              Extensible Provisioning Protocol (EPP) Organization Mapping

Abstract

   This document describes an Extensible Provisioning Protocol (EPP)
   mapping for provisioning and management of organization objects
   stored in a shared central repository.

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
   https://www.rfc-editor.org/info/rfc8543.

Table of Contents

1. Introduction

   There are many entities, such as registrars, resellers, DNS service
   operators, and privacy proxies, involved in the domain registration
   business. These kinds of entities have not been formally defined as
   having an object in Extensible Provisioning Protocol (EPP). This
   document provides a way to specify them as "organization" entities.

   This document describes an organization object mapping for version
   1.0 of the EPP [RFC5730]. This mapping is specified using XML 1.0 as
   described in [W3C.REC-xml-20081126] and XML Schema notation as
   described in [W3C.REC-xmlschema-1-20041028] and
   [W3C.REC-xmlschema-2-20041028].

2. Conventions Used in This Document

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in
   BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

   In examples, "C:" represents lines sent by a protocol client, and
   "S:" represents lines returned by a protocol server. Indentation and
   white space in examples are provided only to illustrate element
   relationships and are not a required feature of this specification.

   XML is case sensitive. Unless stated otherwise, XML specifications
   and examples provided in this document MUST be interpreted in the
   character case presented.

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

3. Object Attributes

   An EPP organization object has attributes and associated values that
   can be viewed and modified by the sponsoring client or the server.
   This section describes each attribute type in detail. The formal
   syntax for the attribute values described here can be found in the
   "Formal Syntax" section of this document and in the appropriate
   normative references.

3.1. Organization Identifier

   All EPP organizations are identified by a server-unique identifier.
   Organization identifiers are character strings with a specified
   minimum length, a specified maximum length, and a specified format.
   Organization identifiers use the "clIDType" client identifier syntax
   described in [RFC5730]. The corresponding element is <org:id>.

3.2. Organization Roles

   The organization roles are used to represent the relationship an
   organization could have. The corresponding element is <org:role>.
   An organization object MUST always have at least one associated role.
   Roles can be set only by the client that sponsors an organization
   object. A client can change the role of an organization object using
   the EPP <update> command (see Section 4.2.5).

3.2.1. Role Type

   An organization role MUST have a type field, which may have any of
   the values listed in Section 7.3. The corresponding element is
   <org:type>. An organization could have multiple roles with different
   role types.

3.2.2. Role Status

   A role of an organization object MAY have its own statuses. The
   corresponding element is <org:status>. The possible values for the
   role status are defined in Section 3.5.

3.2.3. Role Identifier

   A role MAY have a third-party-assigned identifier such as the IANA ID
   for registrars. The corresponding element is <org:roleID>.

   Example of organization role identifier:

   <org:role>
     <org:type>registrar</org:type>
     <org:status>ok</org:status>
     <org:status>linked</org:status>
     <org:roleID>1362</org:roleID>
   </org:role>

3.3. Contact and Client Identifiers

All EPP contacts are identified by server-unique identifiers.
Contact identifiers are character strings with a specified minimum
length, a specified maximum length, and a specified format. Contact
identifiers use the "clIDType" client identifier syntax described in
[RFC5730].

3.4. Organization Status Values

An organization object MUST always have at least one associated
status value. Status values can be set only by the client that
sponsors an organization object and by the server on which the object
resides. A client can change the status of an organization object
using the EPP <update> command. Each status value MAY be accompanied
by a string of human-readable text that describes the rationale for
the status applied to the object.

A client MUST NOT alter server status values set by the server. A
server MAY alter or override status values set by a client, subject
to local server policies. The status of an object MAY change as a
result of either a client-initiated transform command or an action
performed by a server operator.

Status values that can be added or removed by a client are prefixed
with "client". Corresponding server status values that can be added
or removed by a server are prefixed with "server". The "hold" and
"terminated" status values are server managed when the organization
has no parent identifier (Section 3.6) and otherwise MAY be client
managed based on server policy. Other status values that do not
begin with either "client" or "server" are server managed.

Status Value Descriptions:

o ok: This is the normal status value for an object that has no
   operations pending or active prohibitions. This value is set and
   removed by the server as other status values are added or removed.

o hold: Organization transform commands and new links MUST be
   rejected.

o terminated: The organization that has been terminated MUST NOT be
   linked. Organization transform commands and new links MUST be
   rejected.

o linked: The organization object has at least one active
   association with another object. The "linked" status is not
   explicitly set by the client. Servers should provide services to
   determine existing object associations.

o clientLinkProhibited, serverLinkProhibited: Requests to add new
   links to the organization MUST be rejected.

o clientUpdateProhibited, serverUpdateProhibited: Requests to update
   the object (other than to remove this status) MUST be rejected.

o clientDeleteProhibited, serverDeleteProhibited: Requests to delete
   the object MUST be rejected.

o pendingCreate, pendingUpdate, pendingDelete: A transform command
   has been processed for the object, but the action has not been
   completed by the server. Server operators can delay action
   completion for a variety of reasons, such as to allow for human
   review or third-party action. A transform command that is
   processed, but whose requested action is pending, is noted with
   response code 1001.

"pendingCreate", "ok", "hold", and "terminated" are mutually
exclusive statuses. An organization MUST have exactly one of these
statuses set.

"ok" status MAY only be combined with "linked" status.

A client or server MAY combine "linked" with either
"clientLinkProhibited" or "serverLinkProhibited" if new links must be
prohibited.

"pendingDelete" status MUST NOT be combined with either
"clientDeleteProhibited" or "serverDeleteProhibited" status.

The "pendingCreate", "pendingDelete", and "pendingUpdate" status
values MUST NOT be combined with each other.

If "clientUpdateProhibited" or "serverUpdateProhibited" is set, the
client will not be able to update the object. For
"clientUpdateProhibited", the client will first need to remove
"clientUpdateProhibited" prior to attempting to update the object.
The server can modify the object at any time.

3.5. Role Status Values

   A role SHOULD have at least one associated status value. Valid
   values include "ok", "linked", "clientLinkProhibited", and
   "serverLinkProhibited".

   Status Value Descriptions:

   o ok: This is the normal status value for a role that has no
      operations pending or active prohibitions. This value is set and
      removed by the server as other status values are added or removed.

   o linked: The role of an organization object has at least one active
      association with another object. The "linked" status is not
      explicitly set by the client. Servers SHOULD provide services to
      determine existing object associations.

   o clientLinkProhibited, serverLinkProhibited: Requests to add new
      links to the role MUST be rejected.

3.6. Parent Identifier

   Organizations can have more than one layer. The parent identifier,
   as defined with the <org:parentId> element, represents the parent
   organization identifier in a child organization.

   The case of reseller organizations provides an example. The parent
   identifier is not defined for the top-level reseller, namely the
   registrar of the registry. An N-tier reseller has a parent reseller
   and at least one child reseller. A reseller customer has a parent
   reseller and no child resellers.

   Loops MUST be prohibited. For example: if organization A has
   organization B as its parent identifier, organization B cannot have
   organization A as its parent identifier. The same is true for larger
   loops involving three or more organizations.

3.7. URL

   The URL represents the organization web home page, as defined with
   the <org:url> element.

3.8. Dates and Times

   Date and time attribute values MUST be represented in Coordinated
   Universal Time (UTC) using the Gregorian calendar. The extended
   date-time form using uppercase "T" and "Z" characters defined in
   [W3C.REC-xmlschema-2-20041028] MUST be used to represent date-time
   values, as XML Schema does not support truncated date-time forms or
   lowercase "T" and "Z" characters.

4. EPP Command Mapping

   A detailed description of the EPP syntax and semantics can be found
   in the EPP core protocol specification [RFC5730]. The command
   mappings described here are specifically for use in provisioning and
   managing organization information via EPP.

4.1. EPP Query Commands

   EPP provides two commands to retrieve organization information:
   <check> to determine if an organization object can be provisioned
   within a repository and <info> to retrieve detailed information
   associated with an organization object. This document does not
   define a mapping for the EPP <transfer> command to retrieve
   organization-object transfer status information.

4.1.1. EPP <check> Command

   The EPP <check> command is used to determine if an object can be
   provisioned within a repository. It provides a hint that allows a
   client to anticipate the success or failure of provisioning an object
   using the <create> command, as object-provisioning requirements are
   ultimately a matter of server policy.

   In addition to the standard EPP command elements, the <check> command
   MUST contain an <org:check> element. This element or its ancestor
   element MUST identify the organization namespace
   "urn:ietf:params:xml:ns:epp:org-1.0". The <org:check> element
   contains the following child elements:

   o One or more <org:id> elements that contain the server-unique
      identifier of the organization objects to be queried.

Example <check> command:

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <check>
C:      <org:check
C:          xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
C:        <org:id>res1523</org:id>
C:        <org:id>re1523</org:id>
C:        <org:id>1523res</org:id>
C:      </org:check>
C:    </check>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When a <check> command has been processed successfully, the EPP
<resData> element MUST contain a child <org:chkData> element. This
element or its ancestor element MUST identify the organization
namespace "urn:ietf:params:xml:ns:epp:org-1.0". The <org:chkData>
element contains one or more <org:cd> elements that contain the
following child elements:

o An <org:id> element that identifies the queried object. This
   element MUST contain an "avail" attribute whose value indicates
   object availability (can it be provisioned or not) at the moment
   the <check> command was completed. A value of "1" or "true" means
   that the object can be provisioned. A value of "0" or "false"
   means that the object cannot be provisioned.

o An OPTIONAL <org:reason> element that may be provided when an
   object cannot be provisioned. If present, this element contains
   server-specific text to help explain why the object cannot be
   provisioned. This text MUST be represented in the response
   language previously negotiated with the client; an OPTIONAL "lang"
   attribute as defined in [RFC5646] may be present to identify the
   language if the negotiated value is something other than the
   default value of "en"(English).

Example <check> response:

```
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 lang="en">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <org:chkData
S:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
S:        <org:cd>
S:          <org:id avail="1">res1523</org:id>
S:        </org:cd>
S:        <org:cd>
S:          <org:id avail="0">re1523</org:id>
S:          <org:reason lang="en">In use</org:reason>
S:        </org:cd>
S:        <org:cd>
S:          <org:id avail="1">1523res</org:id>
S:        </org:cd>
S:      </org:chkData>
S:    </resData>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54322-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

An EPP error response MUST be returned if a <check> command cannot be
processed for any reason.

4.1.2. EPP <info> Command

The EPP <info> command is used to retrieve information associated
with an organization object. In addition to the standard EPP command
elements, the <info> command MUST contain an <org:info> element.
This element or its ancestor element MUST identify the organization
namespace "urn:ietf:params:xml:ns:epp:org-1.0". The <org:info>
element contains the following child element:

o An <org:id> element that contains the server-unique identifier of
   the organization object to be queried.

Example <info> command:

```
C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:  <command>
C:    <info>
C:      <org:info
C:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
C:        <org:id>res1523</org:id>
C:      </org:info>
C:    </info>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When an <info> command has been processed successfully, the EPP
<resData> element MUST contain a child <org:infData> element. This
element or its ancestor element MUST identify the organization
namespace "urn:ietf:params:xml:ns:epp:org-1.0". The <org:infData>
element contains the following child elements:

o An <org:id> element that contains the server-unique identifier of
   the organization object, as defined in Section 3.1.

o An <org:roid> element that contains the Repository Object
   Identifier assigned to the organization object when the object was
   created.

o One or more <org:role> elements that contain the role type, role
   statuses, and optional role ID of the organization.

   * An <org:type> element that contains the type of the
      organization, as defined in Section 3.2.

   * One or more <org:status> elements that contain the role
      statuses. The values of the role status are defined in
      Section 3.5.

   * An OPTIONAL <org:roleID> element that contains a third-party-
      assigned identifier, such as IANA ID for registrars, as defined
      in Section 3.2.3.

o One or more <org:status> elements that contain the operational
   status of the organization, as defined in Section 3.4.

o An OPTIONAL <org:parentId> element that contains the identifier of
   the parent object, as defined in Section 3.6.

   o Zero to two <org:postalInfo> elements that contain postal-address
      information. Two elements are provided so that address
      information can be provided in both internationalized and
      localized forms; a "type" attribute is used to identify the two
      forms. If an internationalized form (type="int") is provided,
      element content MUST be represented in a subset of Unicode
      [UNICODE] in the range U+0020 - U+007E. If a localized form
      (type="loc") is provided, element content MAY be represented in
      unrestricted UTF-8. The <org:postalInfo> element contains the
      following child elements:

      * An <org:name> element that contains the name of the
         organization.

      * An OPTIONAL <org:addr> element that contains address
         information associated with the organization. An <org:addr>
         element contains the following child elements:

         + One, two, or three <org:street> elements that contain the
            organization's street address.

         + An <org:city> element that contains the organization's city.

         + An OPTIONAL <org:sp> element that contains the
            organization's state or province.

         + An OPTIONAL <org:pc> element that contains the
            organization's postal code.

         + An <org:cc> element that contains the alpha-2 organization's
            country code. The detailed format of this element is
            described in Section 2.4.3 of [RFC5733].

   o An OPTIONAL <org:voice> element that contains the organization's
      voice telephone number. The detailed format of this element is
      described in Section 2.5 of [RFC5733].

   o An OPTIONAL <org:fax> element that contains the organization's
      facsimile telephone number. The detailed format of this element
      is described in Section 2.5 of [RFC5733].

   o An OPTIONAL <org:email> element that contains the organization's
      email address. The detailed format of this element is described
      in [RFC5322].

   o An OPTIONAL <org:url> element that contains the URL to the website
      of the organization. The detailed format of this element is
      described in [RFC3986].

   o Zero or more <org:contact> elements that contain identifiers for
      the contact objects to be associated with the organization object.
      Contact object identifiers MUST be known to the server before the
      contact object can be associated with the organization object.
      The required "type" is used to represent contact types. The type
      values include "admin", "tech", "billing", "abuse", and "custom".
      The OPTIONAL "typeName" attribute is used to define the name of a
      "custom" type.

   o An OPTIONAL <org:clID> element that contains the organization
      identifier of the sponsoring client. There is no <org:clID>
      element if the organization is managed by the registry.

   o An <org:crID> element that contains the identifier of the client
      that created the organization object.

   o An <org:crDate> element that contains the date and time of
      organization object creation.

   o An <org:upID> element that contains the identifier of the client
      that last updated the organization object. This element MUST NOT
      be present if the organization has never been modified.

   o An <org:upDate> element that contains the date and time of the
      most recent organization object modification. This element MUST
      NOT be present if the organization object has never been modified.

   Example <info> response for "Example Registrar Inc." organization
   object with identifier "registrar1362":

   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 lang="en">Command completed successfully</msg>
   S: </result>
   S: <resData>
   S: <org:infData
   S: xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
   S: <org:id>registrar1362</org:id>
   S: <org:roid>registrar1362-REP</org:roid>
   S: <org:role>
   S: <org:type>registrar</org:type>
   S: <org:status>ok</org:status>
   S: <org:status>linked</org:status>
   S: <org:roleID>1362</org:roleID>
   S: </org:role>
   S: <org:status>ok</org:status>

```
S:              <org:postalInfo type="int">
S:                <org:name>Example Registrar Inc.</org:name>
S:                <org:addr>
S:                  <org:street>123 Example Dr.</org:street>
S:                  <org:street>Suite 100</org:street>
S:                  <org:city>Dulles</org:city>
S:                  <org:sp>VA</org:sp>
S:                  <org:pc>20166-6503</org:pc>
S:                  <org:cc>US</org:cc>
S:                </org:addr>
S:              </org:postalInfo>
S:              <org:voice x="1234">+1.7035555555</org:voice>
S:              <org:fax>+1.7035555556</org:fax>
S:              <org:email>contact@organization.example</org:email>
S:              <org:url>https://organization.example</org:url>
S:              <org:contact type="admin">sh8013</org:contact>
S:              <org:contact type="billing">sh8013</org:contact>
S:              <org:contact type="custom"
S:                typeName="legal">sh8013</org:contact>
S:              <org:crID>ClientX</org:crID>
S:              <org:crDate>2018-04-03T22:00:00.0Z</org:crDate>
S:              <org:upID>ClientX</org:upID>
S:              <org:upDate>2018-12-03T09:00:00.0Z</org:upDate>
S:            </org:infData>
S:          </resData>
S:          <trID>
S:            <clTRID>ABC-12345</clTRID>
S:            <svTRID>54322-XYZ</svTRID>
S:          </trID>
S:        </response>
S:</epp>
```

   Example <info> response for "Example Reseller Inc." organization
   object of reseller type managed by identifier "registrar1362":

```
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 lang="en">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <org:infData
S:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
S:        <org:id>reseller1523</org:id>
S:        <org:roid>reseller1523-REP</org:roid>
S:        <org:role>
S:          <org:type>reseller</org:type>
```

```
S:              <org:status>ok</org:status>
S:              <org:status>linked</org:status>
S:           </org:role>
S:           <org:status>ok</org:status>
S:           <org:parentId>registrar1362</org:parentId>
S:           <org:postalInfo type="int">
S:             <org:name>Example Reseller Inc.</org:name>
S:             <org:addr>
S:               <org:street>123 Example Dr.</org:street>
S:               <org:street>Suite 100</org:street>
S:               <org:city>Dulles</org:city>
S:               <org:sp>VA</org:sp>
S:               <org:pc>20166-6503</org:pc>
S:               <org:cc>US</org:cc>
S:             </org:addr>
S:           </org:postalInfo>
S:           <org:fax>+1.7035555556</org:fax>
S:           <org:url>https://organization.example</org:url>
S:           <org:contact type="admin">sh8013</org:contact>
S:           <org:clID>1362</org:clID>
S:           <org:crID>ClientX</org:crID>
S:           <org:crDate>2018-04-03T22:00:00.0Z</org:crDate>
S:           <org:upID>ClientX</org:upID>
S:           <org:upDate>2018-12-03T09:00:00.0Z</org:upDate>
S:         </org:infData>
S:       </resData>
S:       <trID>
S:         <clTRID>ABC-12345</clTRID>
S:         <svTRID>54322-XYZ</svTRID>
S:       </trID>
S:     </response>
S:</epp>
```

An EPP error response MUST be returned if an <info> command cannot be processed for any reason.

4.1.3. EPP <transfer> Query Command

The transfer semantics do not apply to organization objects. No EPP <transfer> query command is defined in this document.

4.2. EPP Transform Commands

This document provides three commands to transform organization
object information: <create> to create an instance of an organization
object, <delete> to delete an instance of an organization object, and
<update> to change information associated with an organization
object. This document does not define a mapping for the EPP
<transfer> and <renew> command.

Transform commands are typically processed and completed in real
time. Server operators MAY receive and process transform commands
but defer completing the requested action if human or third-party
review is required before the requested action can be completed. In
such situations, the server MUST return a 1001 response code to the
client to note that the command has been received and processed but
that the requested action is pending. The server MUST also manage
the status of the object that is the subject of the command to
reflect the initiation and completion of the requested action. Once
the action has been completed, the client MUST be notified using a
service message that the action has been completed and the status of
the object has changed. Other notification methods MAY be used in
addition to the required service message.

4.2.1. EPP <create> Command

The EPP <create> command provides a transform operation that allows a
client to create an organization object. In addition to the standard
EPP command elements, the <create> command MUST contain an
<org:create> element. This element or its ancestor element MUST
identify the organization namespace "urn:ietf:params:xml:ns:epp:org-
1.0". The <org:create> element contains the following child
elements:

o An <org:id> element that contains the desired server-unique
   identifier for the organization to be created, as defined in
   Section 3.1.

o One or more <org:role> elements that contain the role type, role
   statuses, and optional role ID of the organization.

   * An <org:type> element that contains the type of the
      organization, as defined in Section 3.2.

   * Zero or more <org:status> elements that contain the role
      statuses. The possible values of the role statuses are defined
      in Section 3.5.

        * An OPTIONAL <org:roleID> element that contains a third-party-
          assigned identifier, such as IANA ID for registrars, as defined
          in Section 3.2.3.

   o Zero or more <org:status> elements that contain the operational
     status of the organization, as defined in Section 3.4.

   o An OPTIONAL <org:parentId> element that contains the identifier of
     the parent object, as defined in Section 3.6.

   o Zero to two <org:postalInfo> elements that contain postal-address
     information. Two elements are provided so that address
     information can be provided in both internationalized and
     localized forms; a "type" attribute is used to identify the two
     forms. If an internationalized form (type="int") is provided,
     element content MUST be represented in a subset of Unicode
     [UNICODE] in the range U+0020 - U+007E. If a localized form
     (type="loc") is provided, element content MAY be represented in
     unrestricted UTF-8. The <org:postalInfo> element contains the
     following child elements:

     * An <org:name> element that contains the name of the
       organization.

     * An OPTIONAL <org:addr> element that contains address
       information associated with the organization. An <org:addr>
       element contains the following child elements:

       + One, two, or three <org:street> elements that contain the
         organization's street address.

       + An <org:city> element that contains the organization's city.

       + An OPTIONAL <org:sp> element that contains the
         organization's state or province.

       + An OPTIONAL <org:pc> element that contains the
         organization's postal code.

       + An <org:cc> element that contains the alpha-2 organization's
         country code. The detailed format of this element is
         described in Section 2.4.3 of [RFC5733].

   o An OPTIONAL <org:voice> element that contains the organization's
     voice telephone number. The detailed format of this element is
     described in Section 2.5 of [RFC5733].

o   An OPTIONAL <org:fax> element that contains the organization's
    facsimile telephone number. The detailed format of this element
    is described in Section 2.5 of [RFC5733].

o   An OPTIONAL <org:email> element that contains the organization's
    email address. The detailed format of this element is described
    of [RFC5322].

o   An OPTIONAL <org:url> element that contains the URL to the website
    of the organization. The detailed format of this element is
    described in [RFC3986].

o   Zero or more <org:contact> elements that contain identifiers for
    the contact objects associated with the organization object.

Example <create> command:

```
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:      <org:create
C:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
C:        <org:id>res1523</org:id>
C:        <org:role>
C:          <org:type>reseller</org:type>
C:        </org:role>
C:        <org:parentId>1523res</org:parentId>
C:        <org:postalInfo type="int">
C:          <org:name>Example Organization Inc.</org:name>
C:          <org:addr>
C:            <org:street>123 Example Dr.</org:street>
C:            <org:street>Suite 100</org:street>
C:            <org:city>Dulles</org:city>
C:            <org:sp>VA</org:sp>
C:            <org:pc>20166-6503</org:pc>
C:            <org:cc>US</org:cc>
C:          </org:addr>
C:        </org:postalInfo>
C:        <org:voice x="1234">+1.7035555555</org:voice>
C:        <org:fax>+1.7035555556</org:fax>
C:        <org:email>contact@organization.example</org:email>
C:        <org:url>https://organization.example</org:url>
C:        <org:contact type="admin">sh8013</org:contact>
C:        <org:contact type="billing">sh8013</org:contact>
C:      </org:create>
C:    </create>
C:    <clTRID>ABC-12345</clTRID>
```

```
C:    </command>
C:</epp>
```

When a <create> command has been processed successfully, the EPP
<resData> element MUST contain a child <org:creData> element. This
element or its ancestor element MUST identify the organization
namespace "urn:ietf:params:xml:ns:epp:org-1.0". The <org:creData>
element contains the following child elements:

o An <org:id> element that contains the server-unique identifier for
   the created organization, as defined in Section 3.1.

o An <org:crDate> element that contains the date and time of
   organization-object creation.

Example <create> response:

```
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 lang="en">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <org:creData
S:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
S:        <org:id>res1523</org:id>
S:        <org:crDate>2018-04-03T22:00:00.0Z</org:crDate>
S:      </org:creData>
S:    </resData>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54321-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

An EPP error response MUST be returned if a <create> command cannot
be processed for any reason.

4.2.2. EPP <delete> Command

   The EPP <delete> command provides a transform operation that allows a
   client to delete an organization object. In addition to the standard
   EPP command elements, the <delete> command MUST contain an
   <org:delete> element. This element or its ancestor element MUST
   identify the organization namespace "urn:ietf:params:xml:ns:epp:org-
   1.0". The <org:delete> element MUST contain the following child
   element:

   o An <org:id> element that contains the server-unique identifier of
      the organization object to be deleted, as defined in Section 3.1.

   An organization object MUST NOT be deleted if it is associated with
   other known objects. An associated organization MUST NOT be deleted
   until associations with other known objects have been broken. A
   server MUST notify clients that object relationships exist by sending
   a 2305 error response code when a <delete> command is attempted and
   fails due to existing object relationships.

   Example <delete> command:

   C:<?xml version="1.0" encoding="UTF-8" standalone="no"?>
   C:<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
   C: <command>
   C: <delete>
   C: <org:delete
   C: xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
   C: <org:id>res1523</org:id>
   C: </org:delete>
   C: </delete>
   C: <clTRID>ABC-12345</clTRID>
   C: </command>
   C:</epp>

   When a <delete> command has been processed successfully, a server
   MUST respond with an EPP response with no <resData> element.

Example <delete> response:

```
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 lang="en">Command completed successfully</msg>
S:    </result>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54321-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

An EPP error response MUST be returned if a <delete> command cannot
be processed for any reason.

4.2.3. EPP <renew> Command

   Renewal semantics do not apply to organization objects, so there is
   no mapping defined for the EPP <renew> command.

4.2.4. EPP <transfer> Command

   Transfer semantics do not apply to organization objects, so there is
   no mapping defined for the EPP <transfer> command.

4.2.5. EPP <update> Command

   The EPP <update> command provides a transform operation that allows a
   client to modify the attributes of an organization object. In
   addition to the standard EPP command elements, the <update> command
   MUST contain an <org:update> element. This element or its ancestor
   element MUST identify the organization namespace
   "urn:ietf:params:xml:ns:epp:org-1.0". The <org:update> element
   contains the following child elements:

   o An <org:id> element that contains the server-unique identifier of
      the organization object to be updated, as defined in Section 3.1.

   o An OPTIONAL <org:add> element that contains attribute values to be
      added to the object.

   o An OPTIONAL <org:rem> element that contains attribute values to be
      removed from the object.

o An OPTIONAL <org:chg> element that contains attribute values to be
   changed.

At least one <org:add>, <org:rem>, or <org:chg> element MUST be
provided if the command is not being extended. All of these elements
MAY be omitted if an <update> extension is present. The OPTIONAL
<org:add> and <org:rem> elements contain the following child
elements:

o Zero or more <org:contact> elements that contain the identifiers
   for contact objects to be associated with or removed from the
   organization object. Contact object identifiers MUST be known to
   the server before the contact object can be associated with the
   organization object.

o Zero or more <org:role> elements that contain the role type, role
   statuses, and optional role ID of the organization.

   * An <org:type> element that contains the role type of the
      organization, as defined in Section 3.2. The role type
      uniquely identifies the role to update.

   * Zero or more <org:status> elements that contain the role
      statuses. The values of the role status are defined in
      Section 3.5.

   * An OPTIONAL <org:roleID> element that contains a third-party-
      assigned identifier, such as IANA ID for registrars, as defined
      in Section 3.2.3.

o Zero or more <org:status> elements that contain the operational
   status of the organization.

An OPTIONAL <org:chg> element contains the following child elements,
where at least one child element MUST be present:

o An OPTIONAL <org:parentId> element that contains the identifier of
   the parent object.

o Zero to two <org:postalInfo> elements that contain postal-address
   information. Two elements are provided so that address
   information can be provided in both internationalized and
   localized forms; a "type" attribute is used to identify the two
   forms. If an internationalized form (type="int") is provided,
   element content MUST be represented in a subset of Unicode
   [UNICODE] in the range U+0020 - U+007E. If a localized form
   (type="loc") is provided, element content MAY be represented in
   unrestricted UTF-8. The change of the postal info is defined as a

replacement of that postal info element with the contents of the
sub-elements included in the <update> command. An empty
<org:postalInfo> element is supported to allow a type of postal
info to be removed. The <org:postalInfo> element contains the
following child elements:

*   An <org:name> element that contains the name of the
    organization.

*   An OPTIONAL <org:addr> element that contains address
    information associated with the organization. An <org:addr>
    element contains the following child elements:

    +   One, two, or three <org:street> elements that contain the
        organization's street address.

    +   An <org:city> element that contains the organization's city.

    +   An OPTIONAL <org:sp> element that contains the
        organization's state or province.

    +   An OPTIONAL <org:pc> element that contains the
        organization's postal code.

    +   An <org:cc> element that contains the alpha-2 organization's
        country code. The detailed format of this element is
        described in Section 2.4.3 of [RFC5733].

o   An OPTIONAL <org:voice> element that contains the organization's
    voice telephone number. The detailed format of this element is
    described in Section 2.5 of [RFC5733].

o   An OPTIONAL <org:fax> element that contains the organization's
    facsimile telephone number. The detailed format of this element
    is described in Section 2.5 of [RFC5733].

o   An OPTIONAL <org:email> element that contains the organization's
    email address. The detailed format of this element is described
    in [RFC5322].

o   An OPTIONAL <org:url> element that contains the URL to the website
    of the organization. The detailed format of this element is
    described in [RFC3986].

Example <update> command:

```
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:      <org:update
C:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
C:        <org:id>res1523</org:id>
C:        <org:add>
C:          <org:contact type="tech">sh8013</org:contact>
C:          <org:role>
C:            <org:type>privacyproxy</org:type>
C:            <org:status>clientLinkProhibited</org:status>
C:          </org:role>
C:          <org:status>clientLinkProhibited</org:status>
C:        </org:add>
C:        <org:rem>
C:          <org:contact type="billing">sh8014</org:contact>
C:          <org:role>
C:            <org:type>reseller</org:type>
C:          </org:role>
C:        </org:rem>
C:        <org:chg>
C:          <org:postalInfo type="int">
C:            <org:addr>
C:              <org:street>124 Example Dr.</org:street>
C:              <org:street>Suite 200</org:street>
C:              <org:city>Dulles</org:city>
C:              <org:sp>VA</org:sp>
C:              <org:pc>20166-6503</org:pc>
C:              <org:cc>US</org:cc>
C:            </org:addr>
C:          </org:postalInfo>
C:          <org:voice>+1.7034444444</org:voice>
C:          <org:fax/>
C:        </org:chg>
C:      </org:update>
C:    </update>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When an <update> command has been processed successfully, a server
MUST respond with an EPP response with no <resData> element.

Example <update> response:

```
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 lang="en">Command completed successfully</msg>
S:    </result>
S:    <trID>
S:      <clTRID>ABC-12345</clTRID>
S:      <svTRID>54321-XYZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

An EPP error response MUST be returned if an <update> command cannot
be processed for any reason.

4.3. Offline Review of Requested Actions

Commands are processed by a server in the order they are received
from a client. Though an immediate response confirming receipt and
processing of the command is produced by the server, a server
operator MAY perform an offline review of requested transform
commands before completing the requested action. In such situations,
the response from the server MUST clearly note that the transform
command has been received and processed, but the requested action is
pending. The status in the response of the corresponding object MUST
clearly reflect processing of the pending action. The server MUST
notify the client when offline processing of the action has been
completed.

Examples describing a <create> command that requires offline review
are included here. Note the result code and message returned in
response to the <create> command.

```
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="1001">
S:      <msg lang="en">Command completed successfully;
S:        action pending</msg>
S:    </result>
S:    <resData>
S:      <org:creData
S:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
S:        <org:id>res1523</org:id>
S:        <org:crDate>2018-04-03T22:00:00.0Z</org:crDate>
```

```
S:       </org:creData>
S:     </resData>
S:     <trID>
S:       <clTRID>ABC-12345</clTRID>
S:       <svTRID>54321-XYZ</svTRID>
S:     </trID>
S:   </response>
S:</epp>
```

The status of the organization object after returning this response
MUST include "pendingCreate". The server operator reviews the
request offline and informs the client of the outcome of the review
by queuing a service message for retrieval via the <poll> command; it
MAY additionally use an out-of-band mechanism to inform the client of
the outcome.

The service message MUST contain text that describes the notification
in the child <msg> element of the response <msgQ> element. In
addition, the EPP <resData> element MUST contain a child
<org:panData> element. This element or its ancestor element MUST
identify the organization namespace "urn:ietf:params:xml:ns:epp:org-
1.0". The <org:panData> element contains the following child
elements:

o An <org:id> element that contains the server-unique identifier of
   the organization object. The <org:id> element contains a REQUIRED
   "paResult" attribute. A positive boolean value indicates that the
   request has been approved and completed. A negative boolean value
   indicates that the request has been denied and the requested
   action has not been taken.

o An <org:paTRID> element that contains the client transaction
   identifier and server transaction identifier returned with the
   original response to process the command. The client transaction
   identifier is OPTIONAL and will only be returned if the client
   provided an identifier with the original <create> command.

o An <org:paDate> element that contains the date and time describing
   when review of the requested action was completed.

Example "review completed" service message:

```
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="1301">
S:      <msg lang="en">Command completed successfully;
S:        ack to dequeue</msg>
```

```
S:     </result>
S:     <msgQ count="5" id="12345">
S:       <qDate>2018-04-04T22:01:00.0Z</qDate>
S:       <msg>Pending action completed successfully.</msg>
S:     </msgQ>
S:     <resData>
S:       <org:panData
S:        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0">
S:         <org:id paResult="1">res1523</org:id>
S:         <org:paTRID>
S:           <clTRID>ABC-12345</clTRID>
S:           <svTRID>54321-XYZ</svTRID>
S:         </org:paTRID>
S:         <org:paDate>2018-04-04T22:00:00.0Z</org:paDate>
S:       </org:panData>
S:     </resData>
S:     <trID>
S:       <clTRID>BCD-23456</clTRID>
S:       <svTRID>65432-WXY</svTRID>
S:     </trID>
S:   </response>
S:</epp>
```

5. Formal Syntax

An EPP object mapping is specified in XML Schema notation. 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.

```
BEGIN
<?xml version="1.0" encoding="UTF-8"?>

<schema targetNamespace="urn:ietf:params:xml:ns:epp:org-1.0"
        xmlns:org="urn:ietf:params:xml:ns:epp:org-1.0"
        xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
        xmlns:eppcom="urn:ietf:params:xml:ns:eppcom-1.0"
        xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="qualified">

<!--
Import common element types.
-->
 <import namespace="urn:ietf:params:xml:ns:eppcom-1.0"/>
 <import namespace="urn:ietf:params:xml:ns:epp-1.0"/>
```

```
         <annotation>
           <documentation>
             Extensible Provisioning Protocol v1.0
             organization provisioning schema.
           </documentation>
         </annotation>

      <!--
      Child elements found in EPP commands.
      -->
       <element name="create" type="org:createType"/>
       <element name="delete" type="org:sIDType"/>
       <element name="update" type="org:updateType"/>
       <element name="check" type="org:mIDType"/>
       <element name="info" type="org:infoType"/>
       <element name="panData" type="org:panDataType"/>

      <!--
      Utility types.
      -->
       <simpleType name="statusType">
         <restriction base="token">
           <enumeration value="ok"/>
           <enumeration value="hold"/>
           <enumeration value="terminated"/>
           <enumeration value="clientDeleteProhibited"/>
           <enumeration value="clientUpdateProhibited"/>
           <enumeration value="clientLinkProhibited"/>
           <enumeration value="linked"/>
           <enumeration value="pendingCreate"/>
           <enumeration value="pendingUpdate"/>
           <enumeration value="pendingDelete"/>
           <enumeration value="serverDeleteProhibited"/>
           <enumeration value="serverUpdateProhibited"/>
           <enumeration value="serverLinkProhibited"/>
         </restriction>
       </simpleType>

       <simpleType name="roleStatusType">
         <restriction base="token">
           <enumeration value="ok"/>
           <enumeration value="clientLinkProhibited"/>
           <enumeration value="linked"/>
           <enumeration value="serverLinkProhibited"/>
         </restriction>
       </simpleType>

       <complexType name="roleType">
```

```
      <sequence>
        <element name="type" type="token"/>
        <element name="status" type="org:roleStatusType"
          minOccurs="0" maxOccurs="3"/>
        <element name="roleID" type="token" minOccurs="0"/>
      </sequence>
    </complexType>

    <complexType name="postalInfoType">
      <sequence>
        <element name="name"
          type="org:postalLineType"/>
        <element name="addr"
          type="org:addrType" minOccurs="0"/>
      </sequence>
      <attribute name="type"
        type="org:postalInfoEnumType"
        use="required"/>
    </complexType>

    <complexType name="contactType">
      <simpleContent>
        <extension base="eppcom:clIDType">
          <attribute name="type" type="org:contactAttrType"
            use="required"/>
          <attribute name="typeName" type="token"/>
        </extension>
      </simpleContent>
    </complexType>

    <simpleType name="contactAttrType">
      <restriction base="token">
        <enumeration value="admin"/>
        <enumeration value="billing"/>
        <enumeration value="tech"/>
        <enumeration value="abuse"/>
        <enumeration value="custom"/>
      </restriction>
    </simpleType>

    <complexType name="e164Type">
      <simpleContent>
        <extension base="org:e164StringType">
          <attribute name="x" type="token"/>
        </extension>
      </simpleContent>
    </complexType>
```

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

      <simpleType name="postalLineType">
        <restriction base="normalizedString">
          <minLength value="1"/>
          <maxLength value="255"/>
        </restriction>
      </simpleType>

      <simpleType name="optPostalLineType">
        <restriction base="normalizedString">
          <maxLength value="255"/>
        </restriction>
      </simpleType>

      <simpleType name="pcType">
        <restriction base="token">
          <maxLength value="16"/>
        </restriction>
      </simpleType>

      <simpleType name="ccType">
        <restriction base="token">
          <length value="2"/>
        </restriction>
      </simpleType>

      <complexType name="addrType">
        <sequence>
          <element name="street" type="org:optPostalLineType"
            minOccurs="0" maxOccurs="3"/>
          <element name="city" type="org:postalLineType"/>
          <element name="sp" type="org:optPostalLineType"
            minOccurs="0"/>
          <element name="pc" type="org:pcType"
            minOccurs="0"/>
          <element name="cc" type="org:ccType"/>
        </sequence>
      </complexType>

      <simpleType name="postalInfoEnumType">
        <restriction base="token">
          <enumeration value="loc"/>
```

```
        <enumeration value="int"/>
      </restriction>
    </simpleType>

    <!--
    Child element of commands that require only an identifier.
    -->
     <complexType name="sIDType">
       <sequence>
         <element name="id" type="eppcom:clIDType"/>
       </sequence>
     </complexType>

    <!--
    Child element of commands that accept multiple identifiers.
    -->
     <complexType name="mIDType">
       <sequence>
         <element name="id"
           type="eppcom:clIDType" maxOccurs="unbounded"/>
       </sequence>
     </complexType>

    <!--
    Pending action notification response elements.
    -->
     <complexType name="panDataType">
       <sequence>
         <element name="id" type="org:paCLIDType"/>
         <element name="paTRID" type="epp:trIDType"/>
         <element name="paDate" type="dateTime"/>
       </sequence>
     </complexType>

     <complexType name="paCLIDType">
       <simpleContent>
         <extension base="eppcom:clIDType">
           <attribute name="paResult" type="boolean"
             use="required"/>
         </extension>
       </simpleContent>
     </complexType>

    <!--
    Child elements of the <info> commands.
    -->
     <complexType name="infoType">
       <sequence>
```

```
            <element name="id"
              type="eppcom:clIDType"/>
          </sequence>
        </complexType>

      <!--
      Child elements of the <create> command.
      -->
        <complexType name="createType">
          <sequence>
            <element name="id"
              type="eppcom:clIDType"/>
            <element name="role"
              type="org:roleType" maxOccurs="unbounded"/>
            <element name="status"
              type="org:statusType" minOccurs="0" maxOccurs="4"/>
            <element name="parentId"
              type="eppcom:clIDType" minOccurs="0"/>
            <element name="postalInfo"
              type="org:postalInfoType" minOccurs="0" maxOccurs="2"/>
            <element name="voice"
              type="org:e164Type" minOccurs="0"/>
            <element name="fax"
              type="org:e164Type" minOccurs="0"/>
            <element name="email"
              type="eppcom:minTokenType" minOccurs="0"/>
            <element name="url"
              type="anyURI" minOccurs="0"/>
            <element name="contact"
              type="org:contactType"
              minOccurs="0" maxOccurs="unbounded"/>
          </sequence>
        </complexType>

      <!--
      Child elements of the <update> command.
      -->
        <complexType name="updateType">
          <sequence>
            <element name="id"
              type="eppcom:clIDType"/>
            <element name="add"
              type="org:addRemType" minOccurs="0"/>
            <element name="rem"
              type="org:addRemType" minOccurs="0"/>
            <element name="chg"
              type="org:chgType" minOccurs="0"/>
          </sequence>
```

```
          </complexType>

      <!--
      Data elements that can be added or removed.
      -->
       <complexType name="addRemType">
         <sequence>
           <element name="contact"
             type="org:contactType" minOccurs="0"
             maxOccurs="unbounded"/>
           <element name="role" type="org:roleType"
             minOccurs="0" maxOccurs="unbounded"/>
           <element name="status" type="org:statusType"
             minOccurs="0" maxOccurs="9"/>
         </sequence>
       </complexType>

      <!--
      Data elements that can be changed.
      -->
       <complexType name="chgType">
         <sequence>
           <element name="parentId"
             type="eppcom:clIDType" minOccurs="0"/>
           <element name="postalInfo"
             type="org:chgPostalInfoType"
             minOccurs="0" maxOccurs="2"/>
           <element name="voice"
             type="org:e164Type" minOccurs="0"/>
           <element name="fax"
             type="org:e164Type" minOccurs="0"/>
           <element name="email"
             type="eppcom:minTokenType" minOccurs="0"/>
           <element name="url"
             type="anyURI" minOccurs="0"/>
         </sequence>
       </complexType>

       <complexType name="chgPostalInfoType">
         <sequence>
           <element name="name"
             type="org:postalLineType" minOccurs="0"/>
           <element name="addr"
             type="org:addrType" minOccurs="0"/>
         </sequence>
         <attribute name="type"
           type="org:postalInfoEnumType" use="required"/>
       </complexType>
```

```
      <!--
      Child response elements.
      -->
       <element name="chkData" type="org:chkDataType"/>
       <element name="creData" type="org:creDataType"/>
       <element name="infData" type="org:infDataType"/>

      <!--
      <check> response elements.
      -->
       <complexType name="chkDataType">
         <sequence>
           <element name="cd" type="org:checkType"
             maxOccurs="unbounded" />
         </sequence>
       </complexType>

       <complexType name="checkType">
         <sequence>
           <element name="id" type="org:checkIDType"/>
           <element name="reason" type="eppcom:reasonType"
             minOccurs="0"/>
         </sequence>
       </complexType>

       <complexType name="checkIDType">
         <simpleContent>
           <extension base="eppcom:clIDType">
             <attribute name="avail" type="boolean"
               use="required"/>
           </extension>
         </simpleContent>
       </complexType>

      <!--
      <info> response elements.
      -->

       <complexType name="infDataType">
         <sequence>
           <element name="id"
             type="eppcom:clIDType"/>
           <element name="roid"
             type="eppcom:roidType"/>
           <element name="role"
             type="org:roleType" maxOccurs="unbounded"/>
           <element name="status"
             type="org:statusType" maxOccurs="9"/>
```

```
        <element name="parentId"
          type="eppcom:clIDType" minOccurs="0"/>
        <element name="postalInfo"
          type="org:postalInfoType" minOccurs="0" maxOccurs="2"/>
        <element name="voice"
          type="org:e164Type" minOccurs="0"/>
        <element name="fax"
          type="org:e164Type" minOccurs="0"/>
        <element name="email"
          type="eppcom:minTokenType" minOccurs="0"/>
        <element name="url"
          type="anyURI" minOccurs="0"/>
        <element name="contact"
          type="org:contactType" minOccurs="0"
          maxOccurs="unbounded"/>
        <element name="clID"
          type="eppcom:clIDType" minOccurs="0"/>
        <element name="crID"
          type="eppcom:clIDType"/>
        <element name="crDate"
          type="dateTime"/>
        <element name="upID"
          type="eppcom:clIDType" minOccurs="0"/>
        <element name="upDate"
          type="dateTime" minOccurs="0"/>
      </sequence>
    </complexType>
  <!--
  <create> response elements.
  -->
    <complexType name="creDataType">
      <sequence>
        <element name="id" type="eppcom:clIDType"/>
        <element name="crDate" type="dateTime"/>
      </sequence>
    </complexType>

  <!--

  End of schema.
  -->
  </schema>
  END
```

6. Internationalization Considerations

   EPP is represented in XML, which provides native support for encoding
   information using the Unicode character set [UNICODE] and its more
   compact representations, including UTF-8. Conformant XML processors
   recognize both UTF-8 [RFC3629] and UTF-16 [RFC2781]. Though XML
   includes provisions to identify and use other character encodings
   through use of an "encoding" attribute in an <?xml?> declaration, use
   of UTF-8 is RECOMMENDED.

   As an extension of the EPP organization object mapping, the elements
   and element content described in this document MUST inherit the
   internationalization conventions used to represent higher-layer
   domain and core protocol structures present in an XML instance that
   includes this extension.

7. IANA Considerations

7.1. XML Namespace

   This document uses URNs to describe XML namespaces and XML schemas
   conforming to a registry mechanism described in [RFC3688]. IANA has
   assigned the following URI.

   The organization namespace:

      URI: urn:ietf:params:xml:ns:epp:org-1.0

      Registrant Contact: IESG

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

   The organization XML schema:

      URI: urn:ietf:params:xml:schema:epp:org-1.0

      Registrant Contact: IESG

      XML: See the "Formal Syntax" section of RFC 8543 (this document).

7.2. EPP Extension Registry

The EPP extension described in this document has been registered by
IANA in the "Extensions for the Extensible Provisioning Protocol
(EPP)" registry described in [RFC7451]. The details of the
registration are as follows:

Name of Extension: Extensible Provisioning Protocol (EPP)
Organization Mapping

Document status: Standards Track

Reference: RFC 8543

Registrant Name and Email Address: IESG, iesg@ietf.org

TLDs: Any

IPR Disclosure: None

Status: Active

Notes: None

7.3. Role Type Values Registry

IANA has created a new category of protocol registry for values of
the organization roles. The name of this registry is "EPP
Organization Role Values". The registration policy for this registry
is "Expert Review" [RFC8126].

7.3.1. Registration Template

Value: The string value being registered.

Description: Brief description of the organization role values.

Registrant Name: For RFC specifications, state "IESG". For other
specifications, give the name of the responsible party.

Registrant Contact Information: An email address, postal address, or
some other information to be used to contact the registrant.

7.3.2. Initial Registry Contents

   The following are the initial registry contents:

      Value: registrar

      Description: The entity object instance represents the authority
      responsible for the registration in the registry.

      Registrant: IESG, iesg@ietf.org

      Value: reseller

      Description: The entity object instance represents a third party
      through which the registration was conducted (i.e., not the
      registry or registrar).

      Registrant: IESG, iesg@ietf.org

      Value: privacyproxy

      Description: The entity object instance represents a third party
      who could help to register a domain without exposing the
      registrants' private information.

      Registrant: IESG, iesg@ietf.org

      Value: dns-operator

      Description: The entity object instance represents a third-party
      DNS operator that maintains the name servers and zone data on
      behalf of a registrant.

      Registrant: IESG, iesg@ietf.org

8. Security Considerations

   The organization object may have personally identifiable information,
   such as <org:contact>. This information is not a required element in
   this document that can be provided on a voluntary basis. If it is
   provided, both client and server MUST ensure that authorization
   information is stored and exchanged with high-grade encryption
   mechanisms to provide privacy services, which are specified in
   [RFC5733]. The security considerations described in [RFC5730] or
   those caused by the protocol layers used by EPP will apply to this
   specification as well.

9. References

9.1. Normative References

[RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
            Requirement Levels", BCP 14, RFC 2119,
            DOI 10.17487/RFC2119, March 1997,
            <https://www.rfc-editor.org/info/rfc2119>.

[RFC3629]   Yergeau, F., "UTF-8, a transformation format of ISO
            10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November
            2003, <https://www.rfc-editor.org/info/rfc3629>.

[RFC3688]   Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688,
            DOI 10.17487/RFC3688, January 2004,
            <https://www.rfc-editor.org/info/rfc3688>.

[RFC3986]   Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
            Resource Identifier (URI): Generic Syntax", STD 66,
            RFC 3986, DOI 10.17487/RFC3986, January 2005,
            <https://www.rfc-editor.org/info/rfc3986>.

[RFC5322]   Resnick, P., Ed., "Internet Message Format", RFC 5322,
            DOI 10.17487/RFC5322, October 2008,
            <https://www.rfc-editor.org/info/rfc5322>.

[RFC5646]   Phillips, A., Ed. and M. Davis, Ed., "Tags for Identifying
            Languages", BCP 47, RFC 5646, DOI 10.17487/RFC5646,
            September 2009, <https://www.rfc-editor.org/info/rfc5646>.

[RFC5730]   Hollenbeck, S., "Extensible Provisioning Protocol (EPP)",
            STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009,
            <https://www.rfc-editor.org/info/rfc5730>.

[RFC5733]   Hollenbeck, S., "Extensible Provisioning Protocol (EPP)
            Contact Mapping", STD 69, RFC 5733, DOI 10.17487/RFC5733,
            August 2009, <https://www.rfc-editor.org/info/rfc5733>.

[RFC8126]   Cotton, M., Leiba, B., and T. Narten, "Guidelines for
            Writing an IANA Considerations Section in RFCs", BCP 26,
            RFC 8126, DOI 10.17487/RFC8126, June 2017,
            <https://www.rfc-editor.org/info/rfc8126>.

[RFC8174]   Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
            2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
            May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [UNICODE] The Unicode Consortium, "The Unicode Standard",
              <http://www.unicode.org/versions/latest/>.

   [W3C.REC-xml-20081126]
              Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E., and
              F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth
              Edition)", World Wide Web Consortium Recommendation
              REC-xml-20081126, November 2008,
              <https://www.w3.org/TR/xml/>.

   [W3C.REC-xmlschema-1-20041028]
              Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn,
              "XML Schema Part 1: Structures Second Edition", World Wide
              Web Consortium Recommendation REC-xmlschema-1-20041028,
              October 2004,
              <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>.

   [W3C.REC-xmlschema-2-20041028]
              Biron, P. and A. Malhotra, "XML Schema Part 2: Datatypes
              Second Edition", World Wide Web Consortium Recommendation
              REC-xmlschema-2-20041028, October 2004,
              <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>.

9.2. Informative References

   [RFC2781] Hoffman, P. and F. Yergeau, "UTF-16, an encoding of ISO
              10646", RFC 2781, DOI 10.17487/RFC2781, February 2000,
              <https://www.rfc-editor.org/info/rfc2781>.

   [RFC7451] Hollenbeck, S., "Extension Registry for the Extensible
              Provisioning Protocol", RFC 7451, DOI 10.17487/RFC7451,
              February 2015, <https://www.rfc-editor.org/info/rfc7451>.

Authors' Addresses

   Linlin Zhou
   CNNIC
   4 South 4th Street, Zhongguancun, Haidian District
   Beijing, Beijing 100190
   China

   Email: zhoulinlin@cnnic.cn

   Ning Kong
   Consultant

   Email: ietfing@gmail.com

   Jiankang Yao
   CNNIC
   4 South 4th Street, Zhongguancun, Haidian District
   Beijing, Beijing 100190
   China

   Email: yaojk@cnnic.cn

   James Gould
   VeriSign, Inc.
   12061 Bluemont Way
   Reston, VA 20190
   United States of America

   Email: jgould@verisign.com
   URI: http://www.verisign.com

   Guiqing Zhou

   Email: qing.joe@gmail.com