

Contact Eligibility EPP Extension (Version 1.1) for the .tld TLD

CORE Association

1. Extension for Contact Eligibility (Version 1.1)

The CORE Registration System provides a proprietary EPP extension for providing contact eligibility information. This document describes Version 1.1 of that EPP extension.

1.1 Introduction

This document describes an extension mapping for version 1.0 of the Extensible Provisioning Protocol (EPP) described in RFC 5730. This mapping is an extension of the contact mapping described in RFC 5733. It is specified using the Extensible Markup Language (XML) and XML Schema notation.

This extension serves the purpose of specifying contact eligibility information, i.e. additional information to be associated with a contact object that allows the registry to determine a contact's (usually the registrant's) eligibility for acquiring or keeping a certain domain name it is associated with. Depending on a registry's policies, the specification of certain contact eligibility information may be mandatory to obtain or retain a specific domain name. For example, a registry may mandate that a domain's registrant is a registered enterprise, and may thus require a registrar to specify the company's company/enterprise ID as contact eligibility information for contacts used as a registrant for the company's domains. Likewise, if a domain's registrant must be a citizen of a certain country or region, the registry may require a registrar to specify the registrant's person ID (as issued by the country or region) as contact eligibility information.

1.1.1 Typed Public IDs

In its version 1.1, the extension allows the specification of one or more ASCII-only **public IDs** associated with the company or person represented by the contact. Each public ID string has a mandatory type, represented by a "type" string whose permissible content is defined by registry policy.

1.1.2 Natural Person Flag

In addition to the public IDs, the extension also allows to specify whether or not the contact represents a **natural person** or not.

1.1.3 Server Policy

The information is inspected by the registry to decide whether the company is eligible to obtain a certain domain name according to registry policy.

The public ID(s) and "natural person" flag may be specified when creating the contact with the <contact:create> EPP command. This information may be changed or removed later using the <contact:update> EPP command. The current public ID(s) and "natural person" flag on file for a contact may be inquired using the <contact:info> command. In addition, if a contact carries any eligibility information, the response to a <contact:info> command will provide information about the contact's validation status.

Depending on the policy of a registry supporting this extension, it may be optional or mandatory to provide eligibility information with a contact object. If eligibility information is optional, the registry policy may prevent registrars from using contacts lacking eligibility information from being used in certain roles within a domain. If a contact has eligibility information and is associated with an active domain in a certain role, the registry policy may prevent the removal of eligibility information from said contact. Registry policy may also restrict the number of specifiable public IDs, e.g. some registries may only allow one public ID per contact.

1.2 EPP Command Mapping

This section deals with the specific command mappings for the EPP extension for contact eligibility information.

In the following, the respective root elements of the extensions are mentioned. If used, they must be placed or expected within the optional <extension> element at the proper location in the XML document representing the EPP command or response, as described in RFC 5730. Note that the use of the "el:" XML namespace prefix is for documentation purposes only. Conforming to the "Namespaces in XML 1.1" standard, EPP and the registry implementation take only the associated namespace URI into account, and not the prefix itself. So actually any prefix or even the default namespace may be used in requests and must be expected in responses.

The eligibility extension is only used in relation to contact objects. It will not occur in commands that are related to domain and host objects.

1.2.1 EPP Query Commands

There are four EPP commands to retrieve object information: <check> to find out whether an object is known to the server, <info> to ask for detailed information associated with an object, <poll> to discover and retrieve queued service messages for individual clients and <transfer> to get transfer status information for an object.

1.2.1.1 EPP <contact:check> Command

This extension does not add any element to the EPP <check> command.

1.2.1.2 EPP <contact:info> Command

The eligibility extension does not provide an element for the info command. Additional elements are defined for the <info> response, though. When an <info> command has been processed successfully for a contact carrying eligibility information, the EPP <extension> element in the response contains a child <el:infData> element that identifies the contact eligibility extension namespace.

The <el:infData> element may contain one or more <el:publicId> elements, which contain the public ID strings currently present for the inquired contact, each with their type contained in a "type" XML attribute.

Also, the <el:infData> element may contain an <el:naturalPerson> element, which contains the contact's "natural person" status as a boolean value ("true" denoting "natural person", "false" denoting "no natural person").

Furthermore, the <el:infData> element may contain an <el:validationStatus> element denoting the contact's validation status, represented by one of these three values:

- **pendingValidation:** a validation of the contact's eligibility data is pending
- **validated:** the contact's eligibility data was successfully validated, i.e. it was found to be valid; the contact may thus be used in successful domain registrations
- **invalid:** the contact's eligibility data failed the validation, i.e. it was found to be invalid; the contact is unsuitable for use in successful domain registrations

An example of an <info> response containing the extension can be found in the "Examples" section below.

1.2.1.3 EPP <poll> Command

This extension does not add any element to the EPP <poll> command.

1.2.1.4 EPP <transfer> Query Command

This extension does not add any element to the EPP <transfer> query command.

1.2.2 EPP Transform Commands

There are four EPP commands to transform contact objects: <create> to create an instance of an object, <delete> to delete an instance of an object, <transfer> to manage object sponsorship changes and <update> to change information associated with an object.

1.2.2.1 EPP <contact:create> Command

The create command, which allows the registration of contact objects, can be augmented by a <el:create> element in the extension section of the command that identifies the contact eligibility extension namespace. If specified, the <el:create> element may contain an <el:publicId> element whose content specifies one of the contact's public IDs, each along with the its type contained in the mandatory "type" attribute. A public ID is a non-empty, ASCII-only string with a maximum length of 30 characters.

Also, the <el:create> element may contain an <el:naturalPerson> element, which must contain the contact's "natural person" status as a boolean value ("true" denoting "natural person", "false" denoting "no natural person").

After creation, the contact's validation status will be "pendingValidation". It may later change to "validated" or "invalid", depending on the outcome of the registry's validation. Note that the validation process for a contact may not be started until the contact is used in a domain or application, as the registry may choose to only conduct the validation if it is required for deciding whether or not to allocate a domain. A contact's validation status may thus remain "pendingValidation" indefinitely until it is associated with a domain or application.

An example of a <create> command using the extension can be found in the "Examples" section below.

1.2.2.2 EPP <contact:delete> Command

There are no extension elements for the contact delete command and response.

1.2.2.3 EPP <contact:transfer> Command

There are no extension elements for the transfer command and response.

1.2.2.4 EPP <contact:update> Command

For the contact update command, extension elements exist for the command, allowing to change or remove the public IDs and/or "natural person" flag for an existing contact. This is done by using an <el:update> element in the extension section of the command that identifies the contact eligibility extension namespace. The <el:update> element must contain an <el:chg> element, which in

turn may contain one or more <el:publicId> elements and/or an <el:naturalPerson/> element to remove or update the respective data. To remove the public ID of a type from a contact entirely, the <el:publicId> element may be left empty, with only the "type" attribute being present to signal the removal of the public ID of that type. Likewise, to remove the "natural person" flag from a contact entirely, the <el:naturalPerson> element may be left empty to signal the removal.

Note that any change to a contact's eligibility information may trigger re-validation by the registry and may thus change the contact's validation status, plus potentially the status of domains or applications associated with the contact.

1.3 Formal Syntax (Schema Definition)

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://xmlns.corenic.net/epp/contact-eligibility-1.1"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1"
  elementFormDefault="qualified">

  <annotation>
    <documentation>
      Extensible Provisioning Protocol v1.0
      extension schema for provisioning registration eligibility information
      for contact objects
    </documentation>
  </annotation>

  <!-- child elements found in EPP commands -->

  <element name="create" type="el:createType"/>
  <element name="update" type="el:updateType"/>

  <!-- child elements found in EPP responses -->

  <element name="infData" type="el:infDataType"/>

  <!-- optional string type for update commands (allows removing items) -->

  <simpleType name="optLatinType">
    <restriction base="string">
      <pattern value="\p{IsBasicLatin}*" />
    </restriction>
  </simpleType>

  <!-- non-empty string type for create commands and info responses -->

  <simpleType name="reqLatinType">
    <restriction base="string">
      <minLength value="1" />
      <pattern value="\p{IsBasicLatin}*" />
    </restriction>
  </simpleType>

  <!-- possible status values for a contact validation -->

  <simpleType name="validationStatusType">
    <restriction base="token">
      <enumeration value="pendingValidation" />
      <enumeration value="validated" />
    </restriction>
  </simpleType>
</schema>
```

```

        <enumeration value="invalid"/>
    </restriction>
</simpleType>

<!-- possible values for the "natural person" flag -->
<simpleType name="naturalPersonType">
    <restriction base="token">
        <enumeration value="true"/>
        <enumeration value="false"/>
        <enumeration value=""/> <!-- empty value means "unspecified" -->
    </restriction>
</simpleType>

<!-- required public ID + mandatory type attribute -->
<complexType name="reqPublicId">
    <simpleContent>
        <extension base="el:reqLatinType">
            <attribute name="type" use="required"/>
        </extension>
    </simpleContent>
</complexType>

<!-- optional public ID + mandatory type attribute -->
<complexType name="optPublicId">
    <simpleContent>
        <extension base="el:optLatinType">
            <attribute name="type" use="required"/>
        </extension>
    </simpleContent>
</complexType>

<!-- create extension -->
<complexType name="createType">
    <sequence>
        <!-- multiple public IDs may be specified, but each type only once -->
        <element name="publicId" type="el:reqPublicId" minOccurs="0"
            maxOccurs="unbounded"/>

        <element name="naturalPerson" type="el:naturalPersonType" minOccurs="0"/>
    </sequence>
</complexType>

<!-- update extension -->
<complexType name="updateType">
    <sequence>
        <element name="chg" type="el:chgType"/>
    </sequence>
</complexType>

<!-- type for changing / removing eligibility data -->
<complexType name="chgType">
    <sequence>
        <!-- multiple public IDs may be specified, but each type only once;
            empty elements (with mandatory type) indicate that the public ID
            for that type shall be removed -->
        <element name="publicId" type="el:optPublicId" minOccurs="0"
            maxOccurs="unbounded"/>

        <!-- empty element asks the natural person flag to be cleared -->
        <element name="naturalPerson" type="el:naturalPersonType" minOccurs="0"/>
    </sequence>
</complexType>

```

```
<!-- info extension -->

<complexType name="infDataType">
  <sequence>
    <!-- multiple public IDs may occur, but each type only once -->
    <element name="publicId" type="e1:reqPublicId" minOccurs="0"
      maxOccurs="unbounded"/>

    <element name="naturalPerson" type="e1:naturalPersonType" minOccurs="0"/>

    <element name="validationStatus" type="e1:validationStatusType"
      minOccurs="0"/>
  </sequence>
</complexType>
</schema>
```

1.4 Examples

In the following examples, "C:" represents lines sent by an EPP client and "S:" represents lines returned by the EPP server.

1.4.1 EPP <info> Command

1.4.1.1 Example <info> response with eligibility information:

```
S:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
S:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:  <response>
S:    <result code="1000">
S:      <msg>Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <infData xmlns="urn:ietf:params:xml:ns:contact-1.0">
S:        <id>sh8013</id>
S:        <roid>SH8013-REP</roid>
S:        <status s="linked"/>
S:        <status s="clientDeleteProhibited"/>
S:        <postalInfo type="int">
S:          <name>John Doe</name>
S:          <org>Example Inc.</org>
S:          <addr>
S:            <street>124 Example Dr.</street>
S:            <city>Dulles</city>
S:            <sp>VA</sp>
S:            <pc>20166-6503</pc>
S:            <cc>US</cc>
S:          </addr>
S:        </postalInfo>
S:        <voice x="1234">+1.7035555555</voice>
S:        <fax>+1.7035555556</fax>
S:        <email>jdoe@example.com</email>
S:        <clID>ClientY</clID>
S:        <crID>ClientX</crID>
S:        <crDate>1999-04-03T22:00:00.0Z</crDate>
S:        <upID>ClientX</upID>
S:        <upDate>1999-12-03T09:00:00.0Z</upDate>
S:        <trDate>2000-04-08T09:00:00.0Z</trDate>
S:        <authInfo>
S:          <pw>2fooBAR</pw>
S:        </authInfo>
S:        <disclose flag="0">
S:          <voice/>
S:          <email/>
```

```
S:      </disclose>
S:      </infData>
S:    </resData>
S:    <extension>
S:      <el:infData xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
S:        <el:publicId type="ch-upi">756.1234.5678.90</el:publicId>
S:        <el:naturalPerson>true</el:naturalPerson>
S:        <el:validationStatus>validationPending</el:validationStatus>
S:      </el:infData>
S:    </extension>
S:    <trID>
S:      <svTRID>ZYX-99958</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

1.4.2 EPP <create> Command

1.4.2.1 Example <create> command specifying eligibility information:

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <create>
C:      <create xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:        <id>sh8013</id>
C:        <postalInfo type="int">
C:          <name>John Doe</name>
C:          <org>Example Inc.</org>
C:          <addr>
C:            <street>124 Example Dr.</street>
C:            <city>Dulles</city>
C:            <sp>VA</sp>
C:            <pc>20166-6503</pc>
C:            <cc>US</cc>
C:          </addr>
C:        </postalInfo>
C:        <voice x="1234">+1.7035555555</voice>
C:        <fax>+1.7035555556</fax>
C:        <email>jdoe@example.com</email>
C:        <authInfo>
C:          <pw>2fooBAR</pw>
C:        </authInfo>
C:        <disclose flag="0">
C:          <voice/>
C:          <email/>
C:        </disclose>
C:      </create>
C:    </create>
C:    <extension>
C:      <el:create xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:        <el:publicId type="ch-upi">756.1234.5678.90</el:publicId>
C:        <el:naturalPerson>true</el:naturalPerson>
C:      </el:create>
C:    </extension>
C:    <c1TRID>ABC-12345</c1TRID>
C:  </command>
C:</epp>
```

1.4.3 EPP <update> Command

1.4.3.1 Example <update> command changing the eligibility information (public ID of type "ch-uid" and the natural person flag) of a contact, along with other contact data:

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <update>
C:      <update xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:        <id>sh8013</id>
C:        <add>
C:          <status s="clientDeleteProhibited"/>
C:        </add>
C:        <chg>
C:          <postalInfo type="int">
C:            <name>John Doe</name>
C:            <org/>
C:            <addr>
C:              <street>124 Example Dr.</street>
C:              <city>Dulles</city>
C:              <sp>VA</sp>
C:              <pc>20166-6503</pc>
C:              <cc>US</cc>
C:            </addr>
C:          </postalInfo>
C:          <voice>+1.7034444444</voice>
C:          <fax/>
C:          <authInfo>
C:            <pw>2fooBAR</pw>
C:          </authInfo>
C:          <disclose flag="1">
C:            <voice/>
C:            <email/>
C:          </disclose>
C:        </chg>
C:      </update>
C:    </update>
C:    <extension>
C:      <el:update xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:        <el:chg>
C:          <el:publicId type="ch-uid">CHE-123.456.789</el:publicId>
C:          <el:naturalPerson>false</el:naturalPerson>
C:        </el:chg>
C:      </el:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

1.4.3.2 Example <update> command changing only the eligibility information of a contact, retaining other contact data:

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <update>
C:      <update xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:        <id>sh8013</id>
C:      </update>
C:    </update>
C:    <extension>
C:      <el:update xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:        <el:chg>
C:          <el:publicId type="ch-uid">CHE-123.456.789</el:publicId>
C:          <el:naturalPerson>false</el:naturalPerson>
C:        </el:chg>
C:      </el:update>
C:    </extension>
```

```
C: <c1TRID>ABC-12345</c1TRID>
C: </command>
C: </epp>
```

1.4.3.3 Example <update> command removing the "public ID" eligibility information of a contact, retaining other contact data and retaining the "natural person" flag:

```
C: <?xml version="1.0" encoding="UTF-8" standalone="no"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <command>
C:     <update>
C:       <update xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:         <id>sh8013</id>
C:       </update>
C:     </update>
C:     <extension>
C:       <el:update xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:         <el:chg>
C:           <el:publicId type="ch-uid"/>
C:         </el:chg>
C:       </el:update>
C:     </extension>
C:   </command>
C: </epp>
```

1.4.3.4 Example <update> command removing the "natural person" eligibility information of a contact, retaining other contact data and the public IDs:

```
C: <?xml version="1.0" encoding="UTF-8" standalone="no"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <command>
C:     <update>
C:       <update xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:         <id>sh8013</id>
C:       </update>
C:     </update>
C:     <extension>
C:       <el:update xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:         <el:chg>
C:           <el:naturalPerson/>
C:         </el:chg>
C:       </el:update>
C:     </extension>
C:   </command>
C: </epp>
```

1.4.3.5 Example <update> command replacing the "public ID" eligibility information of a contact of another type, retaining other contact data and retaining the "natural person" flag:

```
C: <?xml version="1.0" encoding="UTF-8" standalone="no"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <command>
C:     <update>
C:       <update xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:         <id>sh8013</id>
C:       </update>
C:     </update>
C:     <extension>
C:       <el:update xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:         <el:chg>
C:           <el:publicId type="ch-uid"/>
C:         </el:chg>
C:       </el:update>
C:     </extension>
C:   </command>
C: </epp>
```

```
C:      <el:publicId type="gleif-lei">123400ABCDEFHIJKLM99</el:publicId>
C:      </el:chg>
C:      </el:update>
C:      </extension>
C:      <clTRID>ABC-12345</clTRID>
C:    </command>
C:  </epp>
```

1.4.3.6 Example <update> command removing all the eligibility information of a contact, retaining other contact data:

```
C: <?xml version="1.0" encoding="UTF-8" standalone="no"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <command>
C:     <update>
C:       <update xmlns="urn:ietf:params:xml:ns:contact-1.0">
C:         <id>sh8013</id>
C:       </update>
C:     </update>
C:     <extension>
C:       <el:update xmlns:el="http://xmlns.corenic.net/epp/contact-eligibility-1.1">
C:         <el:chg>
C:           <el:publicId type="gleif-lei"/>
C:           <el:naturalPerson/>
C:         </el:chg>
C:       </el:update>
C:     </extension>
C:     <clTRID>ABC-12345</clTRID>
C:   </command>
C: </epp>
```