   Organization Extension for the Extensible Provisioning Protocol (EPP)

Abstract

   This document describes an extension to Extensible Provisioning
   Protocol (EPP) object mappings that is designed to support assigning
   an organization to any existing object (domain, host, contact) as
   well as any future objects.

Status of This Memo

Copyright Notice

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 are supported in the Extensible Provisioning Protocol (EPP) by the "organization" entities in [RFC8543]. This document provides a way to associate any EPP object such as domain names in [RFC5731], hosts in [RFC5732], and contacts in [RFC5733] to "organization" entities in [RFC8543]. The examples provided in this document are used for the domain object for illustration purposes. The host and contact object could be extended in the same way as the domain object.

Organization object identifiers, defined in [RFC8543], MUST be known to the server before the organization object can be associated with the EPP object.

This document 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 "orgext" is used for the namespace "urn:ietf:params:xml:ns:epp:orgext-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

This extension adds additional elements to EPP object mappings such as the EPP domain name mapping [RFC5731]. Only the new elements are described here.

3.1. Organization Identifier

The organization identifier provides the ID of an organization. Its corresponding element is <orgext:id>, which refers to the <org:id> element defined in [RFC8543]. All organization objects are identified by a server-unique identifier. A "role" attribute is used to represent the relationship that the organization has to the EPP object. Any given object MUST have at most one associated organization ID for any given role value.

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 assigning organizations to EPP objects.

4.1. EPP Query Commands

EPP provides three commands to retrieve EPP object information: <check> to determine if an object can be provisioned within a repository, <info> to retrieve detailed information associated with an object, and <transfer> to retrieve object transfer status information.

4.1.1. EPP <check> Command

This extension does not add any elements to the EPP <check> command or <check> response described in the EPP object mapping.

4.1.2. EPP <info> Command

This extension does not add any elements to the EPP <info> command described in the EPP object mapping. However, additional elements are defined for the <info> response in the EPP object mapping.

When an <info> command has been processed successfully, the EPP <resData> element MUST contain child elements as described in the EPP object extensions. In addition, the EPP <extension> element SHOULD contain a child <orgext:infData> element. This element is returned if the object has data that is associated with this extension and that is based on server policy. This element or its ancestor element

MUST identify the extension namespace
"urn:ietf:params:xml:ns:epp:orgext-1.0". The <orgext:infData>
element contains the following child elements:

o Zero or more <orgext:id> elements are allowed that contain the
   identifier of the organization, as defined in Section 3.1. The
   "role" attribute is used to represent the relationship that the
   organization has to the object. See Section 7.3 of [RFC8543] for
   a list of values.

Example <info> response for an authorized client with multiple
organizations:

```
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-US">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <domain:infData
S:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:        <domain:name>example.com</domain:name>
S:        <domain:roid>EXAMPLE1-REP</domain:roid>
S:        <domain:status s="ok"/>
S:        <domain:registrant>jd1234</domain:registrant>
S:        <domain:contact type="admin">sh8013</domain:contact>
S:        <domain:contact type="billing">sh8013</domain:contact>
S:        <domain:contact type="tech">sh8013</domain:contact>
S:        <domain:ns>
S:          <domain:hostObj>ns1.example.com</domain:hostObj>
S:        </domain:ns>
S:        <domain:clID>ClientX</domain:clID>
S:        <domain:crID>ClientY</domain:crID>
S:        <domain:crDate>2015-02-06T04:01:21.0Z</domain:crDate>
S:        <domain:exDate>2018-02-06T04:01:21.0Z</domain:exDate>
S:        <domain:authInfo>
S:          <domain:pw>2fooBAR</domain:pw>
S:        </domain:authInfo>
S:      </domain:infData>
S:    </resData>
S:    <extension>
S:      <orgext:infData
S:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
S:        <orgext:id role="reseller">reseller1523</orgext:id>
S:        <orgext:id role="privacyproxy">proxy2935</orgext:id>
S:      </orgext:infData>
S:    </extension>
S:    <trID>
S:      <clTRID>ngcl-IvJjzMZc</clTRID>
S:      <svTRID>test142AWQONJZ</svTRID>
S:    </trID>
S:  </response>
S:</epp>
```

Example <info> response for an authorized client with no
organization:

```
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-US">Command completed successfully</msg>
S:    </result>
S:    <resData>
S:      <domain:infData
S:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:        <domain:name>example.com</domain:name>
S:        <domain:roid>EXAMPLE1-REP</domain:roid>
S:        <domain:status s="ok"/>
S:        <domain:registrant>jd1234</domain:registrant>
S:        <domain:contact type="admin">sh8013</domain:contact>
S:        <domain:contact type="billing">sh8013</domain:contact>
S:        <domain:contact type="tech">sh8013</domain:contact>
S:        <domain:ns>
S:          <domain:hostObj>ns1.example.com</domain:hostObj>
S:        </domain:ns>
S:        <domain:clID>ClientX</domain:clID>
S:        <domain:crID>ClientY</domain:crID>
S:        <domain:crDate>2015-02-06T04:01:21.0Z</domain:crDate>
S:        <domain:exDate>2018-02-06T04:01:21.0Z</domain:exDate>
S:        <domain:authInfo>
S:          <domain:pw>2fooBAR</domain:pw>
S:        </domain:authInfo>
S:      </domain:infData>
S:    </resData>
S:    <extension>
S:      <orgext:infData
S:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0"/>
S:    </extension>
S:    <trID>
S:      <clTRID>ngcl-IvJjzMZc</clTRID>
S:      <svTRID>test142AWQOONJZ</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

This extension does not add any elements to the EPP <transfer> query
command or <transfer> query response described in the EPP object
mapping.

4.2. EPP Transform Commands

EPP provides five commands to transform EPP objects: <create> to
create an instance of an object, <delete> to delete an instance of an
object, <renew> to extend the validity period of an object,
<transfer> to manage the object sponsorship changes, and <update> to
change information associated with an object.

4.2.1. EPP <create> Command

This extension defines additional elements for the EPP <create>
command described in the EPP object extensions. No additional
elements are defined for the EPP <create> response.

The EPP <create> command provides a transform operation that allows a
client to create an object. In addition to the EPP command elements
described in the EPP object extensions, the command MUST contain an
<extension> element, and the <extension> element MUST contain a child
<orgext:create> element. This element is used if the client wants to
associate data defined in this extension to the object. This element
or its ancestor element MUST identify the extension namespace
"urn:ietf:params:xml:ns:epp:orgext-1.0". The <orgext:create> element
contains the following child elements:

o One or more <orgext:id> elements that contain the identifier of
   the organization, as defined in Section 3.1. The "role" attribute
   is used to represent the relationship that the organization has to
   the object. See Section 7.3 of [RFC8543] for a list of values.

Example <create> command with only one organization:

```
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:      <domain:create
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:        <domain:period unit="y">3</domain:period>
C:        <domain:ns>
C:          <domain:hostObj>ns1.example.com</domain:hostObj>
C:        </domain:ns>
C:        <domain:registrant>jd1234</domain:registrant>
C:        <domain:contact type="tech">sh8013</domain:contact>
C:        <domain:contact type="billing">sh8013</domain:contact>
C:        <domain:contact type="admin">sh8013</domain:contact>
C:        <domain:authInfo>
C:          <domain:pw>fooBAR</domain:pw>
C:        </domain:authInfo>
C:      </domain:create>
C:    </create>
C:    <extension>
C:      <orgext:create
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:id role="reseller">reseller1523</orgext:id>
C:      </orgext:create>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <create> command with multiple organizations:

```
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:      <domain:create
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:        <domain:period unit="y">3</domain:period>
C:        <domain:ns>
C:          <domain:hostObj>ns1.example.com</domain:hostObj>
C:        </domain:ns>
C:        <domain:registrant>jd1234</domain:registrant>
C:        <domain:contact type="tech">sh8013</domain:contact>
C:        <domain:contact type="billing">sh8013</domain:contact>
C:        <domain:contact type="admin">sh8013</domain:contact>
C:        <domain:authInfo>
C:          <domain:pw>fooBAR</domain:pw>
C:        </domain:authInfo>
C:      </domain:create>
C:    </create>
C:    <extension>
C:      <orgext:create
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:id role="reseller">reseller1523</orgext:id>
C:        <orgext:id role="privacyproxy">proxy2935</orgext:id>
C:      </orgext:create>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When a <create> command has been processed successfully, the EPP
response is as described in the EPP object extension.

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

4.2.2. EPP <delete> Command

This extension does not add any elements to the EPP <delete> command
or <delete> response described in the EPP object mapping.

4.2.3. EPP <renew> Command

This extension does not add any elements to the EPP <renew> command
or <renew> response described in the EPP object mapping.

4.2.4. EPP <transfer> Command

   This extension does not add any elements to the EPP <transfer>
   command or <transfer> response described in the EPP object mapping,
   but after a successful transfer of an object with an assigned
   organization, the handling of the assigned organization is dependent
   on the organization roles and server policy.

4.2.5. EPP <update> Command

   This extension defines additional elements for the EPP <update>
   command described in the EPP domain mapping [RFC5731], host mapping
   [RFC5732], and contact mapping [RFC5733]. No additional elements are
   defined for the EPP <update> response.

   The EPP <update> command provides a transform operation that allows a
   client to modify the attributes of an object. In addition to the EPP
   <update> command elements, the command MUST contain an <extension>
   element, and the <extension> element MUST contain a child
   <orgext:update> element. This element is used if the client wants to
   update the object with data defined in this extension. This element
   or its ancestor element MUST identify the extension namespace
   "urn:ietf:params:xml:ns:epp:orgext-1.0". The <orgext:update> element
   contains the following child elements:

   o An OPTIONAL <orgext:add> element that contains one or more
      <orgext:id> elements, as defined in Section 3.1, that add
      nonexistent organization roles to the object. The <orgext:id>
      element MUST have a non-empty organization identifier value. The
      server SHOULD validate that the <orgext:id> element role does not
      exist.

   o An OPTIONAL <orgext:rem> element that contains one or more
      <orgext:id> elements, as defined in Section 3.1, that remove
      organization roles from the object. The <orgext:id> element MAY
      have an empty organization identifier value. The server SHOULD
      validate the existence of the <orgext:id> element role and the
      organization identifier if provided.

   o An OPTIONAL <orgext:chg> element that contains one or more
      <orgext:id> elements, as defined in Section 3.1, that change
      organization role identifiers for the object. The existing
      organization identifier value will be replaced for the defined
      role. The server SHOULD validate the existence of the <orgext:id>
      element role.

   At least one <orgext:add>, <orgext:rem>, or <orgext:chg> element MUST
   be provided.

Example <update> command, adding a reseller:

```
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:      <domain:update
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:      </domain:update>
C:    </update>
C:    <extension>
C:      <orgext:update
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:add>
C:          <orgext:id role="reseller">reseller1523</orgext:id>
C:        </orgext:add>
C:      </orgext:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <update> command, adding multiple organizations:

```
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:      <domain:update
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:      </domain:update>
C:    </update>
C:    <extension>
C:      <orgext:update
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:add>
C:          <orgext:id role="reseller">reseller1523</orgext:id>
C:          <orgext:id role="privacyproxy">proxy2935</orgext:id>
C:        </orgext:add>
C:      </orgext:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <update> command, removing a reseller:

```
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:      <domain:update
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:      </domain:update>
C:    </update>
C:    <extension>
C:      <orgext:update
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:rem>
C:          <orgext:id role="reseller"/>
C:        </orgext:rem>
C:      </orgext:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <update> command, removing multiple organizations:

```
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:      <domain:update
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:      </domain:update>
C:    </update>
C:    <extension>
C:      <orgext:update
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:rem>
C:          <orgext:id role="reseller"/>
C:          <orgext:id role="privacyproxy"/>
C:        </orgext:rem>
C:      </orgext:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <update> command, updating reseller identifier:

```
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:      <domain:update
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:      </domain:update>
C:    </update>
C:    <extension>
C:      <orgext:update
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:chg>
C:          <orgext:id role="reseller">reseller1523</orgext:id>
C:        </orgext:chg>
C:      </orgext:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

Example <update> command, updating multiple organization identifiers:

```
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:      <domain:update
C:        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:        <domain:name>example.com</domain:name>
C:      </domain:update>
C:    </update>
C:    <extension>
C:      <orgext:update
C:        xmlns:orgext="urn:ietf:params:xml:ns:epp:orgext-1.0">
C:        <orgext:chg>
C:          <orgext:id role="reseller">reseller1523</orgext:id>
C:          <orgext:id role="privacyproxy">proxy2935</orgext:id>
C:        </orgext:chg>
C:       </orgext:update>
C:    </extension>
C:    <clTRID>ABC-12345</clTRID>
C:  </command>
C:</epp>
```

When an extended <update> command has been processed successfully,
the EPP response is as described in the EPP object extension.

An EPP error response MUST be returned if an <update> command cannot
be processed for any reason. An attempt to add one organization ID
or multiple organization IDs with a particular role value when at
least one of them already exists does not change the object at all.
A server SHOULD notify clients that object relationships exist by
sending a 2305 error response code. An attempt to remove an
organization ID or multiple organization IDs with a particular role
value when at least one of them does not exist does not change the
object at all. A server SHOULD notify clients that object
relationships do not exist by sending a 2305 error response code. An
attempt to change an organization ID or multiple organization IDs
with a particular role value when at least one of them does not exist
does not change the object at all. A server SHOULD notify clients
that object relationships do not exist by sending a 2305 error
response code. Response format with error value elements is defined
in Section 2.6 of [RFC5730].

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

  <annotation>
    <documentation>
      Extensible Provisioning Protocol v1.0
      Organization Extension Schema v1.0
    </documentation>
  </annotation>
```

```
      <!-- Child elements found in EPP commands -->
      <element
        name="create"
        type="orgext:createType"/>
      <element
        name="update"
        type="orgext:updateType"/>

      <!--
        Organization identifier with required role
      -->
      <complexType name="orgIdType">
        <simpleContent>
          <extension base="token">
            <attribute
              name="role"
              type="token"
              use="required"/>
          </extension>
        </simpleContent>
      </complexType>

      <!--
        Child elements of the <orgext:create> command.
        All elements must be present at time of creation.
      -->
      <complexType name="createType">
        <sequence>
          <!-- Agent identifier or the organization,
            e.g., registrar, reseller, privacy proxy, etc. -->
          <element
            name="id"
            type="orgext:orgIdType"
            maxOccurs="unbounded"/>
        </sequence>
      </complexType>

      <!--
        Child elements of <orgext:update> command
      -->
      <complexType name="updateType">
        <sequence>
          <element
            name="add"
            type="orgext:addRemChgType"
            minOccurs="0"
          />
```

```
      <element
        name="rem"
        type="orgext:addRemChgType"
        minOccurs="0"
      />
      <element
        name="chg"
        type="orgext:addRemChgType"
        minOccurs="0"
      />
    </sequence>
  </complexType>

  <complexType name="addRemChgType">
    <sequence>
      <!-- Agent identifier of the organization,
        e.g., registrar, reseller, privacy proxy, etc. -->
      <element
        name="id"
        type="orgext:orgIdType"
        maxOccurs="unbounded"/>
    </sequence>
  </complexType>

  <!-- Child response element -->
  <element
    name="infData"
    type="orgext:infDataType"/>

  <!-- <orgext:infData> response elements -->
  <complexType name="infDataType">
    <sequence>
      <!-- Agent identifier the organization,
        e.g., registrar, reseller, privacy proxy, etc. -->
      <element
        name="id"
        type="orgext:orgIdType"
        minOccurs="0"
        maxOccurs="unbounded"/>
    </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 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 extension namespace:

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

Registrant Contact: IESG

XML: See the "Formal Syntax" section of RFC 8544 (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: Organization Extension for the Extensible Provisioning Protocol (EPP)

Document Status: Standards Track

Reference: RFC 8544

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

TLDs: Any

IPR Disclosure: None

Status: Active

Notes: None

8. Security Considerations

The object mapping extension described in this document does not provide any other security services or introduce any additional considerations beyond those described by [RFC5730], [RFC5731], [RFC5732], and [RFC5733] or those caused by the protocol layers used by EPP.

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>.

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

   [RFC5731] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)
              Domain Name Mapping", STD 69, RFC 5731,
              DOI 10.17487/RFC5731, August 2009,
              <https://www.rfc-editor.org/info/rfc5731>.

   [RFC5732] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)
              Host Mapping", STD 69, RFC 5732, DOI 10.17487/RFC5732,
              August 2009, <https://www.rfc-editor.org/info/rfc5732>.

   [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>.

   [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>.

   [RFC8543]   Zhou, L., Kong, N., Yao, J., Gould, J., and G. Zhou,
               "Extensible Provisioning Protocol (EPP) Organization
               Mapping", RFC 8543, DOI 10.17487/RFC8543, March 2019,
               <https://www.rfc-editor.org/info/rfc8543>.

Acknowledgments

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

   Junkai Wei
   CNNIC
   4 South 4th Street, Zhongguancun, Haidian District
   Beijing, Beijing 100190
   China

   Email: weijunkai@cnnic.cn

   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