <simpleContent>
    <extension base="token">
      <attribute name="encoding" type="token" default="base64"/>
    </extension>
  </simpleContent>
</complexType>
</schema>
END

```

3.2. Mark Schema

```

BEGIN
<?xml version="1.0" encoding="UTF-8"?>
<schema
  targetNamespace="urn:ietf:params:xml:ns:mark-1.0"
  xmlns:mark="urn:ietf:params:xml:ns:mark-1.0"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <annotation>
    <documentation>

```

```
    Schema for representing a Trademark, also referred to
    as Mark.
  </documentation>
</annotation>

<!--
Abstract Mark object for replacement via substitution.
-->
<element name="abstractMark" type="mark:abstractMarkType"
  abstract="true"/>

<!--
<mark:mark> element definition
-->
<element name="mark" type="mark:markType"
  substitutionGroup="mark:abstractMark"/>

<!--
Empty type for use in extending for a Mark object.
-->
<complexType name="abstractMarkType"/>

<!--
<mark:mark> child elements
-->
<complexType name="markType">
  <complexContent>
    <extension base="mark:abstractMarkType">
      <sequence>
        <element name="trademark" type="mark:trademarkType"
          minOccurs="0" maxOccurs="unbounded"/>
        <element name="treatyOrStatute"
          type="mark:treatyOrStatuteType" minOccurs="0"
          maxOccurs="unbounded"/>
        <element name="court" type="mark:courtType" minOccurs="0"
          maxOccurs="unbounded"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="holderType">
  <sequence>
    <element name="name" type="token" minOccurs="0"/>
    <element name="org" type="token" minOccurs="0"/>
    <element name="addr" type="mark:addrType"/>
    <element name="voice" type="mark:el64Type" minOccurs="0"/>
    <element name="fax" type="mark:el64Type" minOccurs="0"/>
  </sequence>
</complexType>
```



```
<element name="email" type="mark:minTokenType" minOccurs="0"/>
</sequence>
<attribute name="entitlement" type="mark:entitlementType"/>
</complexType>

<complexType name="contactType">
  <sequence>
    <element name="name" type="token"/>
    <element name="org" type="token" minOccurs="0"/>
    <element name="addr" type="mark:addrType"/>
    <element name="voice" type="mark:e164Type"/>
    <element name="fax" type="mark:e164Type" minOccurs="0"/>
    <element name="email" type="mark:minTokenType"/>
  </sequence>
  <attribute name="type" type="mark:contactTypeType"/>
</complexType>

<complexType name="trademarkType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="markName" type="token"/>
    <element name="holder" type="mark:holderType"
      maxOccurs="unbounded" />
    <element name="contact" type="mark:contactType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="jurisdiction" type="mark:ccType"/>
    <element name="class" type="integer" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="label" type="mark:labelType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="goodsAndServices" type="token" />
    <element name="apId" type="token" minOccurs="0"/>
    <element name="apDate" type="dateTime" minOccurs="0"/>
    <element name="regNum" type="token"/>
    <element name="regDate" type="dateTime"/>
    <element name="exDate" type="dateTime" minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="treatyOrStatuteType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="markName" type="token"/>
    <element name="holder" type="mark:holderType"
      maxOccurs="unbounded" />
    <element name="contact" type="mark:contactType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="protection" type="mark:protectionType">
```

```
        maxOccurs="unbounded"/>
    <element name="label" type="mark:labelType" minOccurs="0"
        maxOccurs="unbounded"/>
    <element name="goodsAndServices" type="token" />
    <element name="refNum" type="token"/>
    <element name="proDate" type="dateTime"/>
    <element name="title" type="token"/>
    <element name="execDate" type="dateTime"/>
</sequence>
</complexType>

<complexType name="courtType">
    <sequence>
        <element name="id" type="mark:idType"/>
        <element name="markName" type="token"/>
        <element name="holder" type="mark:holderType"
            maxOccurs="unbounded" />
        <element name="contact" type="mark:contactType" minOccurs="0"
            maxOccurs="unbounded"/>
        <element name="label" type="mark:labelType" minOccurs="0"
            maxOccurs="unbounded"/>
        <element name="goodsAndServices" type="token" />
        <element name="refNum" type="token"/>
        <element name="proDate" type="dateTime"/>
        <element name="cc" type="mark:ccType"/>
        <element name="region" type="token" minOccurs="0"
            maxOccurs="unbounded"/>
        <element name="courtName" type="token"/>
    </sequence>
</complexType>

<!--
Address (<mark:addr>) child elements
-->
<complexType name="addrType">
    <sequence>
        <element name="street" type="token" minOccurs="1" maxOccurs="3"/>
        <element name="city" type="token"/>
        <element name="sp" type="token" minOccurs="0"/>
        <element name="pc" type="mark:pcType" minOccurs="0"/>
        <element name="cc" type="mark:ccType"/>
    </sequence>
</complexType>

<!--
<mark:protection> child elements
-->
<complexType name="protectionType">
```

```
<sequence>
  <element name="cc" type="mark:ccType"/>
  <element name="region" type="token" minOccurs="0"/>
  <element name="ruling" type="mark:ccType"
    minOccurs="0" maxOccurs="unbounded"/>
</sequence>
</complexType>

<!--
Postal code definition
-->
<simpleType name="pcType">
  <restriction base="token">
    <maxLength value="16"/>
  </restriction>
</simpleType>

<!--
Country code definition
-->
<simpleType name="ccType">
  <restriction base="token">
    <length value="2"/>
  </restriction>
</simpleType>

<!--
Phone number with extension definition
-->
<complexType name="e164Type">
  <simpleContent>
    <extension base="mark:e164StringType">
      <attribute name="x" type="token"/>
    </extension>
  </simpleContent>
</complexType>

<!--
Phone number with extension definition
-->
<simpleType name="e164StringType">
  <restriction base="token">
    <pattern value="(\+[0-9]{1,3}\.[0-9]{1,14})?" />
    <maxLength value="17"/>
  </restriction>
</simpleType>

<!--
```

```
Id type definition
-->
<simpleType name="idType">
  <restriction base="token">
    <pattern value="\d+-\d+"/>
  </restriction>
</simpleType>

<!--
DNS label type definition
-->
<simpleType name="labelType">
  <restriction base="token">
    <minLength value="1"/>
    <maxLength value="63"/>
    <pattern value="[a-zA-Z0-9]([a-zA-Z0-9\-*[a-zA-Z0-9])?" />
  </restriction>
</simpleType>

<!--
Type used for email addresses
-->
<simpleType name="minTokenType">
  <restriction base="token">
    <minLength value="1"/>
  </restriction>
</simpleType>

<simpleType name="entitlementType">
  <restriction base="token">
    <enumeration value="owner"/>
    <enumeration value="assignee"/>
    <enumeration value="licensee"/>
  </restriction>
</simpleType>

<simpleType name="contactTypeType">
  <restriction base="token">
    <enumeration value="owner"/>
    <enumeration value="agent"/>
    <enumeration value="thirdparty"/>
  </restriction>
</simpleType>
</schema>
END
```

4. IANA Considerations

This document uses URNs to describe XML namespaces and XML schemas conforming to the registry mechanism described in [RFC3688]. IANA has registered two URI assignments: signed mark (signedMark-1.0) and mark (mark-1.0).

- o The signed mark namespace (signedMark-1.0) has been registered in the "ns" registry.

URI: urn:ietf:params:xml:ns:signedMark-1.0

Registrant Contact: IESG

XML: None. Namespace URIs do not represent an XML specification.

- o The signed mark schema (signedMark-1.0) has been registered in the "schema" registry.

URI: urn:ietf:params:xml:schema:signedMark-1.0

Registrant Contact: IESG

XML: See the "Formal Syntax" section of this document.

- o The mark namespace (mark-1.0) has been registered in the "ns" registry.

URI: urn:ietf:params:xml:ns:mark-1.0

Registrant Contact: IESG

XML: None. Namespace URIs do not represent an XML specification.

- o The mark schema (mark-1.0) has been registered in the "schema" registry.

URI: urn:ietf:params:xml:schema:mark-1.0

Registrant Contact: IESG

XML: See the "Formal Syntax" section of this document.

5. Security Considerations

The security of a Signed Mark object depends on the security of the underlying XML Digital Signature (DSIG) algorithms. As such, all the security considerations from [XMLDSIG] apply here as well.

The digital signature algorithm used in Signed Mark objects SHOULD be RSA-SHA256 [RFC6931]. The size of the RSA key SHOULD be at least 2048 bits. A valid reason for choosing something else would be if RSA-SHA256 is deemed to not provide sufficient security.

In the case of the ICANN TMCH, Signed Mark objects use the algorithms for digesting and signing recommended in this document.

Signed Mark objects are used primarily for domain name registrations in gTLDs during the Sunrise Period, but other third parties might be using them. A party using Signed Mark objects should verify that the digital signature is valid based on local policy. In the case of gTLDs, the Rights Protection Mechanism Requirements document [ICANN-TMCH] defines such policy, and the PKI is defined in [TMCH]. Implementations will need to implement such a PKI (or an equivalent) in order for the signatures defined in this document to have any useful semantics.

6. References

6.1. Normative References

[ICANN-TMCH]

ICANN, "Trademark Clearinghouse; Rights Protection Mechanism Requirements", 2013,
<<http://newgtlds.icann.org/en/about/trademark-clearinghouse/rpm-requirements-30sep13-en.pdf>>.

[ISO3166-2]

ISO, "International Standard for country codes and codes for their subdivisions", 2006,
<http://www.iso.org/iso/home/standards/country_codes.htm>.

[ITU.E164.2005]

International Telecommunication Union, "The international public telecommunication numbering plan", ITU-T Recommendation E.164, November 2010,
<<https://www.itu.int/rec/T-REC-E.164-201011-I/en>>.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<http://www.rfc-editor.org/info/rfc3688>>.
- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <<http://www.rfc-editor.org/info/rfc4648>>.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <<http://www.rfc-editor.org/info/rfc5322>>.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, DOI 10.17487/RFC5890, August 2010, <<http://www.rfc-editor.org/info/rfc5890>>.
- [RFC6931] Eastlake 3rd, D., "Additional XML Security Uniform Resource Identifiers (URIs)", RFC 6931, DOI 10.17487/RFC6931, April 2013, <<http://www.rfc-editor.org/info/rfc6931>>.
- [WIPO-NICE-CLASSES] WIPO, "Nice Classification", January 2016, <<http://www.wipo.int/classifications/nice/en>>.
- [WIPO.ST3] WIPO, "Recommended standard on two-letter codes for the representation of states, other entities and intergovernmental organizations", Standard ST.3, February 2015, <<http://www.wipo.int/standards/en/pdf/03-03-01.pdf>>.
- [XMLC14N] W3C, "Exclusive XML Canonicalization Version 1.0", W3C Recommendation, July 2002, <<http://www.w3.org/TR/2002/REC-xml-exc-c14n-20020718>>.
- [XMLDSIG] W3C, "XML Signature Syntax and Processing Version 1.1", W3C Recommendation, April 2013, <<http://www.w3.org/TR/xmlsig-core1>>.

6.2. Informative References

- [EPP-LAUNCH] Gould, J., Tan, W., and G. Brown, "Launch Phase Mapping for the Extensible Provisioning Protocol (EPP)", Work in Progress, draft-ietf-regext-launchphase-00, April 2016.
- [RFC5105] Lendl, O., "ENUM Validation Token Format Definition", RFC 5105, DOI 10.17487/RFC5105, December 2007, <<http://www.rfc-editor.org/info/rfc5105>>.
- [RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009, <<http://www.rfc-editor.org/info/rfc5730>>.
- [TMCH] Lozano, G., "ICANN TMCH functional specifications", Work in Progress, draft-ietf-regext-tmch-func-spec-00, April 2016.

Acknowledgements

Special thanks to Chris Wright for creating the first prototype of an SMD and to James Gould, Wil Tan, and Gavin Brown for creating the mark and SMD definitions in "Launch Phase Mapping for the Extensible Provisioning Protocol (EPP)" [EPP-LAUNCH], on which this document is based. Portions of the Security Considerations section were shamefully copied from RFC 5105 [RFC5105]. The author would like to acknowledge the following individuals for their contributions to this document: Scott Hollenbeck and Jan Jansen.

Author's Address

Gustavo Lozano
ICANN
12025 Waterfront Drive, Suite 300
Los Angeles, CA 90292
United States

Phone: +1.3103015800
Email: gustavo.lozano@icann.org

